

First-Order Strong Progression for Local-Effect Basic Action Theories

Stavros Vassos

Department of Computer Science
University of Toronto
Toronto, ON, M5S 3G4, Canada
stavros@cs.toronto.edu

Gerhard Lakemeyer

Department of Computer Science
RWTH Aachen
52056 Aachen, Germany
gerhard@cs.rwth-aachen.de

Hector J. Levesque

Department of Computer Science
University of Toronto
Toronto, ON, M5S 3G4, Canada
hector@cs.toronto.edu

Abstract

In a seminal paper Lin and Reiter introduced the notion of progression for basic action theories in the situation calculus. The idea is to replace an initial database by a new set of sentences which reflect the changes due to an action. Unfortunately, progression requires second-order logic in general. In this paper, we introduce the notion of strong progression, a slight variant of Lin and Reiter that has the intended properties, and we show that in case actions have only local effects, progression is always first-order representable. Moreover, for a restricted class of local-effect axioms we show how to construct a new database that is finite.

Introduction

In this paper, we are concerned with the problem of reasoning about the state of a world that changes as the result of named actions. For such applications, the basic reasoning problem is the so-called *projection problem*: given an action theory that specifies the preconditions and effects of actions, and an initial database (DB), determine whether or not a formula holds after a sequence of named actions is performed. Two settings where this problem arises naturally are for planning and for high-level program execution, *e.g.* Golog (Levesque *et al.* 1997).

In practice, there are two ways to deal with projection: we can *progress* the initial DB wrt the action sequence and answer the query against the resulting DB; or we can *regress* the query wrt the action sequence and answer the resulting query against the initial DB. Progression has at least two advantages: First, it avoids a duplication of effort when multiple queries need to be answered wrt the same action sequence, and especially when that sequence is long. Second, in a robotics setting, a robot can use its “mental idle time” to compute a progression while it is busy performing physical actions. Not surprisingly, virtually all of the implemented planners (based on STRIPS (Fikes & Nilsson 1971)) perform some form of progression, and many of the implemented agent programming languages such as Agent0 (Shoham 1993) and 3APL (Hindriks *et al.* 1999) do as well. It is typical of these implementations, however, to impose

severe restrictions on the form of the initial DB or on the specifications of the effects of actions. In STRIPS planning, for example, the initial DB is usually restricted to be a set of ground literals.

A formal and general definition of progression in the situation calculus was first proposed by Lin and Reiter (1997). They showed that their definition correctly captured simple cases of progression found in many implemented systems. However, they were also able to prove that there were cases where the progression of an initial DB formulated in first-order logic required second-order logic. The need for second-order logic was further justified by Vassos and Levesque who showed that the straightforward weaker definition of progression that is always first-order definable is not correct (2008).

In this paper, we return to the strong notion of progression defined by Lin and Reiter.¹ What we present here are two major results:

1. for a large class of action theories, called local-effect basic action theories, we prove that a first-order strong progression always exists;
2. for a slightly smaller class of action theories, called strictly local-effect basic action theories, we prove that a finite first-order strong progression can also be computed.

In the first case, the results hold without restriction on the form of the initial DB. The second only requires that objects names are unique. This is the first time to our knowledge that strong progression is shown to be first-order definable for an arbitrary first-order initial DB.

The rest of the paper is organized as follows: In the next section, we begin with a quick review of the situation calculus including the definition of basic action theories and our definition of strong progression. Next, we define local-effect basic action theories and present the first result. Next, we define strictly local-effect basic action theories and present the second result. Finally, we discuss some related work and draw some conclusions, including comments on how the restriction to local-effect can be lifted somewhat. Most proofs can be found in the appendix at the end of the paper.

¹As will be explained in the related work section, the formal definition we use is not quite identical to that of Lin and Reiter. It is, however, much simpler and cleaner.

Formal Preliminaries

The language $\mathcal{L}$ of the situation calculus (McCarthy & Hayes 1969) is first-order with equality and some limited second-order features. It is many-sorted, with sorts for actions, situations, and objects (everything else). A *situation* represents a world history as a sequence of actions. The constant S_0 is used to denote the initial situation where no actions have occurred. Sequences of actions are built using the function symbol do , such that $do(a, s)$ denotes the successor situation resulting from performing action a in situation s . We will typically use a to denote a variable of sort action and α to denote a term of sort action, and similarly s and σ for situations.

The language includes an infinite number of constant symbols of sort object that we will typically denote using letters c, d, e . We assume that $\mathcal{L}$ includes a finite number of *situation-independent* predicate symbols of sort object and a finite number of function symbols of sort action. All the situation-independent symbols take arguments of sort object. The language includes a finite number of *relational fluents*, that is predicates whose last argument is a situation, and thus whose value can change from situation to situation. For simplicity we do not allow $\mathcal{L}$ to include *functional fluents* and function symbols of sort object other than constants but note that with some extra effort the results presented in this paper can be proven to hold for the general case. Actions need not be executable in all situations, and the predicate $Poss(a, s)$ states that action a is executable in situation s . The language $\mathcal{L}$ also includes the binary predicate symbol $\sqsubseteq$ which provides an ordering on situations. The atom $s \sqsubseteq s'$ means that the action sequence s' can be obtained from the sequence s by performing one or more actions in s . We will use the notation $\sigma \sqsubseteq \sigma'$ as a macro for $\sigma \sqsubseteq \sigma' \vee \sigma = \sigma'$. Finally, the second-order functionality of $\mathcal{L}$ is limited to allowing (non-fluent) predicate variables that take arguments of sort object.

Often we need to restrict our attention to sentences in $\mathcal{L}$ that refer to a particular situation. For example, the initial database is a finite set of first-order sentences in $\mathcal{L}$ that do not mention any situation terms except for S_0 . For this purpose, for any situation term σ , we define the set of *uniform formulas in σ* to be all those (first-order or second-order) formulas in $\mathcal{L}$ that do not mention any other situation terms except for σ , do not mention $Poss$, and where σ is not used by any quantifier (Lin & Reiter 1997).

Basic action theories

We will be dealing with a specific kind of $\mathcal{L}$ -theory, the so-called basic action theories. The definition that follows is the same as in (Reiter 2001) except that, similarly to (Lake-meyer & Levesque 2004), $\mathcal{D}_{ap}$ consists of a single action precondition axiom for all actions instead of one separate axiom for each action symbol. A *basic action theory* (BAT) $\mathcal{D}$ over relational fluents $F_1, \dots, F_n$ has the following form:²

$$\mathcal{D} = \mathcal{D}_{ap} \cup \mathcal{D}_{ss} \cup \mathcal{D}_{una} \cup \mathcal{D}_0 \cup \mathcal{D}_{fnd}$$

²For the sake of readability we will be omitting the leading universal quantifiers.

1. $\mathcal{D}_{ap}$ contains a single precondition axiom for all actions of the form $Poss(a, s) \equiv \Pi(a, s)$, where $\Pi(a, s)$ is first-order and uniform in s .
2. $\mathcal{D}_{ss}$ is a set of successor state axioms (SSAs), one for each fluent symbol F_i , of the form $F_i(\vec{x}, do(a, s)) \equiv \Phi_i(\vec{x}, a, s)$, where $\Phi_i(\vec{x}, a, s)$ is first-order and uniform in s . SSAs characterize the conditions under which the fluent has a specific truth value at situation $do(a, s)$ as a function of situation s . We assume that $\Phi_i(\vec{x}, a, s)$ has the form
$$\gamma_i^+(\vec{x}, a, s) \vee (F_i(\vec{x}, s) \wedge \neg \gamma_i^-(\vec{x}, a, s)),$$
where $\gamma_i^+(\vec{x}, a, s)$ and $\gamma_i^-(\vec{x}, a, s)$ characterize the conditions under which $F_i(\vec{x}, do(a, s))$ becomes true or false respectively.
3. $\mathcal{D}_{una}$ is the set of unique-names axioms for actions: $A(\vec{x}) \neq A'(\vec{y})$, and $A(\vec{x}) = A'(\vec{y}) \supset \vec{x} = \vec{y}$, for each pair of distinct action symbols A and A' .
4. $\mathcal{D}_0$ is a set of first-order sentences uniform in S_0 . This is the initial DB that describes what holds in S_0 .
5. $\mathcal{D}_{fnd}$ is a set of domain independent foundational axioms which formally define legal situations and $\sqsubseteq$. This is the only part of $\mathcal{D}$ that includes a second-order sentence of $\mathcal{L}$.

Progression

We now proceed to formally define the notion of a strong progression. For variant definitions due to Lin and Reiter, see the related work section.

