Induction of the Effects of Actions by Monotonic Methods

Ramon P. Otero

Department of Computer Science, University of Corunna
Corunna 15071, Galicia, Spain
otero@udc.es

Abstract. Induction of the effects of actions considered here consists in learning an action description of a dynamic system from evidence on its behavior. General logic-based induction methods can deal with this problem but, unfortunately, most of the solutions provided have the frame problem. To cope with the frame problem induction under suitable nonmonotonic formalisms has to be used, though this kind of induction is not well understood yet. We propose an alternative method that relies on the identification of a monotonic induction problem whose solutions correspond one-to-one to those of the original problem without the frame problem. From this result induction of the effects of actions can be characterized under current monotonic induction methods.

1 Preliminaries

Induction of the effects of actions considered here consists in learning an action description of a dynamic system from evidence on its behavior. The area of Inductive Logic Programming (ILP) that studies learning under logic-based formalisms, mainly logic programs (LP), defines an (*explanatory*) induction problem, in general, as follows. Given the sets of clauses E^+, called positive examples, negative examples E^-, and background knowledge B, find a set of clauses H such that

$$B \cup H \models e_i^+ \text{ for every } e_i^+ \in E^+,$$

$$B \cup H \not\models e_j^- \text{ for every } e_j^- \in E^-, \text{ and}$$

$$B \cup H \not\models \bot.$$

Sometimes it is also required that $B \not\models e_i^+$ for every $e_i^+ \in E^+$ and that $B \not\models e_j^-$ for every $e_j^- \in E^-$ but these will not be demanded in this work. On the other hand, it is usually assumed that H is a clause universally quantified at least in one variable.

Induction of the effects of actions can be defined as an induction problem in this framework. Evidence on the behavior of the dynamic system is assumed in the following form.

T. Horváth and A. Yamamoto (Eds.): ILP 2003, LNAI 2835, pp. 299–310, 2003.

Definition 1 *(Narrative) Given a domain with a set of fluent-names $\mathcal{F}$ and a set of action-names $\mathcal{A}$, a narrative is a pair (F, A) where each component is a set of ground unit clauses verifying,*

$$F \subseteq \{f(s_i), \neg f(s_i) \mid f \in \mathcal{F},\ 0 \leq s_i \leq n\}$$

$$A \subseteq \{a(s_i) \mid a \in \mathcal{A},\ 1 \leq s_i \leq n\}$$

Where $f(s_i)$ and $\neg f(s_i)$ are literals on fluent f, complementary to each other. The natural number s_i is the situation constant and 0 names the initial situation. The maximum number of situations in the narrative (length) is $(n+1)$.

A narrative is consistent iff the set of clauses F is consistent, i.e. F does not contain a complementary pair of literals $f(s_i)$ and $\neg f(s_i)$.

Definition 2 *Evidence is a set of narratives, for a given domain.*

Narratives correspond to facts true in the domain at different time points, represented by fluent literals at different situation constants; action literals represent the actions performed in the domain at the corresponding time. A narrative thus represents evidence on the particular behavior of the dynamic system after a particular sequence of actions. Different narratives correspond to different sequences of actions and different initial facts in the domain.

The action description of the domain is a set of action laws as follows.

Definition 3 *(Hypotheses) An action law on an effect of a fluent $f \in \mathcal{F}$ is a clause C*

$$e(S) \leftarrow a(S), prev(S, PS), \textsc{precond}(PS)$$

where $e(S)$ is $f(S)$ or $\neg f(S)$ for the fluent $f \in \mathcal{F}$ and is called the effect *literal; S, PS are situation variables (universally quantified); $a \in \mathcal{A}$; $\textsc{precond}(PS)$ may be missing and is a conjunction of literals $f'(PS)$ and $\neg f'(PS)$, with $f' \in \mathcal{F}$; the relation $prev(S, PS)$ is defined as the successor relation in naturals, i.e. S is the successor of PS.*

Each action law corresponds to one effect literal, the whole description will contain action laws for each effect on the fluents. The induction problem can be defined on one effect at a time, to this end the examples that correspond to the selected target effect are extracted from the evidence narratives.

