Causal Theories of Action: A Computational Core

Jerome Lang
IRIT/UPS
F-31062 Toulouse Cedex, France
lang@irit.fr

Fangzhen Lin
Hong Kong Univ of Sci and Tech
Dept of Computer Science
Clear Water Bay, KLN, Hong Kong
flin@cs.ust.hk

Pierre Marquis
CRIL/Universite d* Artois
F-62307 Lens Cedex, France
marquis@cril.univ-artois.fr

Abstract

We propose a framework for simple causal theories of action, and study the computational complexity in it of various reasoning tasks such as determinism, progression and regression under various assumptions. As it turned out, even the simplest one among them, one-step temporal projection with complete initial state, is intractable. We also briefly consider an extension of the framework to allow truly indeterministic actions, and find that this extension does not increase the complexity of any of the tasks considered here.

1 Introduction

While there are abundance of formalisms for writing action theories that incorporate explicit causal rules not only between actions and fluents but also between fluents (e.g. [Baral, 1995; Lin, 1995; Thielscher, 1995; Gustafsson and Doherty, 1996; McCain and Turner, 1997; Zhang and Foo, 2001; Kakas *et ah,* 2001]), and there are some implementations of causal action theories iLin, 2000; McCain and Turner, 1998; Kakas *et al,* 2001], there have been few formal studies of complexities of various reasoning tasks in these causal action theories. In this paper, we investigate this issue. We first study various reasoning tasks such as computing the precondition of an action, checking if an action theory specifies a deterministic action, performing temporal projection, and computing regression in a very simple causal action framework that is primarily aimed at representing deterministic actions. Basically, an action theory for action a in this framework is just a finite set of action effect rules of the form

if $l_1 \wedge \cdots \wedge l_n$ is true, then action *a* causes / to be true

and causal rules of the form

$$l_1 \wedge \cdots \wedge l_n \text{ causes / to be true}$$

where l's are literals.

Surprisingly, even in this simple framework, one-step temporal projection problem with complete initial state in the following form: given a set of literals that completely determines the initial state, and an action a, determine if a given fluent will hold in the state resulted from doing in the initial state, is intractable. This compares with STRIPS and ADL, which are like this simple framework except that they do not allow causal rules, where problems like temporal projection and regression are easy.

As simple as this framework may be, it nonetheless captures the computational core of causal action theories. Indeed, we show that if one extends it to allow arbitrary formulas in both the right and left hand sides of the effect and causal rules: (1) this extends the expressiveness of the language, but (2) it does not affect the complexities of any of the reasoning tasks considered here.

The rest of this paper is organized as follows. After some formal preliminaries (Section 2), we present a framework for simple action theories in Section 3. Complexity results are presented in Section 4. Before presenting some related work and concluding, we briefly show in Section 5 how our simple action theories can be generalized, without questioning the complexity results (in the general case). For space reasons, proofs are only sketched or omitted.

2 Formal preliminaries

Let $PROP_{PS}$ be a propositional language defined inductively from a finite set *PS* of propositional symbols (atoms), the boolean constants $\top$ (true) and $\bot$ (false), and the connectives $\neg, \vee, \wedge, \Rightarrow$ and $\Leftrightarrow$. L_{PS} is the set of all literals generated from *PS.* For each formula ϕ, $Var(\phi)$ denotes the set of atoms occurring in ϕ.

We shall also consider two isomorphic copies of $PROP_{PS}$' $PROP_{PS_t}$ and $PROP_{PS_{t+1}}$. Each formula ϕ_t (resp. ϕ_{t+1}) from $PROP_{PS_t}$ (resp. $PROP_{PS_{t+1}}$) is obtained by substituting in a uniform way in the formula $\phi \in PROP_{PS}$ every atom $p \in PS$ by the atom p_t (resp. p_{t+1}). Propositions indexed by *t* are used to express conditions about current situation, and those indexed by $t+1$ about the situation resulted from doing action *a* in the current situation. In the following, by an *initial state formula* we mean a formula from $PROP_{PS_t}$, and by a *successor state formula* we mean a formula from $PROP_{PS_{t+1}}$. Finally, we shall consider the language $PROP_{PS_t \cup PS_{t+1}}$ and take advantage of it to characterize the effects of action a.

A truth assignment over *PS* is called a *complete state.* A truth assignment over PS_t is called a *complete initial state,* while a truth assignment over PS_{t+1} is called a *complete successor state.* In order to avoid heavy notations, we shall identify each complete state *s* with the conjunction of literals *s* is the unique model of it.