Let $\mathcal{D}$ be a basic action theory over relational fluents $F_1, \dots, F_n$, as above. Observe that $\mathcal{D}$ already tells us what is known about the situation that results from doing an action α in situation S_0 : $\mathcal{D}_0$ tells what is known about S_0 and the successor state axioms tell us how each fluent changes in going from S_0 to $do(\alpha, S_0)$. So in a sense,

$$\mathcal{D}_0 \wedge \bigwedge_{i=1}^n \forall \vec{x}. F_i(\vec{x}, do(\alpha, S_0)) \equiv \Phi_i(\vec{x}, \alpha, S_0)$$

is already a progression, except for the fact that it includes what is known about S_0 and so is not uniform in $do(\alpha, S_0)$. The progression we propose below removes the dependency on $F_i(\vec{x}, S_0)$ by using second-order quantification over predicates with no situation argument. The resulting sentence will then be uniform in $do(\alpha, S_0)$. More precisely, we have the following:

Definition 1. Let $F_1, \dots, F_n$ be relational fluents, and let $Q_1, \dots, Q_n$ be non-fluent predicate variables. For any formula ϕ in $\mathcal{L}$, let $\phi(\vec{F} : \vec{Q})$ be the formula that results from replacing any fluent atom $F_i(t_1, \dots, t_n, \sigma)$ in ϕ , where σ is a situation term, by $Q_i(t_1, \dots, t_n)$.

Definition 2. Let $\mathcal{D}$ be a basic action theory over fluents $\vec{F}$ as above, and let α be a ground action term. Then $Pro(\mathcal{D}, \alpha)$ is a second-order sentence uniform in $do(\alpha, S_0)$ defined by

$$\begin{aligned} &\exists \vec{Q}. \mathcal{D}_0(\vec{F} : \vec{Q}) \wedge \\ &\bigwedge_{i=1}^n \forall \vec{x}. F_i(\vec{x}, do(\alpha, S_0)) \equiv (\Phi_i(\vec{x}, \alpha, S_0)(\vec{F} : \vec{Q})). \end{aligned}$$

Definition 3. Let $\mathcal{D}$ be a basic action theory, α be a ground action term, and $\mathcal{D}_\alpha$ be a set of sentences uniform in $do(\alpha, S_0)$. $\mathcal{D}_\alpha$ is a *strong progression* of $\mathcal{D}$ wrt α iff $\mathcal{D}_\alpha$ is logically equivalent to $Pro(\mathcal{D}, \alpha)$.³

Although this definition of progression is formulated in second-order logic, like Lin and Reiter, we are concerned with progressions that are first-order. It is not hard to see that in simple cases that are used in practice (such as STRIPS planning), this definition does the right thing and remains within first-order logic. For example, suppose we have two unary fluents, F and G and a single action α that toggles F and leaves G unchanged. Then, if $\mathcal{D}_0$ is

$$F(c, S_0) \wedge G(d, S_0),$$

we get that a progression wrt α is

$$\neg F(c, do(\alpha, S_0)) \wedge G(d, do(\alpha, S_0)),$$

as expected. We also get intuitively plausible first-order results in more general cases. For example, if $\mathcal{D}_0$ is

$$(F(c, S_0) \vee \neg G(c, S_0)) \wedge \exists x \neg F(x, S_0),$$

then a progression wrt α is

$$(\neg F(c, do(\alpha, S_0)) \vee \neg G(c, do(\alpha, S_0))) \wedge \exists x F(x, do(\alpha, S_0)),$$

again as expected. However, as first shown by Lin and Reiter, there are cases of basic action theories where no first-order strong progression exists.

The important property of strong progression is that it is able to answer *unrestricted* queries about all the future situations that come after $do(\alpha, S_0)$ and get the correct answer wrt the original theory $\mathcal{D}$. For example, $Pro(\mathcal{D}, \alpha) \cup (\mathcal{D} - \mathcal{D}_0)$ entails the sentence

$$\exists x \forall s (do(\alpha, S_0) \sqsubseteq s \supset \neg F(x, s))$$

iff $\mathcal{D}$ entails it. Note that this sentence is not uniform in any situation term.

Local-effect BATs

In this paper we restrict our attention to BATs with *local effects* similarly to (Liu & Levesque 2005). In particular we consider BATs where $\mathcal{D}_0$ is unrestricted but the axioms in $\mathcal{D}_{ss}$ are such that if an action $A(\vec{e})$ changes the truth value of the fluent $F(\vec{c}, s)$ then $\vec{c}$ must be contained in $\vec{e}$. The formal definition of local-effect BATs follows.⁴

Definition 4. The successor state axiom for F_i is *local-effect* if both $\gamma_i^+(\vec{x}, a, s)$ and $\gamma_i^-(\vec{x}, a, s)$ are disjunctions of formulas of the form

$$\exists \vec{z} (a = A(\vec{y}) \wedge \phi(\vec{y}, s)),$$

³Note that this definition of progression produces a set of sentences uniform in $do(\alpha, S_0)$. To allow a further progression to take place with another action, it is a simple matter to construct a new basic action theory, with $\mathcal{D}_0$ replaced by a version of the progression that is made uniform in S_0 by simply replacing $do(\alpha, S_0)$ everywhere by S_0 .

⁴See the related work section below for a short discussion of some variant definitions.

where A is an action symbol, $\vec{y}$ contains $\vec{x}$, $\vec{z}$ corresponds to the remaining variables of $\vec{y}$, and $\phi(\vec{y}, s)$ (called a *context formula*) is a first-order formula uniform in s . A basic action theory $\mathcal{D}$ is local-effect if each of the SSAs in $\mathcal{D}_{ss}$ is local-effect.

The following will be our working example.

Example 1. Consider a BAT that represents the dynamics of *narde*, a board game similar to backgammon except that one is forbidden to put his checker at a position occupied by one's opponent's checker. Let $\mathcal{L}$ be a language that includes the fluent $board(x_1, x_2, s)$ which is intended to represent that the checker x_1 lies at position x_2 on the board, and the action $move(y_1, y_2, y_3)$ which represents the action of moving checker y_1 from position y_2 to position y_3 . Assume that there is an appropriate definition of *Poss* that characterizes whether the action $move(y_1, y_2, y_3)$ is executable in each situation. By the rules of *narde* it follows that moving a checker y_1 from position y_2 to y_3 only affects its own position, provided the action is executable. The SSA for *board* is as follows:

$$board(x_1, x_2, do(a, s)) \equiv \exists z (a = move(x_1, z, x_2)) \vee \neg(\exists z (a = move(x_1, x_2, z))) \wedge board(x_1, x_2, s).$$

Note that in this SSA for *board*, $\gamma^+(x_1, x_2, a, s)$ is the formula $\exists z (a = move(x_1, z, x_2))$, $\gamma^-(x_1, x_2, a, s)$ is the formula $\exists z (a = move(x_1, x_2, z))$, and that the SSA is local-effect according to the previous definition. The intuition is that for any ground action of the form $move(c, d_1, d_2)$ we can characterize the ground fluents that may be affected by the action. This will be made precise in the next section.

For local-effect BATs we will show that we can always find a first-order strong progression, and for a special case of the local-effect BATs we will show how to construct one that is finite. We now proceed to present the first result.

A first-order strong progression always exists for local-effect BATs

We first introduce some necessary notation and then we define $\mathcal{U}$, a subset of the first-order sentences uniform in s , whose interpretation remains unaffected when action α is performed in s . We then define the operator $\mathcal{T}(\mathcal{G}, \phi)$ that transforms a first-order sentence ϕ uniform in s while preserving logical equivalence. This operator will play a key role in proving the main result both of this section and the following one.

The set of unaffected sentences $\mathcal{U}$

Similarly to (Liu & Levesque 2005), the instantiation of a local-effect SSA on a ground action term can be simplified using the unique names axioms for actions, so that it explicitly lists a finite number of ground fluent atoms that may be affected by the action.

Lemma 1. Let α be the ground action term $A(\vec{e})$, where $\vec{e}$ is a vector of constants in $\mathcal{L}$ and suppose that the SSA for F_i is local-effect. Then $\gamma_i^+(\vec{x}, \alpha, s)$ is logically equivalent to a formula of the following form:

$$(\vec{x} = \vec{c}_1 \wedge \phi_1(s)) \vee \dots \vee (\vec{x} = \vec{c}_m \wedge \phi_m(s)),$$

where each of the $\vec{c}_1, \dots, \vec{c}_m$ is a distinct vector of constants contained in $\vec{c}$, and $\phi_1(s), \dots, \phi_m(s)$ are first-order and uniform in s . Similarly for $\gamma_i^-(\vec{x}, \alpha, s)$.

