Inteligencia Artificial 41(2009), 54-63
doi: 10.4114/ia.v13i41.1030

INTELIGENCIA ARTIFICIAL

http://erevista.aepia.org/

Integrating State Constraints and Obligations in Situation Calculus

Robert Demolombe and Pilar Pozos Parra
Institut de Recherche en Informatique de Toulouse
118 Route de Narbonne, F-31062 Toulouse Cedex 9, France
robert.demolombe@orange.fr
Universidad Juárez Autónoma de Tabasco
Carretera Cunduacán - Jalpa Km. 1. Cunduacán, Tabasco, Mexico
pilar.pozos@dais.ujat.mx

AbstractThe ramification problem concerns the characterisation of indirect effects of actions. This problem arises when a theory of action is integrated with a set of state constraints. Thus integrating state constraints to a solution of the frame problem must deal with the ramification problem. In the situation calculus a general solution to both the frame and ramification problems has been proposed which includes the indirect effects of actions in the successor state axioms. However, the notion of belief fluents has also been introduced into the situation calculus in order to distinguish between facts that hold in a situation and facts that are believed to hold in a situation. Hence in addition to the traditional frame and ramification problems, a belief counterpart to each of these problems is considered. Building on these approaches, we propose a general solution to the belief frame and ramification problems. The belief counterpart of the problems must deal with two types of constraints: the believed state constraints relating to physical laws and the constraints imposed by social conventions. These second kind of constraints are known in literature as obligations. A practical advantage is that the proposal is straightforward to implement.

Keywords: State Constraints, Situation Calculus, Ramification Problem.

1. Introduction

Agent technology has been a promising new paradigm for the last 12 years, however most developers use an ad-hoc approach, which, while providing maximum flexibility, results in applications of questionable quality and provide no means for measurement and experience transfer [7]. Many of the challenges associated with agent solutions to complex problems refer to individual possibilities for representation and reasoning about actions, beliefs and plans. A promising approach that considers the above concepts is Situation Calculus that provides a very general and practical formalism for modeling dynamical systems; it can be used for designing and implementing planning systems. However this formalism does not consider agents with social abilities. Some problems must be solved, such as ramification problem in the belief context, before implementing agents capable of obeying social convention. An advantage of including such abilities is that from the various options available to it at a certain moment in time, the agent must select the appropriate (obeying social convections) actions to execute in the planning context.

The frame and ramification problems are classical problems that arise in reasoning about action using formal logic. The frame problem concerns the characterisation of facts that do not change when an

ISSN: 1988-3064(on-line)

action is performed. Reiter proposed a solution to the frame problem in [13]. He introduced the successor state axioms, which are obtained from (positive and negative) effect axioms. This solution, based on the situation calculus, can be very efficiently implemented using regression [15]. However, in his solution, he does not consider the ramification problem, i.e. the axioms are obtained by assuming that there are no state constraints. These constraints are a source of deep theoretical and practical difficulties in modeling dynamical systems [15].

Similarly to [8], McIlraith in [11] proposed an extension to successor state axioms that not only considers direct (positive and negative) effects of actions on fluents, but also indirect effects implicitly defined by a set of state constraints. Even when we have to accept limitations of expressiveness in the state constraints, this approach solves both the frame and ramification problems.

Another extension to the successor state axioms has been proposed in [4], where the authors introduced the notion of belief fluents in order to distinguish between facts that hold in a situation and facts that are believed to hold in a situation. Performance of actions not only results in fluent changes, but contributes toward change in belief fluents as well. So the frame and ramification problems related with facts are distinguished from the belief counterpart of each of these problems which are related with facts that are believed by an agent. In order to address the belief frame problem, the notion of successor belief state axioms has been introduced. The expressive power of this solution is restricted to belief literals; however automated reasoning could be easily implemented. For a large class of applications, the restrictions over the beliefs are not real limitations as shown by in the use of Jason, an interpreter for an extended version of AgentSpeak [1]. Jason provides a platform for the development of multi-agent systems, where an agent is defined by a set of beliefs (ground literals) which constitute the agent's belief base, and a set of plans which form its plan library. Note that the platform is unable to make plans; it requires for each agent the definition of a set of plans.