3 Simple action theories

We shall first consider *simple causal action theories.*

Definition 1 (simple action theory) *A* simple action theory *for action* α *is an ordered pair* $T_\alpha = \langle Eff(\alpha),$ *Causal) where*

- $Eff(\alpha)$ *is a finite set of* effect rules, *of the following form:*

$$l_1 \wedge \cdots \wedge l_n \xrightarrow{\alpha} l,$$

where l's are literals from Lps, and l can also he $\perp$. *The left side is understood to he a tautology when* $n = 0$. *When* $l = \perp$, *the effect rule is really an action precondition axiom about a. Notice that for our purpose in this paper, we assume that* α *is the only action.*

- *Causal is a finite set of* causal rules *of the following form:*

$$l_1 \wedge \cdots \wedge l_n \longrightarrow l,$$

where l's are literals from Lps. Again, the left side is understood to he α tautology when $n = 0$.

Such simple action theories are sufficient to encode actions with conditional effects. The ratification problem is addressed through causal rules, which represent static laws and express how fluents are related. While primarily targeted at representing deterministic actions, they can sometimes give rise to indeterministic effects when there arc cyclic causal rules such as $p \longrightarrow p$.

Causal theories of this form are special cases of causal theories in situation calculus [Lin, 1995], domain descriptions in action languages [Giunchiglia and Lifschitz, 1998], and other more general formalisms (e.g. iMcCain and Turner, 1997; Zhang and Foo, 20011). While these formalisms do not always agree on their semantics in the general case, most of them coincide on this special class.

Definition 2 (completion) *The* completion $cl(T_\alpha)$ *of a (simple) action theory* T_α *is the formula defined by*

- $T(\alpha) = \bigwedge\{pre_t \Rightarrow post_{t+1} \mid Eff(\alpha)$ *contains* $pre \xrightarrow{\alpha} post\}$ *;*
- $T(Causal) = \bigwedge\{left_t \Rightarrow right_t, left_{t+1} \Rightarrow right_{t+1} \mid Causal$ *contains* $left \longrightarrow right\}$*;*
- *for each literal* $l \in L_{PS}$, *let*
  $\Phi^\alpha(l) = \bigvee\{pre_t \mid Eff(\alpha)$ *contains* $pre \xrightarrow{\alpha} l\}$,
  $\Phi^C(l) = \bigvee\{left_t \mid Causal$ *contains* $left \longrightarrow l\}$,
  $\Phi(l) = \Phi^\alpha(l) \vee \Phi^C(l)$,
  $frame(l) = (l_t \wedge \neg\Phi(\bar{l})) \Rightarrow l_{t+1}$, *where* $\bar{p} = \neg p$ *for any atom p. The completion of* T_α *is defined as follows:*
- $cl(T_\alpha) = T(\alpha) \wedge T(Causal) \wedge \bigwedge_{l \in L_{PS}} frame(l)$.

We assume that T_α mentions each atom of PS (otherwise some atoms of PS would be useless); under this assumption, the size of $cl(T_\alpha)$ is polynomial in the size of T_α.

Given a simple action theory T_α, a truth assignment over $PROP_{PS_t \cup PS_{t+1}}$ is said to be *a model* of T_α if and only if it satisfies $cl(T_\alpha)$. Each model of T_α is formed by the union (s_t, s'_{t+1}) of a complete initial state s_t and a complete successor state s'_{t+1}.

As one can see, the construction of $cl(T_\alpha)$ from T_α is similar to the way a causal theory is transformed to classical logic in iLin, 1995] and to literal completions IMcCain and Turner, 1997; Lifschitz, 1997]. In fact, all these approaches yield logically equivalent theories.

4 Complexity issues

4.1 Executability and determinism

Several properties are of interest when dealing with action theories. First of all, there is the question of whether an action is executable in a given initial situation.

Definition 3 (executability) *Let* T_α *be an action theory for action* α *and let* φ *be a formula from PROPps-* $\langle T_\alpha, \varphi\rangle$ *is a positive instance of* EXECUTABILITY *(the executability problem) if and only if for any complete state s that satisfies* φ, *there is a state s' such that* $(s_t, s'_{t+1}) \models cl(T_\alpha)$.

Whenever $\langle T_\alpha, \varphi\rangle$ is a positive instance of EXECUTABILITY, *α* is said to be *executable under* φ; α is said to *be fully executable* if and only if it is executable under $\top$.

Proposition 1

1. In the general case, EXECUTABILITY *is* Π_2^p*- complete.*

2. Under the restriction where is a complete state, EXECUTABILITY *is HP-complete.*

Proof.