Definition 4 *(Examples) Given a narrative (F, A) and some effect literal e of a fluent in $\mathcal{F}$, the set of positive examples on e is*

$$E^+ = \{e(s_i) \mid e(s_i) \in F,\ 1 \leq s_i\}$$

and the set of negative examples on e is

$$E^- = \{e(s_i) \mid \overline{e(s_i)} \in F,\ 1 \leq s_i\}$$

where $\overline{e(s_i)}$ is the literal complementary to $e(s_i)$.

Definition 5 *(Induction of the Effects of Actions from One Narrative) Given some evidence $\mathcal{E} = \{(F, A)\}$ on a narrative of a domain and a target effect literal e on a fluent in $\mathcal{F}$, a solution to induction of the effect literal e is a set of action laws H on it, verifying*

$$(X \cup F' \cup A \cup H) \models e^+(s_i) \quad \textit{for every} \quad e^+(s_i) \in E^+$$
$$(X \cup F' \cup A \cup H) \not\models e^-(s_i) \quad \textit{for every} \quad e^-(s_i) \in E^-$$
$$(X \cup F' \cup A \cup H) \not\models \bot$$

where $F' = (F \backslash E^+)$, E^+ and E^- are the set of positive and negative examples on e, and X is a set of clauses defining relation $prev(S, PS)$.

It follows that this induction problem is a case of (explanatory) induction in ILP, with background $B = (X \cup F' \cup A)$.

Definition 6 *(Induction of the Effects of Actions) Given some evidence $\mathcal{E} = \{(F_l, A_l),\ 1 \leq l \leq m\}$ on m narratives of a domain and a target effect literal e on a fluent in $\mathcal{F}$, a solution to induction of the effect literal e is a set of action laws H on it, verifying that H is a solution to induction on every narrative $(F_l, A_l) \in \mathcal{E}$.*

Proposition 1. *There is no solution to induction from inconsistent narratives.*

Proof. For inconsistent narratives two literals $e(s_i)$ and $\overline{e(s_i)}$ belong to F for some fluent f and situation constant s_i. If $e(s_i)$ is a target effect literal, by definition of examples, $e(s_i)$ belongs to both E^+ and E^- then it must be entailed and not entailed by every solution to induction, which is impossible. If $e(s_i)$ is not a target effect, both $e(s_i)$ and $\overline{e(s_i)}$ are included in F', then (monotonically) entailed by it and no solution to induction can satisfy the condition $(X \cup F' \cup A \cup H) \not\models \bot$. ■

As inconsistent narratives in any fluent do not seem to represent actual behaviors, we will consider induction problems from consistent narratives.

Most of the solutions to induction as defined before have the *frame problem*, that can be defined as follows in the represention we are using.

Definition 7 *(Frame Problem) A solution H on effect literal e has the frame problem iff it contains an action law that includes $e(PS)$ in* PRECOND(PS) *(a frame axiom).*

To show the existence of the frame problem, consider an action that does not affect the target fluent—this is a common case as fluents are not affected by every action. The narrative likely contains two situations at which this action is performed but the target fluent holds on complementary literals. If the rest of the fluents are the same at both situations, the only available difference to explain the complementary effects of the action is the literal of the fluent at the

previous situation, which is (respectively) the same as the action has no effect on it. Notice that the number of frame axioms needed for a domain is the number of fluents times the number of actions which do not affect the fluent.

The definition of induction cannot be simply restricted disallowing frame axioms in H as most of the domains will not have solution. The frame problem is solved by inertia axioms under suitable nonmonotonic formalisms. The nonmonotonic behavior of LP due to *negation as failure* (NAF) can be used to represent inertia axioms.