In this paper, we propose a solution to the belief ramification problem. We consider two kind of constrains in the belief context: the believed state constraints relating to physical laws and the constraints imposed by social convention. For example, suppose a bus driver is at an intersection, the driver must stop if he senses a traffic jam. This is a state constraint since the bus cannot advance because there are others vehicles in its path and two objects cannot be in the same place at the same time. On the other hand, if the driver senses a red light he must stop as well, however in this case there are no physical obstacles impeding his progress, there is a social convention that must be respected. So it is worth noticing that state constraints cannot be violated however constraints imposed by social convention can be. The second type of constraints, those imposed by social laws or normative systems, is well known in literature as obligations. If we make the assumption that the agent decides to respect social convections, the obligations can be integrated in a way similar to that in which believed state constraints are integrated. The proposed solution is similar to McIlraith's approach in the sense that we extend the solution of the belief frame problem to include believed constraints. However, our approach is not restrained to stratified theories as McIlraith's approach. Moreover the successor belief state axioms are extended in order to include both state constraints and obligations, thus the agent can reason about social laws in a similar way to that used to reason about nature laws. Moreover automated reasoning could be easily implemented.

The rest of the paper is organized as follows: in the next section intuitive ideas concerning the representation of the world are presented through a simple example. Then we propose a logical solution to the ramification problem concerning the representation of the world. Section 4 gives an example concerning belief representation, followed by the general logical solution for belief problems, and finally the conclusions are presented.

2. World Representation

Situation calculus is a first-order, many-sorted language with equality and the usual alphabet of logical symbols and their respective definitions [15]. Situations represent sequences of actions which have been performed from the initial state to a current state. A situation is syntactically represented by a term of the form $do(a, s)$ where a denotes an action, and s denotes a situation. The initial situation is denoted by S_0. Predicates whose value may change from situation to situation are called *fluents*. The last argument of a fluent is a situation. For instance, $position(x, s)$ represents the fact that a given object is at the position x in the situation s. Action and situation variables can be quantified. For instance, $\neg\exists s(position(2, s))$

represents the fact that in no situation is a given object at position 2. Action quantification is an essential feature in the solution to the frame problem proposed by Reiter.

To intuitively present how the solution to the frame problem can be extended to include state constraints, we use the following scenario.

Let's consider a simple robot that can move forward (action *adv*) or backward (action *rev*) along a rail track. Performing actions *adv* or *rev* changes its position by one distance unit.

The evolution of the fluents is defined by the successor state axioms. These axioms represent a solution to the frame problem as well. For each fluent F we have an axiom of the following form[1]:

$$F(do(a,s)) \leftrightarrow \Upsilon_F^+(a,s) \vee F(s) \wedge \neg\Upsilon_F^-(a,s)$$

where Υ_F^+ characterises all the conditions that have positive effects on the fluent F, and Υ_F^- characterises all the conditions that have negative effects on the fluent F.

For example, for the fluent *position*, the positive effect of performing the action *adv* (respectively *rev*) when the robot is at the position $x-1$ (respectively $x+1$) in the situation s, is that it is at the position x in the situation $do(a,s)$. The positive effects on the position of a robot are represented by:

$$\begin{aligned}&[a = adv \wedge position(x-1,s) \vee \\ &\ a = rev \wedge position(x+1,s)] \\ &\quad \rightarrow position(x, do(a,s))\end{aligned} \tag{1}$$

The negative effect axiom expresses that if the robot is at the position x in the situation s and he performs either the action *adv* or the action *rev*, then in the situation $do(a,s)$ he is no longer at the position x:

$$\begin{aligned}&(a = adv \vee a = rev) \wedge position(x,s) \\ &\quad \rightarrow \neg position(x, do(a,s))\end{aligned} \tag{2}$$

and the successor state axiom is:

$$\begin{aligned}&position(x, do(a,s)) \leftrightarrow \\ &\quad [a = adv \wedge position(x-1,s) \vee \\ &\quad a = rev \wedge position(x+1,s)] \vee position(x,s) \\ &\quad \wedge\neg[(a = adv \vee a = rev) \wedge position(x,s)]\end{aligned}$$

This axiom represents the evolution of *position* in terms of direct (positive and negative) effects of actions but the axiom does not consider the indirect effects of actions.