1. Membership is easy. Hardness comes from the following polynomial reduction from the Σ_2^p-complete problem QBF$_{2,\exists}$ to the problem of checking that α is not executable under $\top$. Let $Q = \langle\{a_1, \ldots, a_n\}, \{b_1, \ldots, b_p\}, \beta\rangle$ be an instance of QBF$_{2,\exists}$[1]. Wlog, we assume that β is a DNF formula $D_1 \vee \ldots \vee D_m$. We now define $M(Q) = T_\alpha = \langle\emptyset, Causal\rangle$ where $Causal = \{D_i \longrightarrow new, D_i \longrightarrow \neg new \mid i = 1 \ldots m\} \bigcup \{b_i \longrightarrow b_i, \neg b_i \longrightarrow \neg b_i \mid i = 1 \ldots p\}$ and new is a new atom, not occurring in β. Q is a positive instance of QBF$_{2,\exists}$ if and only if α (as given by $M(Q)$) is not executable under $\top$.
2. Comes directly from the fact that, when *ip* is complete, then *α* is executable under it iff the propositional theory $\{\varphi\} \cup cl(T_\alpha)$ is consistent. ∎

Another important property is *determinism.* Intuitively, an action is deterministic if there is at most one successor state corresponding to any initial state.

Definition 4 (determinism) *An action theory* T_α *for action* α *is a positive instance of* DETERMINISM *(the determinism problem) if and only if* α *is* deterministic, *i.e., for all* $s_t \in 2^{PS_t}$ *and for all* $s'_{t+1}, s''_{t+1} \in 2^{PS_{t+1}}$ *such that* $(s_t, s'_{t+1}) \models cl(T_\alpha)$ *and* $(s_t, s''_{t+1}) \models cl(T_\alpha)$, *then* $s'_{t+1} = s''_{t+1}$.

Proposition 2 DETERMINISM *is* coNP*-complete.*

[1]Let us recall that Q is a positive instance of QBF$_{2,\exists}$ if and only if it is valid, i.e., there exists an assignment $\vec{a}$ of variables $a_1, \ldots, a_n$ such that for all assignments $\vec{b}$ of variables $b_1, \ldots, b_p$ we have $(\vec{a}, \vec{b}) \models \beta$ (where β is a formula from $PROP_{PS}$ s.t. $Var(\beta) = \{a_1, \ldots, a_n, b_1, \ldots, b_p\}$). QBF$_{2,\exists}$ is still Σ_2^p-complete when β is under DNF.

Proof: Membership is easy. Hardness comes from the following polynomial reduction from DNF-VALIDITY. Let $\xi = D_1 \vee ... \vee D_m$ be any formula from $PROP_{PS}$ under DNF and let $M(\xi)$ be the instance of DETERMINISM defined by $T_\alpha = \langle Eff(\alpha), Causal \rangle$ with $Eff(\alpha) = \{D_1 \xrightarrow{\alpha} new, \ldots, D_m \xrightarrow{\alpha} new\}$ and $Causal = \{new \longrightarrow new, \neg new \longrightarrow \neg new\}$, where new is a new atom not mentioned i ξ. ξ s valid if and only if $M(\xi)$ is deterministic. ■

4.2 Progression

There are two forms *of the progression* problem, also referred to as (one-step) *temporal projection.* The simpler one, in the form of a query about the effects of an action given some information about the current situation, can be described as a triple $\langle T_\alpha, \phi, \psi \rangle$ where

- T_α is an action theory;
- ϕ is a formula from *PROPps* representing the knowledge state before the action is performed; we assume that the action α is executable under ϕ;
- ψ is a formula from *PROPps* we are interested in the truth value of which after the action has been performed.

Definition 5 (progression) *Let T_α be an action theory and ϕ a formula from PROPps such that (\ is executable under ϕ. Let ψ be a formula from PROPps- $\langle T_\alpha, \phi, \psi \rangle$ is a positive instance* ^/PROGRESSION *(the progression problem) if and only if ψ_{t+1} holds in any possible complete successor state by action α of any complete initial state Sf satisfying ϕ_t; equivalently:*

$$\phi_t \wedge cl(T_\alpha) \models \psi_{t+1}.$$

According to this definition, progression is really a two-step process: first make sure that the action is executable under the given condition about the initial state, and then compute the changes that the action will have under the given initial condition. We have shown that the first step is -hard in the worst case. The following proposition shows the complexity of the second step under various assumptions.