Example 2. Let α be the ground action $move(c, d_1, d_2)$. Then, wrt the SSA for *board*, $\gamma^+(x_1, x_2, \alpha, s)$ is logically equivalent to $x_1 = c \wedge x_2 = d_2$ and $\gamma^-(x_1, x_2, \alpha, s)$ is logically equivalent to $x_1 = c \wedge x_2 = d_1$. Note that our example is a simple one as the formulas γ^+ and γ^- for *board* have no context formulas.

Without loss of generality we assume that for a local-effect $\mathcal{D}$ and a ground action α the formulas $\gamma_i^+(\vec{x}, \alpha, s)$ and $\gamma_i^-(\vec{x}, \alpha, s)$ have this simplified form. Note also that all ground actions in $\mathcal{L}$ have the form $A(\vec{c})$ where A is some action function and $\vec{c}$ is some vector of constants in $\mathcal{L}$. The following definition introduces the notation that we will be using to specify the ground fluent atoms that may be affected by a ground action α .

Definition 5. Let $\mathcal{D}$ be a local-effect basic action theory, F_i a fluent symbol in $\mathcal{L}$, and α a ground action. We define the *argument set of F_i wrt α* , which we denote as $\mathcal{C}_i$, to be as follows:

$$\mathcal{C}_i = \{\vec{c} \mid \vec{x} = \vec{c} \text{ appears in } \gamma_i^+(\vec{x}, \alpha, s) \text{ or } \gamma_i^-(\vec{x}, \alpha, s)\}.$$

We define the *characteristic set of α* , which we denote as $\mathcal{G}$, to be the following set of atoms:

$$\mathcal{G} = \{F_i(\vec{c}, s) \mid \vec{c} \in \mathcal{C}_i \text{ for some } i\}.$$

Example 3. Let α be the ground action $move(c, d_1, d_2)$. Then the argument set of *board* wrt α is $\{\langle c, d_1 \rangle, \langle c, d_2 \rangle\}$. Assuming that *board* is the only fluent symbol in $\mathcal{L}$ then the characteristic set of α is the set $\mathcal{G} = \{board(c, d_1, s), board(c, d_2, s)\}$.

Note that the sets $\mathcal{C}_i$ and $\mathcal{G}$ are always finite. The intuition is that for any fluent atom $F_i(\vec{x}, s)$ such that $\vec{x}$ is not equal to some $\vec{c} \in \mathcal{C}_i$, it follows that its truth value remains the same after action α is performed.⁵ This is made precise in the following lemma:

Lemma 2. Let $\mathcal{D}$ be a local-effect BAT, α a ground action, $\mathcal{C}_i$ the argument set of F_i wrt α , M a model of $\mathcal{D}$, and μ a variable assignment. If $M, \mu \models \vec{x} \neq \vec{c}$ for all $\vec{c} \in \mathcal{C}_i$, then

$$M, \mu \models F_i(\vec{x}, do(\alpha, s)) \text{ iff } M, \mu \models F_i(\vec{x}, s).$$

Building on Lemma 2 we now define the set $\mathcal{U}$, a subset of the first-order formulas uniform in s whose satisfaction in a model remains unaffected after the action α is performed.

Definition 6. Let $\mathcal{D}$ be a local-effect BAT and α a ground action. We define $\mathcal{U}$ as the smallest set such that the following conditions hold:

- all first-order situation-independent atoms of $\mathcal{L}$ (including equality atoms) and their negation are in $\mathcal{U}$;

⁵Note also that since we have not assumed unique-names for objects there could be other fluent atoms that may be affected by α that are not in $\mathcal{G}$. This is not a problem because the set $\mathcal{G}$ and equality are sufficient to characterize all those fluent atoms.

- $\bigwedge_{j=1}^m \vec{x} \neq \vec{c}_j \wedge F_i(\vec{x}, s)$ and $\bigwedge_{j=1}^m \vec{x} \neq \vec{c}_j \wedge \neg F_i(\vec{x}, s)$ are in $\mathcal{U}$, where $\mathcal{C}_i = \{\vec{c}_1, \dots, \vec{c}_m\}$ is the argument set of F_i wrt α , and $\vec{x}$ is any vector of variables;
- $\mathcal{U}$ is closed under the logical operators $\wedge, \vee$, and universal and existential quantification over objects.

The following lemma states formally in which sense a formula ϕ in $\mathcal{U}$ remains unaffected wrt s and $do(\alpha, s)$.

Lemma 3. Let $\mathcal{D}$ be a local-effect BAT, α a ground action, M a model of $\mathcal{D}$, μ a variable assignment, and $\phi(s)$ a formula in $\mathcal{U}$. Then $M, \mu \models \phi(s)$ iff $M, \mu \models \phi(do(\alpha, s))$.

The operator $\mathcal{T}(\mathcal{G}, \phi)$

We now proceed to present the operator $\mathcal{T}(\mathcal{G}, \phi)$ that transforms a first-order formula ϕ uniform in s into a special form while preserving logical equivalence. $\mathcal{T}(\mathcal{G}, \phi)$ builds a disjunctive formula by splitting cases among the finitely many possible interpretations for the atoms in the characteristic set of α and simplifying ϕ for each case into one that remains unaffected by action α . The simplification step is performed by the operator $\mathcal{V}(\theta, \phi)$.

Let ϕ be a first-order formula uniform in s . Without loss of generality we assume that ϕ is in *negation normal form (NNF)* such that negation is only applied to atoms in ϕ . We also assume that every fluent atom in ϕ has the form $F(\vec{x}, s)$ where $\vec{x}$ is some vector of variables. These assumptions do not restrict ϕ as it is well-known that for every formula there is a logically equivalent one in NNF, and it is also straightforward that any atom of the form $F(\vec{t}, s)$, where $\vec{t}$ is any set of terms, can be replaced by a formula of the form $\exists \vec{x}(\vec{x} = \vec{t} \wedge F(\vec{x}, s))$ preserving logical equivalence.

The formal definition of a $\mathcal{G}$ -model and $\mathcal{V}(\theta, \phi)$ follows.

Definition 7. Let $\mathcal{D}$ be a local-effect BAT, α a ground action, $\mathcal{G}$ the characteristic set of α , and ϕ a first-order formula uniform in s . The sentence θ is a $\mathcal{G}$ -model iff it is a conjunction of literals such that for each $F(\vec{c}, s) \in \mathcal{G}$ there is exactly one occurrence of $F(\vec{c}, s)$ or $\neg F(\vec{c}, s)$ in θ , and all atoms in θ are in $\mathcal{G}$ and they appear in θ in lexicographical order. The operator $\mathcal{V}(\theta, \phi)$ is defined inductively as follows:

- if ϕ is a first-order situation-independent atom of $\mathcal{L}$ (including equality atoms) or the negation of one, then $\mathcal{V}(\theta, \phi) = \phi$;
- if ϕ is the positive literal $F_i(\vec{x}, s)$ then

$$\mathcal{V}(\theta, \phi) = \bigvee_{j=1}^m (\vec{x} = \vec{c}_j \wedge v_j) \vee \left(\bigwedge_{j=1}^m \vec{x} \neq \vec{c}_j \wedge F_i(\vec{x}, s) \right),$$

where $\mathcal{C}_i = \{\vec{c}_1, \dots, \vec{c}_m\}$ is the argument set of F_i wrt α , and

$$v_j = \begin{cases} \text{true}, & \text{if } F_i(\vec{c}_j, s) \text{ is a conjunct of } \theta; \\ \text{false}, & \text{otherwise;} \end{cases}$$

- if ϕ is the negative literal $\neg F_i(\vec{x}, s)$ then $\mathcal{V}(\theta, \phi)$ is the same as above except that $F_i(\vec{x}, s)$ and $F_i(\vec{c}_j, s)$ are replaced by the corresponding negative literals;
- for the case of $\phi \vee \psi$, $\phi \wedge \psi$, $\exists x\phi$, and $\forall x\phi$, the operator inductively goes into the sub-formulas until it reaches a literal, for example $\mathcal{V}(\theta, \exists x\phi) = \exists x\mathcal{V}(\theta, \phi)$.

The next lemma says that the formula obtained by $\mathcal{V}(\theta, \phi)$ is in $\mathcal{U}$, that is, it remains unaffected by the ground action α .