State constraints represent invariant properties, that is properties that remain unchanged in every situation. For instance, the fact that the robot's position is unique is represented by the constraint:

$$position(x,s) \wedge position(x',s) \rightarrow x = x' \tag{3}$$

Let T be a basic theory of action [15], which includes successor state axioms and unique names axioms for actions and situations, and let ψ be a state constraint. To check if the constraint is satisfied in the theory, we have to prove that ψ is a consequence of T. That is we have to prove: $T \vdash \psi$. The formula ψ is assumed to be a standard formula of the situation calculus language.

McIlraith proposed considering state constraints as special cases of the conditions that make a fluent have the value "true" or "false" [11]. A practical advantage of this approach is that we can use the results proposed in [15] for automated reasoning. In what follows, we present a syntactic manipulation for including constraint (3) to the evolution of fluent *position*. From (3) we have: $position(x,S_0) \wedge position(x',S_0) \rightarrow x = x'$ and $position(x,do(a,s)) \wedge position(x',do(a,s)) \rightarrow x = x'$, and then from the second formula we obtain:

$$\begin{aligned}&position(x', do(a,s)) \wedge \neg(x = x') \\ &\quad \rightarrow \neg position(x, do(a,s))\end{aligned} \tag{4}$$

[1] We adopt the convention that all the variables are universally quantified.

Now by applying properties of classical logic we obtain **all** the conditions that cause negative effects on the fluent from (2) and (4), and these conditions include the state constraint.

$$\begin{aligned}&[(a = adv \lor a = rev) \land position(x, s) \lor\\ &position(x', do(a, s)) \land \neg(x = x')]\\ &\quad \rightarrow \neg position(x, do(a, s))\end{aligned} \tag{5}$$

If it is assumed that the conditions (1) and (5) are a complete representation of the conditions that may change the value of the fluent, we get the following successor state axiom (see [13] for details):

$$\begin{aligned}&position(x, do(a, s)) \leftrightarrow\\ &\quad [a = adv \land position(x-1, s) \lor\\ &\quad a = rev \land position(x + 1, s)] \lor position(x, s)\\ &\quad \land\neg[(a = adv \lor a = rev) \land position(x, s) \lor\\ &\quad position(x', do(a, s)) \land \neg(x = x')]\end{aligned}$$

McIlraith called the result Intermediate Successor State Axioms. In order to obtain the Final Successor State Axioms, she proposed a stratified replacement of atoms expressed in terms of $do(a, s)$ for the corresponding right-hand side (RHS) of the intermediate successor state axioms. The method is restrained to stratified theories in order to guarantee the termination of the process. The RHS of the final axioms are always expressed in terms of s. See [11] for a detailed description.

As we can see the example does not consider a stratified theory. The constraint (4) is not a stratified formula. The intent of replacing the atom $position(x', do(a, s))$ in and by the RHS of the intermediate successor state axiom of *position* forces the method to draw in a infinite cycle.

3. Extending Successor State Axioms

We propose a syntactic manipulation similar to that proposed by McIlraith in order to extend Reiter's successor state axioms to include the state constraints thereby solving the ramification problem.

Let F be a fluent, $\Upsilon^+_F(a, s)$ all the conditions that have direct positive effect on the fluent F, and $\Upsilon^-_F(a, s)$ all the conditions that have direct negative effect on the fluent F. These Υ's are called the primary causes. Moreover, we assume that $\Upsilon^+_F(a, s)$ and $\Upsilon^-_F(a, s)$ are disjoint. With a limited loss of generality, it is assumed that all the state constraints where F occurs can be represented as follows:

$$\begin{aligned}\psi(do(a, s)) &\rightarrow F(do(a, s))\\ \theta(do(a, s)) &\rightarrow \neg F(do(a, s))\end{aligned}$$

$\psi(do(a, s))$ and $\theta(do(a, s))$ are called the secondary causes and their effects over F depend of direct effects on the fluents appearing in them. In order to guarantee the satisfaction of state constraints in any situations the initial theory must satisfy $\psi(S_0) \rightarrow F(S_0)$ and $\theta(S_0) \rightarrow \neg F(S_0)$.