Proposition 3 *We have identified the following complexity results* ("CONP-c" *stands for CONP-complete).*

Complexity of PROCiRhSMON	*with causal rules*	*no causal rules*
any . any V'	CoNP-r	CoNP-r
0 *complete state*	CoNP-c	P
V' *literal*	CoNP-r	coNP-c
€ *complete stale* + € *literal*	CoNP-r	P

Proof:

1. Membership to CONP comes easily from the fact that $\langle T_\alpha, \phi, \psi \rangle$ is a positive instance of PROGRESSION if and only if $\phi_t \wedge cl(T_\alpha) \models \psi_{t+1}$ holds.
2. In the case where ϕ_t is a complete initial state and there are no causal rules, $cl(T_\alpha) = \{(l_i)_{t+1} \Leftrightarrow \chi_{i,t} \mid l_i \in L_{PS} \cup \{\bot\}\}$, where for each i, $Var(\chi_{i,t}) \subseteq PS_t$. When ϕ_t is a complete initial state, each $\chi_{i,t}$ can be evaluated in linear time to a truth value. This truth value is given to the corresponding $(l_i)_{t+1}$. Since α is executable under ϕ, this will not lead to a contradiction, and a complete successor state s_{t+1} is characterized by the values given to the literals $(l_i)_{t+1}$. It remains to check in linear time whether $s_{t+1} \models \psi_{t+1}$ to determine whether the instance is positive or not.
3. What remains to be done is showing CONP-hardncss in these two eases:

   (a) there are no eausal rules and ψ is a literal;

   (b) ϕ is a complete state and ψ is a literal;

   Case 1: no causal rules, ψ is a literal.
   The proof comes from the following polynomial reduction from DNF-VALIDITY. Let $\xi = D_1 \vee ... \vee D_m$ be any DNF formula and let $M(\xi)$ be the instance of PROGRESSION defined by

   - $T_\alpha = \langle Eff(\alpha), Causal \rangle$ where $Eff(\alpha) = \{D_1 \xrightarrow{\alpha} new, \ldots, D_m \xrightarrow{\alpha} new\}$ and $Causal = \emptyset$;
   - $\phi = \top$; $\psi = new$, where new is a new atom not mentioned in ξ.

   Clearly α is executable under $\top$, and ξ is valid iff A/(£) is a positive instance of PROGRESSION.

   Case 2: ϕ is a complete state, ψ is a literal
   The proof comes from the following polynomial reduction from DNF-VALIDITY. Let $\xi = D_1 \vee ... \vee D_m$ be any DNF formula, and let $Var(\xi) = \{x_1, ..., x_p\}$ be the set of variables appearing in ξ. We then define $M(\xi) = \langle T_\alpha, \phi, \psi \rangle$ where $T_\alpha = \langle Eff(\alpha), Causal \rangle$ is given by $Eff(\alpha) = \{\top \xrightarrow{\alpha} new'\}$ and

   $$Causal = \{ D_i \wedge new' \longrightarrow new \mid i = 1 \ldots m \} \\ \bigcup \{ x_i \longrightarrow x_i, \neg x_i \longrightarrow \neg x_i \mid i = 1 \ldots p \}$$

   and new, *new'* are new atoms (not appearing in ξ). Let $\phi = s$, .s being any complete state satisfying $\neg new \wedge \neg new'$, and ψ = *new.* It can be seen that a is executable in (under) .s, and ξ is valid iff $M(\xi)$ is a positive instance of PROGRESSION. ■

Our results about the complexity of executability and progression, taken together, are strongly related to a result in [Turner, 2002] (namely Theorem 8(i)) which says that one-stage conformant planning (without concurrency) is Π_2^p-complete. Indeed, checking that α is a valid plan in this context amounts to checking that α is executable in all possible initial states and that its progression satisfies the goal. By considering executability and progression separately, we see more clearly that Π_2^p-hardness in Turner's result is solely due to the hardness of executability.

A second, perhaps a more difficult way of seeing the progression problem, is the following: given some information about the current situation, compute all possible successor states. Formally, this looks like consequence finding: given a formula $\phi \in$ *PROPps,* compute the strongest successor state formula $\Psi \in$ *PROPps* such that $\phi_t \wedge cl(T_\alpha) \models \Psi_{t+1}$. Model-theoretically, this corresponds to finding all complete successor states s'_{t+1} such that there is a complete initial state s_t that satisfies ϕ_t for which $s_t \wedge cl(T_\alpha) \wedge s'_{t+1}$ is consistent. As it turns out, this formula Ψ_{t+1} is the strongest necessary condition of ϕ_t under $cl(T_\alpha)$ on $PROP_{PS_{t+1}}$ [Lin, 2001]:

Definition 6 ($Pro(T_\alpha, \phi)$) *Given an action theory T_α and a formula ϕ from $PROP_{PS}$ such that α is executable under ϕ, the* **progression formula** *$Pro(T_\alpha, \phi)$ is the formula Ψ from $PROP_{PS}$ (unique up to logical equivalence) such that $\langle T_\alpha, \phi, \psi \rangle$ is a positive instance of* **PROGRESSION** *if and only if $\Psi \models \psi$ holds.*

$Pro(T_\alpha, \phi)$ can be characterized equivalently by:

Proposition 4

1. *Let s be any complete state. We have $s \models Pro(T_\alpha, \phi)$ iff $\exists s'_t \in 2^{PS_t}$, $(((s'_t, s_{t+1}) \models cl(T_\alpha)$ and $s'_t \models \phi_t)$,* i.e., *there exists a complete initial state s'_t such that $s'_t \models \phi_t$ and $(s'_t, s_{t+1}) \models cl(T_\alpha)$ (or equivalently, $\phi_t \wedge cl(T_\alpha) \wedge s_{t+1}$ is consistent).*
2. $Pro(T_\alpha, \phi)_{t+1} \equiv \exists PS_t.(\phi_t \wedge cl(T_\alpha))$.