Lemma 4. *Let $\mathcal{D}$ be a local-effect BAT, α a ground action, $\mathcal{G}$ the characteristic set of α , θ a $\mathcal{G}$ -model, and ϕ a first-order formula uniform in s . Then $\mathcal{V}(\theta, \phi)$ is a formula in $\mathcal{U}$.*

Example 4. Let α and $\mathcal{G}$ be as in the previous example, and ϕ be the formula $\exists x_1 \exists x_2 \text{board}(x_1, x_2, s)$. Given a $\mathcal{G}$ -model θ we will use $\mathcal{V}(\theta, \phi)$ to simplify ϕ into a formula in $\mathcal{U}$. Let θ be the conjunction $\text{board}(c, d_1, s) \wedge \neg \text{board}(c, d_2, s)$. The operator $\mathcal{V}(\theta, \phi)$ simplifies ϕ according to θ by splitting cases for each of the fluent literals in ϕ : if $\text{board}(x_1, x_2, s)$ unifies with some atom in θ then it replaces $\text{board}(x_1, x_2, s)$ by its truth value in θ ; otherwise if it does not unify with any of the atoms in θ then it keeps $\text{board}(x_1, x_2, s)$ unaffected. $\mathcal{V}(\theta, \phi)$ is the following formula:

$$\begin{aligned} & \exists x_1 \exists x_2 (x_1 = c \wedge x_2 = d_1 \wedge \text{true} \vee \\ & \quad x_1 = c \wedge x_2 = d_2 \wedge \text{false} \vee \\ & (x_1 \neq c \vee (x_2 \neq d_1 \wedge x_2 \neq d_2)) \wedge \text{board}(x_1, x_2, s)). \end{aligned}$$

Observe that the sentence we obtain is in $\mathcal{U}$.

The operator $\mathcal{T}(\mathcal{G}, \phi)$ splits cases wrt all the (finitely many) possible $\mathcal{G}$ -models and uses the operator $\mathcal{V}$ to simplify the formula ϕ according to each one.

Definition 8. Let $\mathcal{D}$ be a local-effect BAT, α a ground action, $\mathcal{G}$ the characteristic set of α , $\Theta = \{\theta_1, \dots, \theta_l\}$ the set of all $\mathcal{G}$ -models, and $\phi(s)$ a first-order formula uniform in s . We define $\mathcal{T}(\mathcal{G}, \phi)$ as follows:

$$\mathcal{T}(\mathcal{G}, \phi) = \bigvee_{k=1}^l (\theta_k \wedge \mathcal{V}(\theta_k, \phi)).$$

As the following lemma states, the $\mathcal{T}(\mathcal{G}, \phi)$ operator preserves logical equivalence.

Lemma 5. *Let $\mathcal{D}$ be a local-effect BAT, α a ground action, $\mathcal{G}$ the characteristic set of α , and ϕ a first-order formula uniform in s . Let M be a structure of the language and μ be a variable assignment. Then $M, \mu \models \phi \equiv \mathcal{T}(\mathcal{G}, \phi)$.*

A strong progression based on the first-order entailments of $\mathcal{D}$

Before we state the main result of this section we need to prove one more thing. The following lemma relates the models of two first-order theories that have the same entailments wrt a set of first-order sentences Γ . The lemma shows that for every model of one theory we can always find a model of the other such that the two models satisfy the same set of sentences in Γ . Intuitively this might seem like an obvious fact but the proof is not straightforward and involves a non-constructive argument that makes use of the Compactness Theorem of first-order logic.

Lemma 6. *Let $\Sigma_1, \Sigma_2, \Gamma$ be sets of sentences in $\mathcal{L}$ such that the following conditions hold:*

1. Γ is closed under logical conjunction and negation;
2. for all sentences ϕ in Γ , $\Sigma_1 \models \phi$ iff $\Sigma_2 \models \phi$.

For all structures M , if M is a model of Σ_1 then there is a model M' of Σ_2 such that for all ϕ in Γ , $M \models \phi$ iff $M' \models \phi$.

We now state the main result of the section, namely that when $\mathcal{D}$ is local-effect we can always specify a first-order strong progression.

Theorem 1. *Let $\mathcal{D}$ be a local-effect BAT with a finite $\mathcal{D}_0$, α be a ground action, and $\mathcal{F}_\alpha$ be a set that is logically equivalent to the set*

$$\{\phi \mid \mathcal{D} \models \phi, \phi \text{ first-order and uniform in } do(\alpha, S_0)\}.$$

Then $\mathcal{F}_\alpha$ is a strong progression of $\mathcal{D}_0$ wrt α .

In order to prove this we need to show that $\mathcal{F}_\alpha$ is logically equivalent to $Pro(\mathcal{D}, \alpha)$. One direction, namely that $Pro(\mathcal{D}, \alpha) \models \mathcal{F}_\alpha$, is immediate from the following basic property of strong progression that follows from Theorem 1 and Theorem 2 of (Lin & Reiter 1997):

Lemma 7. *Let $\mathcal{D}$ be a BAT, α a ground action, and ϕ a sentence uniform in $do(\alpha, S_0)$. Then,*

$$\mathcal{D} \models \phi \text{ iff } \mathcal{D}_{una} \cup Pro(\mathcal{D}, \alpha) \models \phi.$$

The challenging direction is to show that $\mathcal{F}_\alpha \models Pro(\mathcal{D}, \alpha)$. The second-order sentence $Pro(\mathcal{D}, \alpha)$ expresses that there exists a configuration for the truth value of the fluents such that it is consistent with the initial DB and it also respects the SSAs for the transition from S_0 to $do(\alpha, S_0)$. Essentially we need to show that for every model M of $\mathcal{F}_\alpha$ there is a configuration for the truth value of the fluents that qualifies as a predecessor of the situation $do(\alpha, S_0)$. This configuration is expressed using the predicate variables $\bar{Q}$. Our proof method relies on Lemma 6 so that to specify a model M' of $\mathcal{D}$ such that M and M' satisfy the same set of first-order sentences uniform in $do(\alpha, S_0)$, and then use this model as a guideline for specifying an interpretation for $\bar{Q}$ in M . In order to relate M and M' wrt the interpretation of fluents in S_0 and $do(\alpha, S_0)$ the operator $\mathcal{T}(\mathcal{G}, \phi)$ is used, taking advantage of its structure and the fact that each of the simplified ϕ sub-formulas it consists of are in $\mathcal{U}$ and therefore their truth value in S_0 remains unchanged in $do(\alpha, S_0)$.

A finite first-order strong progression for strictly local-effect BATs

In the previous section we showed that a first-order strong progression always exists for local-effect BATs, but there were no guarantees that it is a workable one in practice as it may be infinite. We now turn our attention to a restriction of the local-effect BATs for which we can always specify a finite first-order strong progression. Like before, if an action $A(\vec{e})$ changes the truth value of the fluent $F(\vec{c}, s)$ then $\vec{c}$ is contained in $\vec{e}$, but we also insist that if the change depends on the fluent $G(\vec{d}, s)$ then $\vec{d}$ is also contained in $\vec{e}$. Moreover, we assume uniqueness of names for constants. Finally, for simplicity we assume for this section that $\mathcal{L}$ includes no situation-independent predicate symbols, but note that the general case can be handled either by extending the definitions that follow or by using relational fluents that do not change in order to encode these predicates.

Definition 9. The successor state axiom for F_i is *strictly local-effect* if both $\gamma_i^+(\vec{x}, a, s)$ and $\gamma_i^-(\vec{x}, a, s)$ are disjunctions of formulas of the form

$$\exists \vec{z} (a = A(\vec{y}) \wedge \phi(\vec{y}, s)),$$

where A is an action symbol, $\vec{y}$ contains $\vec{x}$, $\vec{z}$ is the remaining variables of $\vec{y}$, and the context formula $\phi(\vec{y}, s)$ is uniform in s and such that every fluent atom mentioned in $\phi(\vec{y}, s)$ has the form $F(\vec{w}, s)$ for some vector of variables $\vec{w}$ and $\vec{w}$ is included in $\vec{y}$. A basic action theory $\mathcal{D}$ is strictly local-effect if each of the SSAs in $\mathcal{D}_{ss}$ is strictly local-effect and $\mathcal{D}$ includes the set $\mathcal{E}$ that consists of the uniqueness of names axioms for constants.

The definition of a strictly local-effect SSA is exactly the same with that of a local-effect SSA, except for the restriction that the context formulas in the SSA also have a special form. With this restriction, given a ground action, a context formula ϕ can also be simplified so that it mentions only ground fluents. Before moving to the next example that illustrates this, observe that the context formula of a strictly local-effect SSA cannot quantify over the arguments of a fluent.