Definition 8 *(Inertia Axioms) An inertia axiom on a literal e of a fluent in $\mathcal{F}$ is a clause I*

$$e(S) \leftarrow prev(S, PS), e(PS), not\ \overline{e(S)}$$

where S, PS are situation variables (universally quantified), $\overline{e(S)}$ is the literal complementary to $e(S)$, $prev(S, PS)$ is the successor relation as defined before.

Inertia axioms do not mention action literals, being just one of these enough for each effect, strongly reducing the size of the description. Furthermore they have a general form, independent on the domain, and known in advance.

Definition 9 *(Induction of the Effects of Actions without the Frame Problem) Given some evidence $\mathcal{E} = \{(F, A)\}$ on a narrative of a domain and a target effect literal e on a fluent in $\mathcal{F}$, a solution to induction of the effect literal e is a set of action laws H_I on it, not including frame axioms, verifying*

$$(X \cup I \cup F' \cup A \cup H_I) \models e^+(s_i) \quad \textit{for every} \quad e^+(s_i) \in E^+$$

$$(X \cup I \cup F' \cup A \cup H_I) \not\models e^-(s_i) \quad \textit{for every} \quad e^-(s_i) \in E^-$$

$$(X \cup I \cup F' \cup A \cup H_I) \not\models \bot$$

where $F' = (F \backslash E^+)$, E^+ and E^- are the set of positive and negative examples on e, X is a set of clauses defining relation $prev(S, PS)$, and I is the inertia axiom on e.

Again this induction problem is a case of (explanatory) induction in ILP, with background $B = (X \cup I \cup F' \cup A)$. This time however, B is a normal logic program using negation as failure, *not* operator, in the inertia axioms. Unfortunately, it is not known how to efficiently solve an induction problem under normal background knowledge. But see [1] for a characterization of induction in normal logic programs. As an indication of the nonmonotonic behavior note that it will usually be the case that $B \models e^-(s_i)$ for some $e^-(s_i) \in E^-$, while this will not imply that there is no solution, for this reason the condition was not required in the definition.

2 Monotonic Method

Once inertia axioms are included in the background knowledge some of the examples are entailed by inertia—the so called *persistent* examples. We can focus our attention on the rest of the examples and define the sub-problem of induction from the examples that can never be entailed from just inertia. We call these, *examples on change*.

Definition 10 *(Examples on Change) Given a narrative* (F, A) *and some effect literal* e *of a fluent in* $\mathcal{F}$*, the set of positive examples on change to* e *is*

$$PE^+ = \{e(s_i) \mid e(s_i) \in F \text{ and } \overline{e(s_{i-1})} \in F\}$$

and the set of negative examples on change to e *is*

$$PE^- = \{e(s_i) \mid \overline{e(s_i)} \in F,\ 1 \leq s_i\}$$

where $\overline{e(s_i)}$ *is the literal complementary to* $e(s_i)$ *and* s_i*,* s_{i-1} *verify* $prev(s_i, s_{i-1})$*.*

Compared with the whole sets of examples, $PE^+ \subseteq E^+$ and $PE^- = E^-$. Notice that positive examples on change intuitively correspond with change on the fluent whereas negative examples are not defined following the intuition of non-change on the fluent, this is important for completeness.

Definition 11 *(Monotonic Induction of the Effects of Actions) Given some evidence* $\mathcal{E} = \{(F, A)\}$ *on a narrative of a domain and a target effect literal* e *on a fluent in* $\mathcal{F}$*, a solution to monotonic induction of the effect literal* e *is a set of action laws* H_M *on it, not including frame axioms, verifying*

$$(X \cup F' \cup A \cup H_M) \models e^+(s_i) \quad \textit{for every} \quad e^+(s_i) \in PE^+$$

$$(X \cup F' \cup A \cup H_M) \not\models e^-(s_i) \quad \textit{for every} \quad e^-(s_i) \in PE^-$$

$$(X \cup F' \cup A \cup H_M) \not\models \bot$$

where $F' = (F \setminus E^+)$*,* PE^+ *and* PE^- *are the set of positive and negative examples on change to literal* e*,* E^+ *is the set of positive examples on literal* e*, and* X *is a set of clauses defining relation* $prev(S, PS)$*.*

This induction problem is a case of (explanatory) induction in ILP, with background $B = (X \cup F' \cup A)$, and it is a case of monotonic induction as B, H_M, PE^+, and PE^- are sets of Horn clauses.