4.3 Regression

There are (at least) two possible definitions for *regression,* each of which corresponds to a given need: *deductive* regression (also referred to as *temporal explanation* or *weak preimage)* and *abductive regression* (also referred to as *strong preimage).*

Definition 7 (deductive regression) *Let T_α be an action theory and let ϕ, ψ be two formulas from PROPPS- $\langle T_\alpha, \phi, \psi\rangle$ is a positive instance of DEDUCTIVE* REGRESSION *(the deductive regression problem) if and only if*

$$\psi_{t+1} \wedge cl(T_\alpha) \models \phi_t.$$

Similar to progression formulas, we can define deductive regression formulas:

Definition 8 *(Regp(T_α, ψ)) Given an action theory T_α and a formula ψ from PROPps, the* deductive regression formula *Rcgo(T,i(j) is the formula Φ from PROPps (unique up to logical equivalence) such that $\langle T_\alpha, \phi, \psi\rangle$ is a positive instance* fl/DEDUCTIVE REGRESSION *ij and only if $\Phi \models \phi$.*

We immediately get that for any two formulas ϕ, ψ from *PROPps,* $\langle T_\alpha, \phi, \psi\rangle$ is a positive instance of DEDUCTIVE REGRESSION if and only if $\langle T_\alpha, \neg\phi, \neg\psi\rangle$ is a positive instance of PROGRESSION. We also have:

Proposition 5

1. *Let s be any complete state. We have $s \models Reg_D(T_\alpha, \psi)$ if and only if $\exists s'_{t+1} \in 2^{PS_{t+1}}$, $(((s_t, s'_{t+1}) \models cl(T_\alpha)$ and $s'_{t+1} \models \psi_{t+1})$,* i.e., *there exists a complete successor state s'_{t+1} such that $s'_{t+1} \models \psi_{t+1}$ and $(s_t, s'_{t+1}) \models cl(T_\alpha)$ (or equivalently, $\psi_{t+1} \wedge cl(T_\alpha) \wedge s_t$ is consistent).*
2. $Reg_D(T_\alpha, \psi)_t \equiv \exists PS_{t+1}.(\psi_{t+1} \wedge cl(T_\alpha))$.

This characterization helps understanding deductive regression: it intuitively means that we are interested in finding the set of states *which could be* possible *states before the action, knowing that ψ holds after the action.* This type of regression is useful for *postdiction,* i.e., reasoning about the past state of the system.

Since deductive regression is expressed as a deduction problem, its complexity is easy to find out.

Proposition 6 DEDUCTIVE REGRESSION *is* coNP-*complete.*

Definition 9 (abductive regression) *Let T_α be an action theory and let ϕ, ψ be two formulas from PROPps- $\langle T_\alpha, \phi, \psi\rangle$ is a positive instance* o/ABDUCTIVE REGRESSION *(the abductive regression problem) if and only if for any complete state s, $s_t \wedge cl(T_\alpha) \models \psi_{t+1}$ implies $s \models \phi$.*

The corresponding search problem is defined by:

Definition 10 *Given an action theory T_α and a formula ψ from PROPps, the* abductive regression formula *Reg (T_α, ψ) is the formula Φ (unique up to logical equivalence) such that $\langle T_\alpha, \phi, \psi\rangle$ is a positive instance* #/ ABDUCTIVE REGRESSION *if and only if $\phi \models \Phi$.*

In other words, Φ_t is the weakest sufficient condition of ψ_{t+1} under $cl(T_\alpha)$ *on PROPPS,* [Lin, 2001 J.

Proposition 7

1. *Let s be any complete state. $s \models Reg_A(T_\alpha, \psi)$ if and only if $\forall s'_{t+1} \in 2^{PS_{t+1}}$, if $(s_t, s'_{t+1}) \models cl(T_\alpha)$ then $s'_{t+1} \models \psi_{t+1}$ (or, equivalently, $s_t \wedge cl(T_\alpha) \models \psi_{t+1}$).*
2. $Reg_A(T_\alpha, \psi)_t \equiv \forall PS_{t+1}.(cl(T_\alpha) \Rightarrow \psi_{t+1})$.