Example 5. Consider a BAT that represents the dynamics of a fantasy world in some role playing game. The agent is trapped inside a dungeon and in order to get out she has to unlock the main door. The main door can only be unlocked by putting a number of levers in the right position and then pressing a button. Let $\mathcal{L}$ be a language that includes the fluent $lever(x_1, s)$ which represents that x_1 is a lever, the fluent $state(x_1, x_2, s)$ which is intended to represent that lever x_1 is at state x_2 , fluent $unlocked(s)$ which represents that the main door is unlocked, and the action pb which represents the action of pressing the button. Assume that pressing the button will unlock the main door provided all the levers are in state up . The SSA of $unlocked(s)$ is as follows.

$$unlocked(do(a, s)) \equiv$$

$$a = pb \wedge \forall w (lever(w, s) \supset state(w, up, s)) \vee unlocked(s).$$

In this SSA for $unlocked$, $\gamma^-(a, s)$ is the empty disjunction and $\gamma^+(a, s)$ is the formula $a = pb \wedge \phi(s)$, where the context formula $\phi(s)$ is $\forall w (lever(w, s) \supset state(w, up, s))$. Here the vectors $\vec{x}, \vec{y}$, and $\vec{z}$ are empty according to Definition 4 and Definition 9. Observe that this SSA is local-effect but not strictly local-effect since $lever(w, s)$ is mentioned in $\phi(s)$ but w is not included in $\vec{y}$. Finally, note that in this case where the domain is intended to be finite, the SSA could be easily transformed into one that is strictly local-effect by replacing the quantified sub-formula by an equivalent one that explicitly lists the finitely many levers in the world. For example, assuming that there are only two levers in the world, objects c_1 and c_2 , the following is a strictly local-effect version of the SSA for $unlocked$:

$$unlocked(do(a, s)) \equiv$$

$$a = pb \wedge state(c_1, up, s) \wedge state(c_2, up, s) \vee unlocked(s).$$

The instantiation of a strictly local-effect SSA on a ground action can be simplified so that all fluent atoms that appear in γ_i^+ and γ_i^- have ground arguments except for the situation argument, and similarly to the local-effect BATs, a finite number of ground fluent atoms that may be affected by the action are explicitly listed.

Lemma 8. Let α be the ground action term $A(\vec{e})$, where $\vec{e}$ is a vector of constants in $\mathcal{L}$ and suppose that the SSA for F_i is

strictly local-effect. Then $\gamma_i^+(\vec{x}, \alpha, s)$ is logically equivalent to a formula of the following form:

$$(\vec{x} = \vec{c}_1 \wedge \phi_1(s)) \vee \dots \vee (\vec{x} = \vec{c}_m \wedge \phi_m(s)),$$

where each of the $\vec{c}_1, \dots, \vec{c}_m$ is a distinct vector of constants contained in $\vec{e}$, and $\phi_1(s), \dots, \phi_m(s)$ are uniform in s and such that all fluent atoms in them have ground arguments of sort object that are contained in $\vec{e}$. $\gamma_i^-(\vec{x}, \alpha, s)$ is logically equivalent to a similar formula.

Without loss of generality we assume that for a strictly local-effect $\mathcal{D}$ the formulas $\gamma_i^+(\vec{x}, \alpha, s)$ and $\gamma_i^-(\vec{x}, \alpha, s)$ have this simplified form.

Definition 10. Let $\mathcal{D}$ be a strictly local-effect BAT and α a ground action. We define the *context set* of α , which we denote as $\mathcal{J}$, to be the union of the characteristic set $\mathcal{G}$ of α and the set of all fluent atoms $F(\vec{c}, s)$ that appear in some $\gamma_i^+(\vec{x}, \alpha, s)$ or $\gamma_i^-(\vec{x}, \alpha, s)$. A $\mathcal{J}$ -model is defined the same way as the $\mathcal{G}$ -model of Definition 7.

Note that $\mathcal{J}$ is finite. The intuition behind our progression mechanism is that we can progress $\mathcal{D}_0$ by progressing each of the possible $\mathcal{J}$ -models accordingly.

Definition 11. Let $\mathcal{D}$ be a strictly local-effect BAT, α a ground action, $\mathcal{G}$ the characteristic set, $\mathcal{J}$ the context set of α , and $\theta(s)$ a $\mathcal{J}$ -model. We define the *progression* of θ wrt α to be the $\mathcal{J}$ -model θ^* such that:

- for each atom $F_i(\vec{c}, s)$ in $\mathcal{G}$, $F_i(\vec{c}, s)$ appears positive in θ^* if $\mathcal{D}_{una} \cup \mathcal{D}_0 \cup \mathcal{E} \cup \{\theta(S_0)\} \models \Phi_i(\vec{c}, \alpha, S_0)$ and negated otherwise;
- the rest of the atoms in $\mathcal{J}$ appear in θ^* in the same polarity as in θ .

Now we can present the second result of this paper.

Theorem 2. Let $\mathcal{D}$ be a strictly local-effect BAT with a finite $\mathcal{D}_0$, α be a ground action, and $\phi(S_0)$ be the conjunction of all sentences in $\mathcal{D}_0$. Let $\mathcal{J}$ be the context set of α and $\Theta = \{\theta_1, \dots, \theta_l\}$ be the set of all the $\mathcal{J}$ -models. Let $\psi(s)$ be the following sentence:

$$\bigvee_{k=1}^l \left(\theta_k^* \wedge \mathcal{V}(\theta_k, \phi(s)) \right),$$

where θ_k^* is the progression of θ_k wrt α . Then $\psi(do(\alpha, S_0))$ is a strong progression of $\mathcal{D}_0$ wrt α .

The proof of this theorem is based on the same ideas as the one for Theorem 1.

Related Work

The inspiration for our work, and the research most closely related to it, is that of Lin and Reiter (1997). They give a model-theoretic definition of progression and show that when $\mathcal{D}_0$ is finite, a progression is always representable using a second-order formula very similar to ours. We will not present their definition here,⁶ but simply use the syntactic representation instead, which we call LR-progression:

⁶The definition by Lin and Reiter, which uses models explicitly, is considerably more complex than ours, but has the advantage that it also applies when the initial database $\mathcal{D}_0$ is infinite.

Definition 12 (LR-Progression).

Let $\mathcal{D}$ be a basic action theory over fluents $\vec{F}$, and let α be a ground action term. An LR-progression is any set of sentences uniform in $do(\alpha, S_0)$ that is logically equivalent to $\mathcal{D}_{una}$ conjoined with the following sentence:

$$\exists \vec{Q}. \mathcal{D}_0 \langle \vec{F} : \vec{Q} \rangle \wedge [\Pi(\alpha, S_0) \langle \vec{F} : \vec{Q} \rangle \supset \bigwedge_{i=1}^n \forall \vec{x}. F_i(\vec{x}, do(\alpha, S_0)) \equiv (\Phi_i(\vec{x}, \alpha, S_0) \langle \vec{F} : \vec{Q} \rangle)].$$

Note the close correspondence with our definition of strong progression. In fact, there are only two differences:

1. LR-progression includes $\mathcal{D}_{una}$ while we leave it out. This is really just a cosmetic difference, as we reinsert $\mathcal{D}_{una}$ as needed, for example, in Lemma 7.
2. LR-progression includes $\Pi(\alpha, S_0) \langle \vec{F} : \vec{Q} \rangle$, the precondition axiom for α , which is missing from $Pro(\mathcal{D}, \alpha)$. This has to do with an older version of successor state axioms used by Lin and Reiter. These all had the form $Poss(a, s) \supset F_i(\vec{x}, do(a, s)) \equiv \Phi_i(\vec{x}, a, s)$. Hence it was necessary to replace $Poss(a, s)$ by its definition in the progression. In cases where $\mathcal{D}_0 \models Poss(\alpha, S_0)$, it is easy to see that $Pro(\mathcal{D}, \alpha)$ is actually logically equivalent to the above sentence.

Reiter subsequently updated the definition of progression in his (2001) to conform to the new form of successor state axioms. This new version omits the $\Pi(\alpha, S_0) \langle \vec{F} : \vec{Q} \rangle$ and so is quite close to ours. Unfortunately, that new definition has the problem that in certain cases, unintended progressions arise. For example, when $\mathcal{D}_0$ is empty, then $\mathcal{D}_0$ itself always counts as a legal progression for trivial reasons. So with this definition, a BAT can have two progressions that are not logically equivalent, counter to the claim by Reiter (2001, Theorem 9.1.2). The same is true for the definition of progression used in (Claßen *et al.* 2007).

Other work on progression has looked at various ways of staying within first-order logic. Liu and Levesque (2005) proposed a much weaker version of progression that remained first-order definable and computationally tractable. Vassos and Levesque (2007) considered a less weak progression but relied on the assumption that there is a finite domain and a restricted form of disjunctive knowledge in the initial DB in order to remain first-order and tractable. Vassos and Levesque (2008) investigated the properties of another weaker version of progression first proposed by Lin and Reiter and which is always first-order definable, namely to let the set $\mathcal{F}_\alpha$ of Theorem 1 be the new initial DB. They proved that this form of progression is useful for answering a wide range of queries that go beyond those uniform in some situation term, but cannot be used to construct a new initial DB without losing information. Our Theorem 1 shows that for local-effect BATs, $\mathcal{F}_\alpha$ does not lose information. Shirazi and Amir (2005) proposed logical filtering as a way to progress the initial DB. They do not investigate the conditions under which progression is first-order definable but prove that when it is, logical filtering is correct for answering queries uniform in some situation term. Outside of

the situation calculus, Thielscher defined a dual representation for basic action theories based on state update axioms that explicitly define the direct effects of each action (1999).