3 Correspondence

Solutions to the monotonic induction problem correspond one-to-one to solutions to induction of the effects of actions without the frame problem.

Definition 12 *(Complete Narrative) A narrative (F, A) is complete on fluent f iff for every situation s_i, $0 \leq s_i \leq n$ there is either $f(s_i) \in F$ or $\neg f(s_i) \in F$, where $(n+1)$ is the length of the narrative.*

Proposition 2. *(Correspondence) From evidence on narratives consistent and complete on the target fluent, H_I is a solution of nonmonotonic induction with inertia (Definition 9) if and only if it is a solution of monotonic induction (Definition 11).*

Proof. 1. ($H_I \Leftarrow H_M$) Consider a monotonic solution H_M is not a solution of the nonmonotonic induction problem. Then one of these is true:

a) there is some $e(s_i) \in E^+$ such that $(X \cup I \cup F' \cup A \cup H_M) \not\models e(s_i)$,
b) there is some $e(s_i) \in E^-$ such that $(X \cup I \cup F' \cup A \cup H_M) \models e(s_i)$,
c) $(X \cup I \cup F' \cup A \cup H_M) \models \bot$.

In case (a), consider $e(s_i)$ corresponds in monotonic induction to an example on change, $e(s_i) \in PE^+$, recall $PE^+ \subseteq E^+$. Then $(X \cup F' \cup A \cup H_M) \models e(s_i)$, and it follows that $(X \cup I \cup F' \cup A \cup H_M) \models e(s_i)$ because the set of clauses $(X \cup F' \cup A \cup H_M) \not\models \bot$ and it is a Horn program, thus its consequences are monotonically preserved in any other program it belongs to.

Alternatively, consider $e(s_i)$ does not correspond to an example on change, $e(s_i) \notin PE^+$. There are two cases here depending on whether the monotonic solution H_M eventually implies $e(s_i)$ or not. (Note that $e(s_i) \notin PE^-$, as $e(s_i) \in E^+$, $PE^- = E^-$ and F is assumed consistent, thus a solution to induction is free implying the instance $e(s_i)$ or not.) If eventually $(X \cup F' \cup A \cup H_M) \models e(s_i)$, then we already shown that $e(s_i)$ also follows in nonmonotonic induction.

Alternatively, if $(X \cup F' \cup A \cup H_M) \not\models e(s_i)$, then we show that $(X \cup I \cup F' \cup A \cup H_M) \models e(s_i)$, i.e. $e(s_i)$ is entailed by inertia. By definition of PE^+, as $e(s_i) \notin PE^+$ and $e(s_i) \in E^+$, then $\overline{e(s_{i-1})} \notin F$. As narratives are complete on target fluent it follows $e(s_{i-1}) \in F$, then also $e(s_{i-1}) \in E^+$. Consider the inertia axiom for the instance at s_i, $e(s_i) \leftarrow prev(s_i, s_{i-1}), e(s_{i-1}), not\overline{e(s_i)}$, it is the case that $\overline{e(s_i)} \notin F'$ because $e(s_i) \in E^+$, thus $e(s_i) \in F$, and F is consistent. Then $not\,\overline{e(s_i)}$ holds as there is no other means of inferring $\overline{e(s_i)}$. That $e(s_{i-1})$ is also entailed follows from mathematical induction over situations, as $e(s_{i-1}) \in E^+$. Mathematical induction ends always in some $e(s_{i-k})$ in one of two cases (being all the $e(s_{i-j}) \in E^+, 1 \leq j \leq k$ as shown before): either $e(s_{i-k}) \in PE^+$, in which case we already shown it follows in nonmonotonic induction, or alternatively $e(s_{i-k}) = e(0)$, which also follows as $e(0) \in F'$ (notice that instances at initial situation do not belong to the example set).