Thus, abductive regression amounts to finding the set of all complete initial states s_t for which all possible complete successor states s'_{t+1} w.r.t. T_α satisfy ψ. This means that we are interested in finding the set of states *in which performing the action leads to states* necessarily *satisfying ψ - provided that the action can be performed in them.* This type of regression is useful for *planning,* i.e., reasoning about the minimal conditions under which an action succeeds in reaching the goals. Thus, $\langle T_\alpha, \phi, \psi\rangle$ is a positive instance of ABDUCTIVE REGRESSION if and only if ϕ implies the minimal conditions under which the action leads to the goal.

This shows that the qualifications "deductive'[1] and "abductive" are only related to the way initial states can be inferred, and not to the confidence we have in them to be satisfied. Indeed, as far as reasoning is concerned, deductive conclusions can be taken for sure since deduction is truth-preserving while abductive conclusions cannot be taken for sure in general since abduction is only falsity-preserving. Contrariwise to such a reasoning situation, initial states s_t obtained through deductive regression are only possible ones given ψ and α: it could be the case that performing action α in s leads to a successor state in which ψ does not hold. Contrastingly, initial states s_t obtained through abductive regression lead to successor states where ψ necessarily holds.

The following proposition makes precise the links between both forms of regression:

Proposition 8

1. *$Reg_A(T_\alpha, \psi) \models Reg_D(T_\alpha, \psi)$ if and only if α is fully executable.*
2. *$Reg_A(T_\alpha, \psi) \equiv Reg_D(T_\alpha, \psi)$ if and only if α is fully executable and deterministic.*

Proof: We give the proof of the first point:

$Reg_A(T_\alpha, \psi) \models Reg_D(T_\alpha, \psi)$
iff $\forall PS_{t+1}.(\neg cl(T_\alpha))$ is inconsistent
iff $\exists PS_t.(\forall PS_{t+1}.(\neg cl(T_\alpha)))$ is inconsistent
iff $\forall PS_t.(\exists PS_{t+1}.cl(T_\alpha))$ is valid
iff α is fully executable. ■

Abductive regression is computationally more expensive than deductive regression:

Proposition 9 ABDUCTIVE REGRESSION *is Π_2^p-complete.*

Proof. Membership is easy. Hardness comes from the following polynomial reduction from the Σ_2^p-complete problem QBF$_{2,\exists}$ to ABDUCTIVE REGRESSION. Let $Q = \langle\{a_1, \ldots, a_n\}, \{b_1, \ldots, b_p\}, \beta\rangle$ be an instance of QBF$_{2,\exists}$. Wlog, we assume that β is a DNF formula $D_1 \vee \ldots \vee D_m$. We now define $M(Q) = \langle T_\alpha, \phi, \psi\rangle$ where

- T_α is concerned by u e n t s built over $\{a_1, \dots, a_n, b_1, \dots, b_p, new\}$ (where *new* is a new atom, not appearing in β) and the following set of static causal rules:
$$\{ D_i \longrightarrow new \mid i = 1 \dots m \} \cup \{ b_i \longrightarrow b_i \mid i = 1 \dots p \} \cup \{ \neg b_i \longrightarrow \neg b_i \mid i = 1 \dots p \}$$
(and no effect rule);
- $\phi = new$;
- $\psi = new$.

Q is a positive instance of $QBF_{2,\exists}$ iff $M(Q)$ is a positive instance of ABDUCTIVE REGRESSION. ■

5 Generalized action theories

Here we want to extend causal and effect rules so as to allow for any possible consequents, *including disjunctions,* while keeping the framework simple enough.

Definition 11 (generalized action theory) *A* generalized action theory T_α *for action* α *is an ordered pair* $\langle Eff(\alpha)$, *Causal)* *where*

- $Eff(\alpha)$ *is a finite set of effect* rules *of the form pre* $\xrightarrow{\alpha}$ $post[L]$) *where pre, post are formulas from PBDPps and* $L \subseteq$ *LPS.*
- *Causal is a finite set of* causal rules *of the form left* $\longrightarrow$ *right[L] where left and right are formulas from PROPps and L* $\subseteq$ *L_{PS}.*

The completion consists now in writing down that every literal / persists if and only if there is no effect (action or causal) whose precondition is verified and on which / depends. More generally, rather than writing "there is no effect (action or causal) whose precondition is verified and on which / depends", we specify for each action rule the fluents *L* that are influenced and for which there should not be any frame axiom if the action rule is enabled.