In terms of local effect basic action theories, a slightly more restrictive definition was first presented by Liu and Levesque (2005). The difference is that they required the context formula to be quantifier-free, a constraint not needed here. A less syntactic version of the constraint also appeared unnamed in an earlier paper by Lin (2004). For his purposes (unrelated to progression), he required that for any fluent F and action symbol A , the following should be an entailment of the action theory:

$$\neg subset(\vec{x}, \vec{y}) \supset F(\vec{x}, do(A(\vec{y}), s)) \equiv F(\vec{x}, s)$$

where $subset(\vec{x}, \vec{y})$, meaning $\vec{x}$ is a subset of $\vec{y}$, is an abbreviation for the following formula:

$$\bigwedge_{x \in \vec{x}} \bigvee_{y \in \vec{y}} x = y.$$

Note that our syntactic constraint guarantees these entailments. It remains to be seen whether having these entailments would be sufficient to ensure the existence of first-order strong progressions.

Recall that a first-order strong progression of an initial database is a first-order sentence which is equivalent to the second-order sentence of Definition 2. In a sense this is a special case of what is called *quantifier elimination*, which refers to techniques to transform second-order sentences into first-order equivalents. Quantifier elimination was first studied by Ackermann (1935), and a survey of more recent work on this topic can be found in (Nonnengart, Ohlbach, & Szalas 1999). In (Doherty, Lukasiewicz, & Szalas 2001), the authors apply quantifier elimination to a number of issues related to progression, including *forgetting*. One limitation of this line of work is that completeness results for finding first-order equivalents exist only for certain classes of formulas. In contrast, we consider arbitrary initial databases, and the existence of a first-order strong progression depends crucially on the form of the successor state axioms. It seems unlikely that general quantifier elimination techniques would return useful results in such cases.

Conclusion

In this paper we presented two results about the progression of basic action theories:

1. for local-effect basic action theories, we showed that a first-order strong progression always exists;
2. for strictly local-effect basic action theories, we showed that a finite first-order strong progression can also be computed.

This is the first time to our knowledge that strong progression is shown to be first-order definable for an arbitrary first-order $\mathcal{D}_0$. The first result is general in that we impose no restrictions at all on the initial database $\mathcal{D}_0$. The second result is slightly less general as it requires that objects have unique names. For both results we do, however, require the action theories to have local effects, which rules out actions like

exploding a bomb, where objects unrelated to the arguments of the action can be affected by the action.

In future work, we intend to generalize the results presented here to even less restrictive action theories. We believe that the real difficulty with progression is not when an action can affect objects other than its arguments, but rather when we cannot bound in advance the objects that can be affected by the action. We hope to define a larger class of bounded action theories for which it will be possible to construct a first-order strong progression using a technique similar to the one presented here.

References

- Ackermann, W. 1935. Untersuchung über das eliminationsproblem der mathematischen logik. *Mathematische Annalen* 110:390–413.
- Claßen, J.; Eyerich, P.; Lakemeyer, G.; and Nebel, B. 2007. Towards an integration of Golog and planning. In Veloso, M. M., ed., *Proc. of IJCAI-07*, 1846–1851.
- Doherty, P.; Łukaszewicz, W.; and Szałas, A. 2001. Computing strongest necessary and weakest sufficient conditions of first-order formulas. In *Proc. of IJCAI-01*, 145–151.
- Fikes, R., and Nilsson, N. 1971. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence* 2:189–208.
- Hindriks, K. V.; De Boer, F. S.; Van der Hoek, W.; and Meyer, J.-J. C. 1999. Agent programming in 3APL. *Autonomous Agents and Multi-Agent Systems* 2(4):357–401.
- Lakemeyer, G., and Levesque, H. J. 2004. Situations, si! situation terms, no! In Dubois, D.; Welty, C. A.; and Williams, M.-A., eds., *Proc. of KR-04*, 516–526.
- Levesque, H. J.; Reiter, R.; Lespérance, Y.; Lin, F.; and Scherl, R. B. 1997. Golog: A logic programming language for dynamic domains. *Journal of Logic Programming* 31(1-3):59–83.
- Lin, F., and Reiter, R. 1997. How to progress a database. *Artificial Intelligence* 92(1-2):131–167.
- Lin, F. 2004. Discovering state invariants. In Dubois, D.; Welty, C. A.; and Williams, M.-A., eds., *Proc. of KR-04*, 536–544. AAAI Press.
- Liu, Y., and Levesque, H. J. 2005. Tractable reasoning with incomplete first-order knowledge in dynamic systems with context-dependent actions. In Veloso, M. M., and Kambhampati, S., eds., *Proc. of IJCAI-05*, 639–644.
- McCarthy, J., and Hayes, P. J. 1969. Some philosophical problems from the standpoint of artificial intelligence. *Machine Intelligence* 4:463–502.
- Nonnengart, A.; Ohlbach, H. J.; and Szałas, A. 1999. Elimination of predicate quantifiers. In Ohlbach, H. J., and Reyle, U., eds., *Logic, Language and Reasoning. Essays in Honor of Dov Gabbay*. Kluwer. 159–181.
- Reiter, R. 2001. *Knowledge in Action. Logical Foundations for Specifying and Implementing Dynamical Systems*. MIT Press.
- Shirazi, A., and Amir, E. 2005. First-order logical filtering. In Veloso, M. M., and Kambhampati, S., eds., *Proc. of IJCAI-05*, 589–595.
- Shoham, Y. 1993. Agent-oriented programming. *Artificial Intelligence* 60(1):51–92.
- Thielscher, M. 1999. From situation calculus to fluent calculus: State update axioms as a solution to the inferential frame problem. *Artificial Intelligence* 111(1-2):277–299.
- Vassos, S., and Levesque, H. 2007. Progression of situation calculus action theories with incomplete information. In Veloso, M. M., ed., *Proc. of IJCAI-07*, 2024–2029.
- Vassos, S., and Levesque, H. 2008. On the progression of situation calculus basic action theories: Resolving a 10-year-old conjecture. In *Proc. AAAI-08*. (To appear).

Appendix: Proofs of lemmas and theorems

Lemma 1 Let α be the ground action term $A(\vec{e})$, where $\vec{e}$ is a vector of constants in $\mathcal{L}$ and suppose that the SSA for F_i is local-effect. Then $\gamma_i^+(\vec{x}, \alpha, s)$ is logically equivalent to a formula of the following form:

$$(\vec{x} = \vec{c}_1 \wedge \phi_1(s)) \vee \dots \vee (\vec{x} = \vec{c}_m \wedge \phi_m(s)),$$

where each of the $\vec{c}_1, \dots, \vec{c}_m$ is a distinct vector of constants contained in $\vec{e}$, and $\phi_1(s), \dots, \phi_m(s)$ are first-order and uniform in s . Similarly for $\gamma_i^-(\vec{x}, \alpha, s)$.

Proof Sketch: Let $\exists \vec{z}(a = A(\vec{y}) \wedge \phi(\vec{y}))$ be a disjunct of $\gamma_i^+(\vec{x}, \alpha, s)$. Recall that $\vec{y}$ contains $\vec{x}$ and $\vec{z}$ is the remaining variables of $\vec{y}$. By the uniqueness of names for actions it follows that the disjunct is equivalent to $\vec{x} = \vec{c} \wedge \phi(\vec{c})$, where $\vec{x} = \vec{c}$ is contained in $\vec{y} = \vec{c}$. ■

Lemma 2 Let $\mathcal{D}$ be a local-effect BAT, α a ground action, $\mathcal{C}_i$ the argument set of F_i wrt α , M a model of $\mathcal{D}$, and μ a variable assignment. If $M, \mu \models \vec{x} \neq \vec{c}$ for all $\vec{c} \in \mathcal{C}_i$, then

$$M, \mu \models F_i(\vec{x}, do(\alpha, s)) \text{ iff } M, \mu \models F_i(\vec{x}, s).$$

Proof Sketch: Observe the definition of $\mathcal{C}_i$ and the simplified version of $\gamma_i^+(\vec{x}, \alpha, s)$ and $\gamma_i^-(\vec{x}, \alpha, s)$. Note that M may satisfy any of the γ_i formulas only if $\mu(\vec{x})$ is the same as the denotation of some $\vec{c} \in \mathcal{C}_i$ wrt equality. If this is not the case then both γ_i^+ and γ_i^- are false and the SSA for F_i reduces to $F_i(\vec{x}, do(\alpha, s)) \equiv F_i(\vec{x}, s)$. ■

Lemma 3 Let $\mathcal{D}$ be a local-effect BAT, α a ground action, M a model of $\mathcal{D}$, μ a variable assignment, $\mathcal{G}$ the characteristic set of α , and $\phi(s)$ a formula in $\mathcal{U}$. Then $M, \mu \models \phi(s)$ iff $M, \mu \models \phi(do(\alpha, s))$.