The first step is the following replacement: every fluent $\alpha(do(a, s))$ appearing in $\psi(do(a, s))$ or $\theta(do(a, s))$ is replaced with the condition $\Upsilon^+_\alpha(a, s)$, and the condition $\Upsilon^-_\alpha(a, s)$ substitutes any negation of fluent $\alpha(do(a, s))$ appearing in $\psi(do(a, s))$ or $\theta(do(a, s))$. So the resulting formulas $\psi'(a, s) = replacement(\psi(do(a, s)))$ and $\theta'(a, s) = replacement(\theta(do(a, s)))$ are expressed solely in terms of s. So we have:

$$\begin{aligned}\psi'(a, s) &\rightarrow F(do(a, s))\\ \theta'(a, s) &\rightarrow \neg F(do(a, s))\end{aligned}$$

The substitution means intuitively that if the direct effects of actions trigger the condition $\psi'(a, s)$ or $\theta'(a, s)$ then fluent F changes due to indirect effects. In this way the primary causes define the indirect effects as well. It is worth noticing that while McIlraith's replacement considers formulas equivalents, we obtain the result by the transitive property of implication.

The next step considers the following causal completeness assumption: all the conditions under which an action a can lead, directly or indirectly, to fluent F becoming true or false in the successor state are characterised for $\Upsilon^+_F(a,s) \vee \psi'(a,s)$ and $\Upsilon^-_F(a,s) \vee \theta'(a,s)$, respectively. Then the new form of the successor state axiom that includes the state constraints is as follows:

$$\begin{aligned} F(do(a,s)) \leftrightarrow & [\Upsilon^+_F(a,s) \vee \psi'(a,s)] \vee \\ & F(s) \wedge \neg[\Upsilon^-_F(a,s) \vee \theta'(a,s)] \end{aligned}$$

This axiom may be understood as follows: F holds in $do(a,s)$ if and only if an action made it true or an indirect effect made it true or F was true in s and neither an action nor an indirect effect made it false.

It is assumed that for each fluent F, the positive and negative conditions are disjoint[2]:

$$\vdash T \rightarrow \forall\neg([\Upsilon^+_F(a,s) \vee \psi'(a,s)] \wedge [\Upsilon^-_F(a,s) \vee \theta'(a,s)])$$

4. Belief Representation

To define the subjective representation of the evolution of the world, the language is extended with belief fluents of the form B_iF [4]. We say that the fluent B_iF holds in situation s if agent i believes that F holds in situation s and represent it as $B_iF(s)$. Similarly $B_i\neg F(s)$ represents the fact that the fluent $B_i\neg F$ holds in situation s: the agent i believes that F does not hold in situation s. Note that the approach extends the language of situation calculus by two symbols: B_iF and $B_i\neg F(s)$ for each fluent F appearing in the original language of situation calculus, which are not modal operators.

To introduce intuitively the solution to the frame problem extended to the belief context let's consider the robot's scenario. To represent the evolution of robot's beliefs, we have to consider four effect axioms for each fluent F. For example, for the fluent $position(x,s)$, there are four distinct possible attitudes of the robot which are formally represented by: $B_rposition(x,s)$, $B_r\neg position(x,s)$, $\neg B_rposition(x,s)$ and $\neg B_r\neg position(x,s)$. Their corresponding axioms (6), (7), (8) and (9) are given below.

The effect of performing action adv (respectively rev) when the robot believes that it is at the position $x-1$ (respectively $x+1$) in the situation s is that it believes that it is at the position x in the situation $do(a,s)$. The same belief is produced by the following condition: it senses the position (action $sense$) in s and the real position is x:

$$\begin{aligned} &[a = adv \wedge B_rposition(x-1,s) \vee \\ &\ a = rev \wedge B_rposition(x+1,s) \vee \\ &\ a = sense \wedge position(x,s)] \\ &\quad \rightarrow B_rposition(x, do(a,s)) \end{aligned} \tag{6}$$

Note that the two first conditions are examples of updates and the last one which concerns a sensing action is a revision. The revision does not depend of an earlier epistemic state of the agent. Moreover revisions make the connections with the properties representing the world.