In case (b), by definition of E^-, for every $e(s_i) \in E^-$, $\overline{e(s_i)} \in F$ thus also in F'. Then $\overline{e(s_i)}$ is monotonically entailed by $(X \cup F' \cup A)$. As H_M is solution and $PE^- = E^-$, $(X \cup F' \cup A \cup H_M) \not\models e(s_i)$. Consider the inertia axiom for the instance at s_i, $\overline{e(s_i)} \leftarrow prev(s_i, s_{i-1}), e(s_{i-1}), not\overline{e(s_i)}$, it is the case that $not\overline{e(s_i)}$ do not hold as $\overline{e(s_i)}$ is monotonically entailed by F', thus $(X \cup I \cup F' \cup A \cup H_M) \not\models e(s_i)$.

In case (c), as H_M is solution, $(X \cup F' \cup A \cup H_M) \not\models \bot$. The inertia axiom can only entail some additional $e(s_i)$, thus $(X \cup I \cup F' \cup A \cup H_M) \models \bot$ only if

the entailed $e(s_i) \in E^-$, thus $\overline{e(s_i)} \in F'$. We already shown in case (b) that this is not the case.

2. ($H_I \Rightarrow H_M$) Consider a nonmonotonic solution H_I is not a solution of the monotonic induction problem. Then one of these is true:

a) there is some $e(s_i) \in PE^+$ such that $(X \cup F' \cup A \cup H_I) \not\models e(s_i)$,

b) there is some $e(s_i) \in PE^-$ such that $(X \cup F' \cup A \cup H_I) \models e(s_i)$,

c) $(X \cup F' \cup A \cup H_I) \models \bot$.

In case (a), consider some $e(s_i) \in PE^+$ such that $(X \cup F' \cup A \cup H_I) \not\models e(s_i)$, then consider the program with the inertia axiom, its instance at s_i, $e(s_i) \leftarrow prev(s_i, s_{i-1}), e(s_{i-1}), not\ \overline{e(s_i)}$, as $e(s_i) \in PE^+$ it follows $\overline{e(s_{i-1})} \in F$, thus $e(s_{i-1}) \in E^-$ and as H_I is solution $(X \cup I \cup F' \cup A \cup H_I) \not\models e(s_{i-1})$, then the inertia axiom is not applicable at s_i, leading to $(X \cup I \cup F' \cup A \cup H_I) \not\models e(s_i)$, which is contradictory with the assumption that H_I is solution, thus it must be that $(X \cup F' \cup A \cup H_I) \models e(s_i)$.

In case (b), for every $e(s_i) \in PE^-$ as $PE^- = E^-$, $(X \cup I \cup F' \cup A \cup H_I) \not\models e(s_i)$. Consider $(X \cup F' \cup A \cup H_I) \models e(s_i)$, as $(X \cup F' \cup A \cup H_I)$ is a Horn program its consequences are monotonically preserved in any other program it belongs to, then it will follow that $(X \cup I \cup F' \cup A \cup H_I) \models e(s_i)$ which contradicts the initial assumption on H_I, thus $(X \cup F' \cup A \cup H_I) \not\models e(s_i)$.

In case (c), consider $(X \cup F' \cup A \cup H_I) \models \bot$, it is a Horn program thus inconsistency will hold in any other program it belongs to, which is a contradiction with the assumption that $(X \cup I \cup F' \cup A \cup H_I) \not\models \bot$. ■

Under some conditions there are efficient monotonic methods sound and complete for induction under Horn logic programs. This is the case of our definition of monotonic induction of the effects of actions in which the background knowledge as well as the examples are sets of ground facts, and the hypotheses are function-free positive rules. Tractability further comes from the fact that hypotheses, as required for the representation of action laws, are 12-determinate clauses, with respect to the background and examples defined [2]. Domains may have a relational (static) structure for which fluents with more arguments than just one for the situation term are used, leading to higher values of i and j in the ij-determinacy of hypotheses, but efficiency holds as far as the ij-determinate restriction does.