Proof. By induction on the construction of formulas in $\mathcal{U}$. We only show the base case since the induction step is straightforward. When $\phi(s)$ does not mention a fluent atom then $\phi(s)$ and $\phi(do(\alpha, s))$ coincide and the lemma follows. When $\phi(s)$ mentions a fluent atom then by the definition of the set $\mathcal{U}$ it follows that $\phi(s)$ is a conjunction of an equality sub-formula of the form $\bigwedge_{j=1}^m \vec{x} \neq \vec{c}_j$ and a fluent literal of the form $F(\vec{x}, s)$ or $\neg F(\vec{x}, s)$, where $\mathcal{C}_i = \{c_1, \dots, c_m\}$ is the argument set of F_i wrt α . We consider two cases. Case i): $M, \mu \models \vec{x} \neq \vec{c}$ for all $\vec{c} \in \mathcal{C}_i$. Then the above equality sub-formula holds. By Lemma 2 it follows that $M, \mu \models \phi(s)$ iff $M, \mu \models \phi(do(\alpha, s))$ and thus the lemma follows. Case ii): $M, \mu \models \vec{x} = \vec{c}$ for some $\vec{c} \in \mathcal{C}_i$. Then the equality sub-formula does not hold and the lemma follows. ■

Lemma 4 Let $\mathcal{D}$ be a local-effect BAT, α a ground action, $\mathcal{G}$ the characteristic set of α , θ a $\mathcal{G}$ -model, and ϕ a first-order formula uniform in s . Then $\mathcal{V}(\theta, \phi)$ is a formula in $\mathcal{U}$.

Proof. By induction on the construction of first-order uniform formulas in s . The base case is immediate from Definition 7 and the induction step is straightforward. ■

Lemma 5 Let $\mathcal{D}$ be a local-effect BAT, α a ground action, $\mathcal{G}$ the characteristic set of α , and ϕ a first-order formula uniform in s . Let M be a structure of the language and μ be a variable assignment. Then $M, \mu \models \phi \equiv \mathcal{T}(\mathcal{G}, \phi)$.

Proof. By induction on the construction of first-order uniform formulas in s . We only show the base case since the induction step is straightforward. Let $\Theta = \{\theta_1, \dots, \theta_l\}$ be the set of all $\mathcal{G}$ -models. When ϕ does not mention a fluent atom then $\mathcal{T}(\mathcal{G}, \phi)$ is $\bigvee_{k=1}^l (\theta_k \wedge \phi)$ which is logically equivalent to $(\bigvee_{k=1}^l \theta_k) \wedge \phi$. By the definition of a $\mathcal{G}$ -model and Θ it follows that $\bigvee_{k=1}^l \theta_k$ is valid and therefore the lemma follows. When ϕ is a fluent literal then it has the form $F_i(\vec{x}, s)$ or $\neg F_i(\vec{x}, s)$. We will show the proof for the positive literal and the other case is similar. First note that by the definition of a $\mathcal{G}$ -model and Θ it follows that there is exactly one $\theta_k \in \Theta$ such that $M, \mu \models \theta_k$. Therefore in order to prove the lemma it suffices to show that $M, \mu \models \mathcal{T}(\mathcal{G}, \phi)$ iff $M, \mu \models \mathcal{V}(\theta_k, \phi)$, where $M, \mu \models \theta_k$. Let $\mathcal{C}_i = \{\vec{c}_1, \dots, \vec{c}_m\}$ be the argument set of F_i wrt α . Recall that

$$\mathcal{V}(\theta, \phi) = \bigvee_{j=1}^m (\vec{x} = \vec{c}_j \wedge v_j) \vee \left(\bigwedge_{j=1}^m \vec{x} \neq \vec{c}_j \wedge F_i(\vec{x}, \sigma) \right),$$

where v_j depends on θ_k as in Definition 7.

For the ($\Rightarrow$) direction let $M, \mu \models F_i(\vec{x}, \sigma)$. We take two cases. Case i): for all j , $\mu(\vec{x}) \neq \vec{c}_j$. Then $M, \mu \models \bigwedge_{j=1}^m \vec{x} \neq \vec{c}_j \wedge F_i(\vec{x}, \sigma)$ and so $M, \mu \models \mathcal{V}(\theta_k, \phi)$. Case ii): there is a j such that $\mu(\vec{x}) = \vec{c}_j$. Since $M, \mu \models F_i(\vec{x}, s)$ it follows that $M, \mu \models F_i(\vec{c}_j, s)$. By Definition 7 and since $M, \mu \models \theta_k$ it follows that $F_i(\vec{c}_j, s)$ is a conjunct of θ_k and so v_j is true. It follows that $M, \mu \models \vec{x} = \vec{c}_j \wedge v_j$ and so $M, \mu \models \mathcal{V}(\theta_k, \phi)$. The ($\Leftarrow$) direction is similar. ■

Lemma 6 Let $\Sigma_1, \Sigma_2, \Gamma$ be sets of sentences in $\mathcal{L}$ such that the following conditions hold:

1. Γ is closed under logical conjunction and negation;
2. for all sentences ϕ in Γ , $\Sigma_1 \models \phi$ iff $\Sigma_2 \models \phi$.

For all structures M , if M is a model of Σ_1 then there is a model M' of Σ_2 such that for all ϕ in Γ , $M \models \phi$ iff $M' \models \phi$.

Proof. We will do a proof by contradiction as follows. Suppose otherwise. Then there is a model M_c of Σ_1 such that there is no model M' of Σ_2 so that for all sentences ϕ in Γ , $M_c \models \phi$ iff $M' \models \phi$. Equivalently, there is a model M_c of Σ_1 such that for every model M' of Σ_2 there is some sentence ψ in Γ so that $M_c \models \psi$ but $M' \not\models \psi$.⁷ Let Δ be the following set: $\{\psi : \psi \in \Gamma \text{ and there is a model } M' \text{ of } \Sigma_2 \text{ such that } M_c \models \psi \text{ but } M' \not\models \psi\}$.

Clearly Δ is consistent as $M_c \models \Delta$. Moreover there is no model M' of Σ_2 that satisfies Δ since for each model M' there is at least one sentence ψ in Δ such that $M' \not\models \psi$. Therefore $\Sigma_2 \cup \Delta \models \square$. By the Compactness Theorem for first-order logic we get that there is a finite subset of Δ , Δ' , such that $\Sigma_2 \cup \Delta' \models \square$. Let γ be the conjunction of all sentences in Δ' . Then $\Sigma_2 \cup \{\gamma\} \models \square$ which implies that $\Sigma_2 \models \neg\gamma$.

Since Γ is closed under logical conjunction and negation, $\gamma \in \Gamma$ as it is a conjunction of sentences in Γ and also $\neg\gamma \in \Gamma$. By point 2 in the hypothesis Σ_1 and Σ_2 entail the same

⁷Note that if instead of this there is a sentence ψ' in Γ so that $M_c \not\models \psi'$ but $M' \models \psi'$ we can just take ψ to be $\neg\psi'$.

set of sentences in Γ therefore we also get that $\Sigma_1 \models \neg\gamma$. Note that this yields a contradiction because M_c is a model of Σ_1 and M_c satisfies γ . ■

Theorem 1 Let $\mathcal{D}$ be a local-effect BAT with a finite $\mathcal{D}_0$, α be a ground action, and $\mathcal{F}_\alpha$ be a set that is logically equivalent to the set

$$\{\phi \mid \mathcal{D} \models \phi, \phi \text{ first-order and uniform in } do(\alpha, S_0)\}.$$

Then $\mathcal{F}_\alpha$ is a strong progression of $\mathcal{D}_0$ wrt α .

Proof. We need to show that $\mathcal{F}_\alpha$ is logically equivalent to $Pro(\mathcal{D}, \alpha)$. By Lemma 7 it follows that $Pro(\mathcal{D}, \alpha) \models \mathcal{F}_\alpha$. For the other direction we will do as follows. For simplicity assume that $\mathcal{D}_0$ is empty and let M be an arbitrary model of $\mathcal{F}_\alpha$ (we will lift this restriction later). We will specify an interpretation R_i for each predicate variable Q_i by considering two cases. Let $\vec{o}$ be a vector of objects in the domain of M of the same arity as $\vec{x}$, μ be a variable assignment, and $\mathcal{C}_i = \{c_1, \dots, c_m\}$ be the argument set of F_i wrt α .