Definition 12 (completion) *The* completion $cl(T_\alpha)$ *of generalized action theory* T_α *is the formula defined by*

- $T(\alpha) = \bigwedge\{pre_t \Rightarrow post_{t+1} \mid Eff(\alpha)$ *contains* $pre \xrightarrow{\alpha} post[L]\}$;
- $T(Causal) = \bigwedge\{left_t \Rightarrow right_t, left_{t+1} \Rightarrow right_{t+1} \mid Causal$ *contains* $left \longrightarrow right[L]\}$;
- *for each literal* $l \in L_{PS}$, *let*
$\Phi^\alpha(l) = \bigvee\{pre_t \mid Eff(\alpha)$ *contains pre* $\xrightarrow{\alpha} post[L]$ *and* $l \in L\}$,
$\Phi^C(l) = \bigvee\{left_{t+1} \mid Causal$ *contains* $left \longrightarrow right[L]$ *and* $l \in L\}$,
$\Phi(l) = \Phi^\alpha(l) \vee \Phi^C(l)$,
$frame(l) = (l_t \wedge \neg\Phi(\hat{l})) \Rightarrow l_{t+1}$;
- $cl(T_\alpha) = T(\alpha) \wedge T(Causal) \wedge \bigwedge_{l \in L_{PS}} frame(l)$.

As for simple action theories, we assume that each symbol of *PS* occurs in T_α so that the size of $cl(T_\alpha)$ is polynomial in the size of T_α.

In the previous definition, $\Phi^\alpha(l)$ gathers the preconditions of effect rules whose postcondition influences positively / (and similarly for $\Phi^C(l)$); the completion *frame{l)* means that if / is initially true and there is no active action rule nor any causal rule whose consequent part influences negatively /, then / persists after a is performed.

By default, the set *L* associated with an action rule (resp. a causal rule) is the set of literals *Lit(post)* (resp. *Lit{right))* positively mentioned in the negation normal form of *post* (resp. *of right)* - an alternative, more refined possibility being the set of literals *DepLit(post)* (resp. *DepLit(right))* on which *post* (resp. *right)* semantically depends [Lang and Marquis, 1998]. It is easy to show that the completion of T_α as defined above (Definition 12) is equivalent to the completion given at Definition 2 whenever T_α is simple (in that case, the default choice for *L* in each rule is considered, i.e., $L = \{l\}$ whenever the consequent part of the rule is *l).*

For instance, so to say that flipping a coin release *head,* we simply write *holdingCoin* $\xrightarrow{flip} true[head, \neg head]$, i.e., if one is holding the coin initially, then after the action of flipping it, we can neither infer *head* nor *-head* using inertia.

Due to space limitation, we cannot provide the full details here. In a nutshell, the main reasons why generalized action theories are interesting are twofold. On the one hand, we can prove that generalized action theories can be used to represent any nondeterministic action (associating a nonempty set of successor states to any initial state), while such a completeness property is not satisfied by simple action theories. On the other hand, the complexity of EXECUTABILITY, DETERMINISM, PROGRESSION, DEDUCTIVE/ABDUCTIVE REGRESSION, from generalized action theories coincide with the corresponding complexity result for simple action theories in the general case. This shows that the gain in expressiveness offered by generalized action theories is not balanced by a complexity increase for any of these reasoning tasks. Furthermore, this suggests that the simple action theories considered here constitute the computational core of causal action theories.

6 Other related work

iLiberatore, 1997] investigates the complexity of reasoning about action in the language *A* iGelfond and Lifschitz, 1993], therefore he considers only deterministic actions without static causal rules. He shows the CONP-completeness of the progression problem in language A, to be related to the right-upmost square in our Proposition 3. lEiter *et ai,* 2001] study the computational of many planning problems (including the progression problem as a particular case) using an action description language based on answer set semantics.

[Drakengren and Bjareland, 1997; 1999] investigate the complexity of checking the consistency of a scenario description in a temporal logic in the style of [Sandewall, 19941 which shares some similarities with causal theories of action, although both the syntax and the semantics are different. No static causal rules are considered (but on the other hand, the language allows for explicit time and concurrency). Checking the consistency of a scenario description is NP-complete and falls in P under specific syntactical restrictions.

Reasoning about action has strong connections with *belief update,* and therefore, the complexity study of belief update operators is relevant to our concern. The update operators whose complexity is studied in LEiter and Gottlob, 1992;

Herzig and Rifi, 1999; Liberatore, 2000] do not consider static causal rules, but allow for expressing any effects, especially disjunctions. The complexity of checking whether $\psi \in KB \diamond \varphi$ - where o is the update operator, mapping a logical theory and a formula to a logical theory - corresponds to the progression problem[2].

7 Conclusions and future work

The main contribution of this paper is the identification of complexity results corresponding to many reasoning tasks considered when dealing with causal action theories. It remains to be checked to what extent these results are changed when considering additional capabilities such as concurrency (as in iGiunchiglia and Lifschitz, 1998]) or resources.