Corollary 1. *The monotonic method provides an efficient, sound and complete induction of the effects of actions without the frame problem from evidence on consistent and complete narratives.*

Proof. Follows from the fact that there are efficient monotonic methods sound and complete for the setting we are using in monotonic induction of actions and from Prop. 2 establishing one-to-one correspondence to nonmonotonic induction with inertia. ■

4 Example

To illustrate the method consider the simple Yale Shooting Scenario. There is a turkey and a gun, the gun can be loaded or not, and the turkey will be dead when shooting with the gun loaded. There are actions shoot s, load l, and wait w; and fluents loaded ld, and dead d. Consider the following narrative (F, A)

$$\begin{aligned} F = \{&nld(0),\ nld(1),\ ld(2),\ ld(3),\ ld(4),\ ld(5),\\ &nd(0),\ nd(1),\ nd(2),\ nd(3),\ d(4),\ d(5)\}\\ A = \{&s(1),\ l(2),\ w(3),\ s(4),\ w(5)\} \end{aligned}$$

Where nd (not dead) is the complementary fluent[1] to d and nld the complementary of ld. An example of a learning problem in this scenario is the induction of a description of effect d, the examples from the narrative would be

$$\begin{aligned} E^+ &= \{d(4),\ d(5)\}\\ E^- &= \{d(1),\ d(2),\ d(3)\} \end{aligned}$$

After (direct) monotonic induction is applied to this problem one solution would be

$$\begin{aligned} H = \{\ &d(S) \leftarrow\ s(S),\ prev(S, PS),\ ld(PS).\\ &d(S) \leftarrow\ s(S),\ prev(S, PS),\ d(PS).\\ &d(S) \leftarrow\ w(S),\ prev(S, PS),\ d(PS).\\ &d(S) \leftarrow\ l(S),\ prev(S, PS),\ d(PS).\} \end{aligned}$$

The last three clauses are frame axioms, the solution has the frame problem. The examples on change corresponding to the narrative would be

$$\begin{aligned} PE^+ &= \{d(4)\}\\ PE^- &= \{d(1),\ d(2),\ d(3)\} \end{aligned}$$

After the monotonic induction method is applied to this problem one solution would be

$$H = \{\ d(S) \leftarrow\ s(S),\ prev(S, PS),\ ld(PS).\}$$

5 Extended Monotonic Method

Monotonic induction of the effects of actions introduced so far relies on narratives complete on the target fluent, this condition can be removed. When narratives are not complete, the target fluent has missing instances at some situation constants which makes the definition of examples on change incomplete.

[1] We are using a different predicate (nd) for the (classically) negated fluent ($\neg d$), this allows, as far as this work, for a Horn logic program representation—instead of an Extended logic program with classical negation—when the corresponding constraint ($\leftarrow d(S), nd(S).$) is in the program.

Definition 13 *(Examples on Change, Missing Instances) Given a narrative* (F, A) *and some effect literal* e *of a fluent in* $\mathcal{F}$*, the set of positive examples on change to* e *is*

$$\begin{aligned} PE'^{+} = \{e(s_i) \mid e(s_i) \in F \text{ and } \overline{e(s_{i-1})} \in F\} \cup \\ \{e(s'_i) \mid e(s_i) \in F \text{ and } \overline{e(s_{i-k})} \in F \text{ and} \\ \text{for all } s_j,\ s_{i-k} < s_j < s_i,\ e(s_j) \notin F \text{ and } \overline{e(s_j)} \notin F\} \end{aligned}$$

and the set of negative examples on change to e *is*

$$PE^{-} = \{e(s_i) \mid \overline{e(s_i)} \in F,\ 1 \leq s_i\}$$

where $\overline{e(s_i)}$ *is the literal complementary to* $e(s_i)$ *and* s_i*,* s_{i-1} *verify* $prev(s_i, s_{i-1})$ *as well as the* $<$ *relation represents the transitive closure of* $prev/2$*. For each* $e(s_i)$ *instance, the constant* s'_i *is a new situation constant not present elsewhere in the description.*