The effect of performing either action adv or rev when the robot believes that it is at the position x in the situation s is that in the situation $do(a,s)$ it no longer believe it is at position x:

$$\begin{aligned} &(a = adv \vee a = rev) \wedge B_rposition(x,s) \\ &\quad \rightarrow \neg B_rposition(x, do(a,s)) \end{aligned} \tag{7}$$

We have two similar axioms defining, in the situation $do(a,s)$, the belief and unbelief attitudes of the robot with respect to the fact that it is not at position x. We have:

$$\begin{aligned} &(a = adv \vee a = rev) \wedge B_rposition(x,s) \\ &\quad \rightarrow B_r\neg position(x, do(a,s)) \end{aligned} \tag{8}$$

$$\begin{aligned} &[a = adv \wedge B_rposition(x-1,s) \vee \\ &\ a = rev \wedge B_rposition(x+1,s) \vee \\ &\ a = sense \wedge position(x,s)] \\ &\quad \rightarrow \neg B_r\neg position(x, do(a,s)) \end{aligned} \tag{9}$$

[2]We use the symbol $\forall$ to denote the universal closure of all the free variables in the scope of $\forall$.

If we extend the causal completeness assumptions to the robot's beliefs, we get, after some simplifications, the successor belief state axioms for the belief of robot concerning its position (see axioms (12) and (13) for the general form):

$$\begin{aligned} & B_r position(x, do(a, s)) \leftrightarrow \\ & \quad [a = adv \wedge B_r position(x{-}1, s) \vee \\ & \quad a = rev \wedge B_r position(x{+}1, s) \vee \\ & \quad a = sense \wedge position(x, s)] \vee \\ & \quad B_r position(x, s) \wedge \neg[(a = adv \\ & \quad \vee a = rev) \wedge B_r position(x, s)] \end{aligned} \tag{10}$$

$$\begin{aligned} & B_r \neg position(x, do(a, s)) \leftrightarrow \\ & \quad [(a = adv \vee a = rev) \wedge \\ & \quad B_r position(x, s)] \vee \\ & \quad B_r \neg position(x, s) \wedge \\ & \quad \neg[a = adv \wedge B_r position(x{-}1, s) \vee \\ & \quad a = rev \wedge B_r position(x{+}1, s) \vee \\ & \quad a = sense \wedge position(x, s)] \end{aligned} \tag{11}$$

The definitions assume implicitly that if the robot performs the actions *adv*, *rev* or *sense*, it believes that it has performed these actions. However, the beliefs of an agent who does not know about the performance of these actions can evolve in a different way. It is worth noticing that the approach allows the representation of different evolutions of beliefs concerning the same fluent. Consider a robot's pilot or controller, if the only way for the pilot to be informed about the position is by checking it on a control panel (action $obs.position(x)$), then we have the following successor belief axioms for the pilot.

$$\begin{aligned} & B_p position(x, do(a, s)) \leftrightarrow \\ & \quad a = obs.position(x) \vee B_p position(x, s) \end{aligned}$$

$$\begin{aligned} & B_p \neg position(x, do(a, s)) \leftrightarrow \\ & \quad B_p \neg position(x, s) \wedge \neg(a = obs.position(x)) \end{aligned}$$

In general for each agent i and each fluent F there are two axioms of the form:

$$\begin{aligned} & B_i F(do(a, s)) \leftrightarrow \\ & \quad \gamma^+_{i_1,F}(a, s) \vee B_i F(s) \wedge \neg\gamma^-_{i_1,F}(a, s) \end{aligned} \tag{12}$$

$$\begin{aligned} & B_i \neg F(do(a, s)) \leftrightarrow \\ & \quad \gamma^+_{i_2,F}(a, s) \vee B_i \neg F(s) \wedge \neg\gamma^-_{i_2,F}(a, s) \end{aligned} \tag{13}$$

There are some assumptions that guarantee consistent beliefs [4].