Because of the way the progression problem is stated, it was expected that for arbitrary description φ of the initial situation and arbitrary query ψ about the successor situation, the problem is coNP-complete. But when we first started studying the problem, we were really expecting that when φ is a complete state, the problem would be easier; we were surprised that even in this case it turned out to be intractable. In retrospect, what happens is that even though both action effect rules and causal rules have very restricted form, the complete action theory $cl(T_\alpha)$ can be complex, and may not always be deterministic. An interesting question is then if we already know that the action theory is deterministic, would progression with complete initial state still be intractable? We do not know the answer at that stage.

Acknowledgements

Tha authors would like to thank the anonymous referees for helpful comments. The second authors work was supported in part by HK RGC CERG HKUST6182/0IE. The third author has been partly supported by the IUT de Lens, the Universite d'Artois, the Region Nord / Pas-de-Calais under the TACT-TIC project, and by the European Community FEDER Program.

References

[Baral, 1995] C. Baral. Reasoning about actions: nondeterministic effects, constraints, and qualification. In *IJCAI-95*, pages 2017-2023,1995.

[Drakengren and Bjareland, 1997] T. Drakengren and M. Bjareland. Reasoning about action in polynomial time. In *IJCAI-91,* pages 1447-1453,1997.

[Drakengren and Bjareland, 1999] T. Drakengren and M. Bjareland. Expressive reasoning about action in nondeterministic polynomial time. In I*JCA1-99,* pages 161-171,1999.

LEiter and Gottlob, 1992] T. Eiter and G. Gottlob. On the complexity of propositional knowledge base revision, updates and counterfactuals. *Artificial Intelligence,* 57:227-270, 1992.

[2]Note that the fact that the complexity of update lies most often at the second level of the polynomial hierarchy (therefore more complex than progression in action theories) is not due to the presence of disjunctions but is due mainly to the minimization-based semantics of the update operators considered.

LEiter *etai,* 2001] T. Eiter, W. Faber, N. Leone, G. Pfeifer, and A. Polleres. A logic programming approach to knowledge-state planning: Semantics and complexity. Technical report, Tech. University Wien, 2001.

iGelfond and Lifschitz, 1993] M. Gelfond and V. Lifschitz. Representing action and change by logic programs. *Journal of Logic Programming,* pages 301-322, 1993.

[Giunchiglia and Lifschitz, 1998] E. Giunchiglia and V. Lifschitz. An action language based on causal explanation: Preliminary report. In *AAA1-98,* pp. 623-630, 1998.

[Gustafsson and Doherty, 1996] J. Gustafsson and P. Doherty. Embracing occlusion in specifying the indirect effects of actions. In *KR-96,* pages 87-98, 1996.

[Herzig and Rifi, 1999] A. Herzig and O. Rifi. Propositional belief update and minimal change. *Artificial Intelligence,* 115:107-138,1999.

[Kakas *etai,* 2001] A. C. Kakas, R. Miller, and F Toni. E-RES: reasoning about actions, events and observations. In *LPNMR-0I,* pages 254-266, 2001.

[Lang and Marquis, 1998] J. Lang and P. Marquis. Complexity results for independence and definability in propositional logic. In *KR-98,* pages 356-367,1998.

iLiberatore, 1997] P. Liberatore. The complexity of the language A. *Electronic Transactions on Artificial Intelligence, \A3-3S, \991.*

ILiberatore, 2000] P. Liberatore. The complexity of belief update. *Artificial Intelligence,* 119:141-190,2000.

[Lifschitz, 1997] V. Lifschitz. On the logic of causal explanation. *Artificial Intelligence,* 96:451-465,1997.

[Lin, 1995] F Lin. Embracing causality in specifying the indirect effects of actions. In *UCAI-95,* pages 1985-1993, 1995.

[Lin, 2000] F. Lin. From causal theories to successor state axioms and STRIPS-like systems. In *AAAI-00,* pages 786-791,2000.

[Lin, 2001] F Lin. On the strongest necessary and weakest sufficient conditions. *Artificial Intelligence,* 128:143-159, 2001.

[McCain and Turner, 1997] N. McCain and H. Turner. Causal theories of action and change. In *AAAI-97,* pages 460-465,1997.

[McCain and Turner, 1998] N. McCain and H. Turner. Satisfiability planning with causal theories. In *KR-98,* pages 212-223,1998.

[Sandewall, 1994] E. Sandewall. *Features and Fluents.* Oxford University Press, 1994.

[Thielscher, 1995] M. Thielscher. Computing ramification by post-processing. In *UCAI-95,* pages 1994-2000,1995.

[Turner, 2002] H. Turner. Polynomial-length planning spans the polynomial hierarchy. In *JELIA-02,* pages 111-124, 2002.

[Zhang and Foo, 2001] D. Zhang and N. Foo. EPDL: a logic for causal reasoning. In *IJCAI-OI,* pages 131-138,2001.