The new component of PE'^{+} corresponds to segments of consecutive situation constants with missing instances on the target fluent, the new situation constant s'_i can be understood as representing the missing segment as a whole. Intuitively a missing segment with complementary literals at both edge situations must contain an example on change, the available evidence in the narrative does not tell us the particular situation inside the segment at which the change occurs. This is indeed a case of *multiple instance* learning. To deal with it the representation of actions is also extended to have an extra argument for the missing segment the action belongs to in case there is one. For each narrative (F, A) missing segments are named with different new constants and given as extra argument to action instances, the new narrative (F, A') contains thus

$$A' = \{a(s'_i, s_i) \mid a(s_i) \in A\}$$

where s'_i is the constant naming the missing segment or just s_i if there is no missing target fluent at s_i. Correspondingly, action laws (hypotheses) will have the new form

$$e(ES) \leftarrow a(ES, S), prev(S, PS), \text{PRECOND}(PS)$$

Definition 11 of monotonic induction of actions is still valid considering the extended PE'^{+} set instead of PE^{+}, the narratives (F, A') instead of (F, A) and the new hypotheses language. The induction problem is still a case of (explanatory) induction in ILP, with background $B = (X \cup F' \cup A')$, and also a case of monotonic induction as B, H_M, PE'^{+}, and PE^{-} are sets of Horn clauses.

After induction of the action laws the extra situation argument for missing segments is discarded, so the hypotheses recover the regular form. Correspondence with nonmonotonic induction with inertia follows.

Proposition 3. *(Correspondence Extended) From evidence on narratives consistent,* H_I *is a solution of nonmonotonic induction with inertia (Definition 9)*

if and only if it is a solution of monotonic induction (Definition 11 with the extension).

Proof. We only shown the part of the proof corresponding to the extension wrt Prop. 2.

1. ($H_I \Leftarrow H_M$) Consider a monotonic solution H_M is not a solution of the nonmonotonic induction problem. Then one of these is true:

a) there is some $e(s_i) \in E^+$ such that $(X \cup I \cup F' \cup A \cup H_M) \not\models e(s_i)$,
b) there is some $e(s_i) \in E^-$ such that $(X \cup I \cup F' \cup A \cup H_M) \models e(s_i)$,
c) $(X \cup I \cup F' \cup A \cup H_M) \models \bot$.

In case (a), consider $e(s_i)$ corresponds in monotonic induction to an (extended) example on change, $e(s_i') \in PE'^+$, but $e(s_i) \notin PE^+$, then it must be the case that $e(s_i)$ is the ending edge of a missing segment, as H_M is solution the corresponding $e(s_i') \in PE'^+$ is (monotonically) entailed. After discarding the extra situation argument in H_M, the solution H_M' will entail one of the $e(s_j)$ inside the missing segment and inertia (added after learning) will propagate the instance from the induced situation of change to the ending edge of the missing segment.

In case (b) and (c) the proof is that of Prop. 2.

2. ($H_I \Rightarrow H_M$) Consider a nonmonotonic solution H_I is not a solution of the monotonic induction problem. Then one of these is true:

a) there is some $e(s_i) \in PE'^+$ such that $(X \cup F' \cup A' \cup H_I) \not\models e(s_i)$,
b) there is some $e(s_i) \in PE^-$ such that $(X \cup F' \cup A' \cup H_I) \models e(s_i)$,
c) $(X \cup F' \cup A \cup H_I) \models \bot$.

In case (a), consider some $e(s_i') \in PE'^+$, $e(s_i') \notin PE^+$, such that $(X \cup F' \cup A' \cup H_I) \not\models e(s_i')$, after H_I is rewritten in the extended form to include the reference to missing segments, the corresponding instance at $e(s_i')$ is entailed.