- $M, \mu_{\vec{o}} \models \bigwedge_{j=1}^m \vec{x} \neq \vec{c}_j$. We define $R_i(\vec{o})$ to be true iff $M, \mu_{\vec{o}} \models F_i(\vec{x}, do(\alpha, S_0))$. The intuition is that these $\vec{o}$ correspond to fluent atoms that remain unchanged in the transition from S_0 to $do(\alpha, S_0)$.
- $M, \mu_{\vec{o}} \models \vec{x} = \vec{c}_j$ for some j . For this case we will use a model of $\mathcal{D}$ to guide us. In particular we will use a model M' of $\mathcal{D}$ that for every first-order sentence ϕ uniform in $do(\alpha, S_0)$,

$$M \models \phi \text{ iff } M' \models \phi. \quad (1)$$

Lemma 6 guarantees that we can always find such an M' .⁸ We define $R_i(\vec{o})$ to be true iff $M' \models F_i(\vec{c}_j, S_0)$. The intuition is that these $\vec{o}$ correspond to fluent atoms that may have changed in the transition from S_0 to $do(\alpha, S_0)$, and we will make $R_i(\vec{o})$ follow the interpretation of the fluent in M' .

Before we continue note that using a similar argument as in Lemma 3 it is easy to show that for all $\phi(s)$ in $\mathcal{U}$,

$$M \models \phi(do(\alpha, S_0)) \text{ iff } M \models \phi(\vec{F} : \vec{Q}). \quad (2)$$

Now we proceed to show that the relations R_i we have specified qualify as a predecessor of $do(\alpha, S_0)$. Since we assumed that $\mathcal{D}_0$ is empty we only need to show that for each of the F_i ,

$$M, \mu \models \forall \vec{x} (F_i(\vec{x}, do(\alpha, s)) \equiv \Phi_i(\vec{x}, \alpha, s) \langle \vec{F} : \vec{Q} \rangle),$$

where μ interprets s as S_0 and each of the predicate variables Q_i as the relation R_i . We will take the same cases again as follows. Let $\vec{o}$ and $\mathcal{C}_i$ be as above, and μ be a variable assignment that interprets s as S_0 and each of the predicate variables Q_i as the relation R_i .

⁸(1) follows from the foundational result in (Lin & Reiter 1997) that $\mathcal{D} - \mathcal{D}_{fnd}$ is equivalent to $\mathcal{D}$ wrt the entailment of first-order sentences uniform in σ , where here σ is $do(\alpha, S_0)$, and the application of Lemma 6, where Σ_1 is $\mathcal{F}_\alpha$, Σ_2 is $\mathcal{D} - \mathcal{D}_{fnd}$, and Γ is the set of first-order sentences uniform in $do(\alpha, S_0)$.

- $M, \mu_{\vec{o}} \models \bigwedge_{j=1}^m \vec{x} \neq \vec{c}_j$. By Lemma 1 it follows that $\Phi_i(\vec{x}, \alpha, s)$ reduces to $F_i(\vec{x}, s)$, and so $\Phi_i(\vec{x}, \alpha, s) \langle \vec{F} : \vec{Q} \rangle$ reduces to $Q_i(\vec{x})$. Therefore it suffices to show that $M, \mu_{\vec{o}} \models F_i(\vec{x}, do(\alpha, s)) \equiv Q_i(\vec{x})$ which follows from the way we defined R_i .
- $M, \mu_{\vec{o}} \models \vec{x} = \vec{c}_j$ for some j . We need to show that

$$M, \mu_{\vec{o}} \models F_i(\vec{x}, do(\alpha, s)) \equiv \Phi_i(\vec{x}, \alpha, s) \langle \vec{F} : \vec{Q} \rangle,$$

or equivalently that

$$M, \mu \models F_i(\vec{c}_j, do(\alpha, s)) \equiv \Phi_i(\vec{c}_j, \alpha, s) \langle \vec{F} : \vec{Q} \rangle$$

Let $M, \mu \models F_i(\vec{c}_j, do(\alpha, s))$. We will show that $M, \mu \models \Phi_i(\vec{c}_j, \alpha, s) \langle \vec{F} : \vec{Q} \rangle$. Since μ interprets s as S_0 it follows that $M \models F_i(\vec{c}_j, do(\alpha, S_0))$. By (1) it follows that $M' \models F_i(\vec{c}_j, do(\alpha, S_0))$ and $M', \mu' \models F_i(\vec{c}_j, do(\alpha, s))$, where μ' interprets s as S_0 . Since $M' \models \mathcal{D}_{ss}$ it follows that $M', \mu' \models \Phi_i(\vec{c}_j, \alpha, s)$. Let $\mathcal{G}$ be the characteristic set of α and $\Theta = \{\theta_1(s), \dots, \theta_l(s)\}$ be the set of all $\mathcal{G}$ -models. By Lemma 4 it follows that $M', \mu' \models \mathcal{T}(\mathcal{G}, \Phi_i(\vec{c}_j, \alpha, s))$. By the definition of a $\mathcal{G}$ -model and Θ it follows that there is exactly one k such that $M', \mu' \models \theta_k(s)$. Therefore by the definition of the $\mathcal{T}$ operator it follows that $M', \mu' \models \psi(s)$, where $\psi(s) = \mathcal{V}(\theta_k(s), \Phi_i(\vec{c}_j, \alpha, s))$. By Lemma 5 it follows that $\psi(s)$ is in $\mathcal{U}$ and by Lemma 3 it follows that $M', \mu' \models \psi(do(\alpha, s))$. Therefore $M' \models \psi(do(\alpha, S_0))$. By (1) it follows that $M \models \psi(do(\alpha, S_0))$ and $M, \mu \models \psi(do(\alpha, s))$. By (2) it follows that $M, \mu \models \psi(do(\alpha, s)) \langle \vec{F} : \vec{Q} \rangle$. Therefore

$$M, \mu \models \mathcal{V}(\theta_k(s), \Phi_i(\vec{c}_j, \alpha, s)) \langle \vec{F} : \vec{Q} \rangle. \quad (3)$$

Recall that $M', \mu' \models \theta_k(s)$, where μ' interprets s as S_0 . It follows that $M' \models \theta_k(S_0)$. Since $\theta_k(s)$ is a $\mathcal{G}$ -model it follows that it is a conjunction of literals of the form $\mathcal{F}_i(\vec{c}_j, s)$ or $\neg \mathcal{F}_i(\vec{c}_j, s)$, where $\vec{c}_j$ is in $\mathcal{C}_i$, the argument set of F_i wrt α . By the way we defined R_i it follows that

$$M, \mu \models \theta_k(s) \langle \vec{F} : \vec{Q} \rangle \quad (4)$$

By (3) and (4) it follows that

$$M, \mu \models \theta_k(s) \langle \vec{F} : \vec{Q} \rangle \wedge \mathcal{V}(\theta_k(s), \Phi_i(\vec{c}_j, \alpha, s)) \langle \vec{F} : \vec{Q} \rangle.$$

By the definition of the operator $\mathcal{T}$ it follows that

$$M, \mu \models \mathcal{T}(\mathcal{G}, \Phi_i(\vec{c}_j, \alpha, s)) \langle \vec{F} : \vec{Q} \rangle.$$

By Lemma 4 then follows that

$$M, \mu \models \Phi_i(\vec{c}_j, \alpha, s) \langle \vec{F} : \vec{Q} \rangle.$$

The case where $M, \mu \not\models F_i(\vec{c}_j, do(\alpha, s))$ is very similar.

Finally, to lift the assumption that $\mathcal{D}_0$ is empty we do the following. Let $\phi(S_0)$ uniform in S_0 be the conjunction of all sentences in $\mathcal{D}_0$. In order to show that $M, \mu \models \phi(s) \langle \vec{F} : \vec{Q} \rangle$ we follow the same reasoning as we did to show that $M, \mu \models \Phi_i(\vec{c}_j, \alpha, s) \langle \vec{F} : \vec{Q} \rangle$ starting from the fact that $M', \mu' \models \Phi_i(\vec{c}_j, \alpha, s)$, but starting from the fact that $M', \mu' \models \phi(s)$ instead. ■



2008

This paper was published in the *Proceedings, Eleventh International Conference on Principles of Knowledge Representation and Reasoning*, edited by Gerhard Brewka and Jérôme Lang (Menlo Park, California: AAAI Press).

The 2008 KR conference was held in Sydney, Australia, 16–19 September, 2008.