5. Extending Successor Belief State Axioms

We consider now the integration of invariant beliefs, that is integration of beliefs that remain unchanged in every situation. We include first the believed state constraints. These are invariant beliefs concerning physical laws. For instance, the fact that the robot believes that the position is unique is represented by the following believed state constraint:

$$\begin{aligned} & B_r position(x, s) \wedge B_r position(x', s) \\ & \quad \rightarrow x = x' \end{aligned} \tag{14}$$

The procedure for including state constraints to the successor state axiom can be adapted to include believed state constraints to the successor belief state axioms.

Let B_iF and $B_i\neg F$ be the belief fluents associated to the agent i and the fluent F, $\gamma^+_{i_1,F}(a,s)$ all the conditions that have direct positive effect on the fluent B_iF, $\gamma^-_{i_1,F}(a,s)$ all the conditions that have direct negative effect on the fluent B_iF, $\gamma^+_{i_2,F}(a,s)$ all the conditions that have direct positive effect on the fluent $B_i\neg F$, and $\gamma^-_{i_2,F}(a,s)$ all the conditions that have direct negative effect on the fluent $B_i\neg F$. With a limited loss of generality, it is assumed that all the believed state constraints where B_iF and $B_i\neg F$ occur can be represented as follows:

$$\begin{array}{rcl}\psi_1(do(a,s)) & \rightarrow & B_iF(do(a,s))\\ \theta_1(do(a,s)) & \rightarrow & \neg B_iF(do(a,s))\\ \psi_2(do(a,s)) & \rightarrow & B_i\neg F(do(a,s))\\ \theta_2(do(a,s)) & \rightarrow & \neg B_i\neg F(do(a,s))\end{array}$$

We assume that the initial theory satisfies $\psi_1(S_0) \rightarrow B_iF(S_0)$, $\theta_1(S_0) \rightarrow \neg B_iF(S_0)$, $\psi_2(S_0) \rightarrow B_i\neg F(S_0)$, and $\theta_2(S_0) \rightarrow \neg B_i\neg F(S_0)$.

The first step transforms the formulas $\psi_1(do(a,s))$, $\theta_1(do(a,s))$, $\psi_2(do(a,s))$, and $\theta_2(do(a,s))$ in terms of direct effects over belief fluents. Every belief $B_i\alpha$ ($B_i\neg\alpha$, resp.) appearing in these formulas is replaced with $\gamma^+_{i_1,\alpha}(a,s)$ ($\gamma^+_{i_2,\alpha}(a,s)$, resp.) and every unbelief $\neg B_i\alpha$ ($\neg B_i\neg\alpha$, resp.) is replaced with $\gamma^-_{i_1,\alpha}(a,s)$ ($\gamma^-_{i_2,\alpha}(a,s)$, resp.). If $\psi'_1(a,s)$, $\theta'_1(a,s)$, $\psi'_2(a,s)$ and $\theta'_2(a,s)$ represent the result of the replacement in $\psi_1(do(a,s))$, $\theta_1(do(a,s))$, $\psi_2(do(a,s))$ and $\theta_2(do(a,s))$, respectively, the believed state constrains take the following form:

$$\begin{array}{rcl}\psi'_1(a,s) & \rightarrow & B_iF(do(a,s))\\ \theta'_1(a,s) & \rightarrow & \neg B_iF(do(a,s))\\ \psi'_2(a,s) & \rightarrow & B_i\neg F(do(a,s))\\ \theta'_2(a,s) & \rightarrow & \neg B_i\neg F(do(a,s))\end{array}$$

Next step considers the following causal completeness assumption: all the conditions under which an action a can lead, directly or indirectly, to belief fluent B_iF becoming true or false in the successor state are characterised for $\gamma^+_{i_1,F}(a,s) \vee \psi'_1(a,s)$ and $\gamma^-_{i_1,F}(a,s) \vee \theta'_1(a,s)$, respectively. And all the conditions under which an action a can lead, directly or indirectly, to belief fluent $B_i\neg F$ becoming true or false in the successor state are characterised for $\gamma^+_{i_2,F}(a,s) \vee \psi'_2(a,s)$ and $\gamma^-_{i_2,F}(a,s) \vee \theta'_2(a,s)$, respectively. Then the successor belief state axioms that include the believed state constraints take the following forms:

$$\begin{array}{l}B_iF(do(a,s)) \leftrightarrow [\gamma^+_{i_1,F}(a,s) \vee \psi'_1(a,s)]\\ \qquad \vee B_iF(s) \wedge \neg[\gamma^-_{i_1,F}(a,s) \vee \theta'_1(a,s)]\end{array}$$

$$\begin{array}{l}B_i\neg F(do(a,s)) \leftrightarrow [\gamma^+_{i_2,F}(a,s) \vee \psi'_2(a,s)]\\ \qquad \vee B_i\neg F(s) \wedge \neg[\gamma^-_{i_2,F}(a,s) \vee \theta'_2(a,s)]\end{array}$$

The first axiom may be understood as follows: i believes F holds in $do(a,s)$ if and only if an action made i believe F holds or an indirect effect made i believe F holds or i believes F was true in s and neither an action nor an indirect effect made disbelieve it. The second axiom has a similar interpretation.

In order to maintain consistent beliefs, it is assumed that the theory T meets the following conditions:

(P1) $\vdash T \rightarrow \forall\neg([\gamma^+_{i_1,F}(a,s) \vee \psi'_1(a,s)] \wedge [\gamma^-_{i_1,F}(a,s) \vee \theta'_1(a,s)])$

(P2) $\vdash T \rightarrow \forall\neg([\gamma^+_{i_2,F}(a,s) \vee \psi'_2(a,s)] \wedge [\gamma^-_{i_2,F}(a,s) \vee \theta'_2(a,s)])$

(P3) $\vdash T \rightarrow \forall\neg([\gamma^+_{i_1,F}(a,s) \vee \psi'_1(a,s)] \wedge [\gamma^+_{i_2,F}(a,s) \vee \psi'_2(a,s)])$

(P4) $\vdash T \rightarrow \forall(B_iF(s) \wedge [\gamma^{+}_{i_2,F}(a,s) \vee \psi'_2(a,s)] \rightarrow [\gamma^{-}_{i_1,F}(a,s) \vee \theta'_1(a,s)])$

(P5) $\vdash T \rightarrow \forall(B_i\neg F(s) \wedge [\gamma^{+}_{i_1,F}(a,s) \vee \psi'_1(a,s)] \rightarrow [\gamma^{-}_{i_2,F}(a,s) \vee \theta'_2(a,s)])$

We include now the believed social conventions or obligations. These are invariant beliefs concerning social laws. Notice that state constraints cannot be physically violated. For instance, an object cannot be in two places at the same time (see for example (3)). However, in the case of obligations (social constraints), agents might violate them. Suppose the following formula:

$$traffic\_light(x, red, s) \rightarrow \neg position(x, s)$$

representing a social convention, where the fluent $traffic\_light(x, color, s)$ means there is a traffic light corresponding to a road intersection located at x to indicate when it is safe to advance in s using the following color code: if *color* is *red*, it is not safe to advance; if *color* is *green*, it is safe to advance; if *color* is *yellow*, it is a preventive sing saying that *color* is changing from *green* to *red*. Obviously the color *red* in the traffic light is not a physical barrier to movement into the position that constitutes the intersection and the rule can be violated by the agent. However in order to allow agents to advance safely, the normative system defines the "social"barrier. Suppose also an ideal world where every agent obeys the rule. This ideal world is defined through a set of obligations. For instance, we have the following obligation:

$$traffic\_light(x, red, s) \rightarrow O\neg position(x, s) \tag{15}$$

which means if in the real world there is a crossroads with red traffic light then ideally the agent is not at the crossroads. Notice that the consequent does not represent a real fact nor a fact believed by an agent. $O\neg position(x, s)$ represents an ideal fact imposes by the normative system. Here, our interest does not concern the representation of ideal world evolution which must integrate a solution to the corresponding frame problem (see for example [3, 5, 12]). Our concern is the design of obedient agents, i.e. agents respecting social constraints.