In case (b) and (c) the proof is that of Prop. 2. ■

The monotonic induction problem extended for missing segments requires the induction of nondeterminate clauses—the action literal is nondeterminate with respect to the background and examples. This compromises efficiency, as induction of 12-nondeterminate clauses is not PAC-learnable [3].

Corollary 2. *The extended monotonic method provides a sound and complete induction of the effects of actions without the frame problem from evidence on consistent narratives.* ■

To illustrate the extended method consider in the previous YSS domain the following narrative (F, A)

$$\begin{aligned} F = \{&nld(0),\ nld(1),\ nld(2),\ ld(3),\ ld(4),\ ld(5),\\ &nd(0),\ nd(1),\ nd(2),\ d(4),\ d(5)\}\\ A = \{&s(1),\ w(2),\ l(3),\ s(4),\ w(5)\} \end{aligned}$$

The (extended) examples on change corresponding to the narrative would be

$$PE'^{+} = \{d(m34)\}$$
$$PE^{-} = \{d(1),\ d(2)\}$$

Where $m34$ is the new situation constant naming the missing segment. The extended component of the narrative A' would be

$$A' = \{s(1,1),\ w(2,2),\ l(m34,3),\ s(m34,4),\ w(5,5)\}$$

After the extended monotonic induction method is applied to this problem one solution would be

$$H = \{\ d(ES) \leftarrow\ s(ES,S),\ prev(S,PS),\ ld(PS).\}$$

Discarding the extra ES term becomes the intended solution seen before.

6 Related Work and Discussion

Moyle and Muggleton [4] study the induction of Event Calculus programs, a particular action formalism, being [5] the most recent work. The methods proposed there are different and rely on working with negation as failure in the background (for inertia) during induction, but it has been shown [6], [1] that monotonic induction does not extend well to normal programs. The approach in [7] uses some causality-based formalism for actions, requiring examples on complete causality on the evidence, i.e. a closed world assumption is assumed on them, and the so-called nonmonotonic setting of induction is used. This restricts the range of applicability of the method, as causality is usually not directly observable in the domains. In our method evidence on change is extracted from regular evidence on the domain, and no closed world assumption is assumed on it. Another important difference with these two experimental approaches is that they are restricted to complete narratives.

The results provided here make induction of actions without the frame problem, a nonmonotonic induction problem that no efficient system can deal with, actually efficiently solvable using most of the current (monotonic) ILP systems.

Acknowledgments

I would like to thank the anonymous reviewers for helpful comments. This research is partially supported by the Government of Spain, National Commission of Science and Tecnology grant TIC-2001-0393 and by the Government of Galicia-Spain, Secretary of R&D grant PGIDIT-02-PXIC-10502PN.

References

1. Otero, R.: Induction of stable models. In: Proc. of the 11th Int. Conference on Inductive Logic Programming, ILP 01, LNAI 2157. (2001) 193–205
2. Muggleton, S., Feng, C.: Efficient induction of logic programs. Inductive Logic Programming (1992)
3. Kietz, J.: Some lower bounds on the computational complexity of inductive logic programming. In: Proc. of the 6th European Conference on Machine Learning, ECML 93, LNAI 667. (1993) 115–123
4. Moyle, S., Muggleton, S.: Learning programs in the event calculus. In: Proc. of the 7th Int. Workshop on Inductive Logic Programming, ILP 97, LNAI 1297. (1997) 205–212
5. Moyle, S.: Using theory completion to learn a robot navigation control program. In: Proc. of the 12th Int. Conf. on Inductive Logic Programming, ILP 02, LNAI 2583. (2003) 182–197
6. Sakama, C.: Inverse entailment in nonmonotonic logic programs. In: Proc. of the 10th Int. Conf. on Inductive Logic Programming, ILP 00, LNAI 1866. (2000) 209–224
7. Lorenzo, D., Otero, R.: Learning to reason about actions. In: Proc. of the 14th European Conference on Artificial Intelligence, ECAI 00. (2000) 316–320