It is worth noticing that when a state constraints such as (3) is believed by an agent, the corresponding believed state constraint has the same schema (see (14)). However when an obligation as (15) is known by an agent the mapping is not clear. Suppose a similar schema:

$$\begin{aligned} &B_r traffic\_light(x, red, s) \\ &\rightarrow B_r O(\neg position(x, s))) \end{aligned} \tag{16}$$

where the belief fluent $B_r traffic\_light(x, color, s)$ represents the robot's belief about the traffic light located at x, and $B_rO(\neg position(x, s))$ represents the robot's belief about the obligation concerning the position. So the formula (16) represents the understanding of the obligation but this formula does not define an obedient attitude about such an obligation.

Our approach considers the fact that agents create habits by force of respecting social rules. For example, some people stop automatically when they see a red traffic light without thinking about the related social rule. Our approach proposes to formalise this behaviour in order to assist a planner in the search of actions that are socially valid. So we make the assumption that the agent decides to respect social convections. For example, the believed social constraint:

$$\begin{aligned} &B_r position(x, s) \wedge B_r traffic\_light(x + 1, red, s) \\ &\rightarrow B_r position(x, do(a, s)) \end{aligned}$$

represents robot's custom in order to obey the rule (15). The intuitive meanings is that if the robot believes itself to be in front of a crossroads which has a red traffic light then wherever that happens, it does not change its position (it stops). The interesting aspect of integration of obligations in this way concerns plan generation. In order to choose the appropriate actions the agent must project its beliefs, if the agent want to avoid punishments, its projections must consider the satisfaction of social rules. These kind of constraints can be integrated to the successor belief state axioms in a similar manner to that in which believed state constraints were integrated.

Using the proposal we obtain the extended successor belief state axiom for $B_r position$ which includes both believed state and social constraints:

$$\begin{aligned}
&B_r position(x, do(a, s)) \leftrightarrow \\
&\quad [a = adv \wedge B_r position(x{-}1, s) \vee \\
&\quad a = rev \wedge B_r position(x{+}1, s) \vee \\
&\quad a = sense \wedge position(x, s) \vee \\
&\quad B_r position(x, s) \wedge \\
&\quad B_r traff\,ic_light(x{+}1, red, s)] \vee \\
&\quad B_r position(x, s) \wedge \neg[(a = adv \vee a = rev) \wedge \\
&\quad B_r position(x, s) \vee \\
&\quad (a = adv \wedge B_r position(y{-}1, s) \vee \\
&\quad a = rev \wedge B_r position(y{+}1, s) \vee \\
&\quad a = sense \wedge position(y, s)) \wedge \neg y = x]
\end{aligned}$$

The agent might violate the obligations in cases where it is not aware about some properties. For instance, if the robot does not know its position or the traffic light color, it can draw in a violation state. The proposal allows the definition of different behaviour as a result of the believed social constraints. For example, if we need a robot that tries to exit of a state that violates some social constraints as soon as it is aware of such violation, we can define the following constraint:

$$\begin{aligned}
&B_r position(x, s) \wedge B_r traffic_light(x, red, s) \\
&\quad \rightarrow B_r position(x + 1, do(a, s))
\end{aligned}$$

meaning that if the robot believes itself to be at a crossroads with a red traffic light, which can happen after a sensing action, it will try to move to the next position in order to pass the intersection.

6. Conclusion

We have presented a general solution to solve the frame and ramification problems in the belief context. A significant difference between this approach and that presented in [14] is that it can be easily implemented and the proof by induction is restricted to two steps (just as for mathematical induction): verification in the initial state and verification of the successor state. Constraints that are included in the successor belief state axioms could influence the agent to take a decision. For example, if a robot believes it cannot be stopped at an intersection with a red traffic light, then it will reject the goal of having to wait for someone at the intersection. The regression theorem presented in [15] justifies a simple Prolog implementation for reasoning with this proposal assuming a closed initial database.

Referencias

[1] R. H. Bordini, J. F. Hübner, and M. Wooldridge. *Programming Multi-Agent Systems in AgentSpeak Using Jason.* John Wiley and Sons, Ltd, 2007.

[2] F. M. Brown, editor. *The Frame Problem in Artificial Intelligence.* Morgan Kaufmann Publishers, 1987.

[3] R. Demolombe and A. Herzig. Obligation change in dependence logic and situation calculus. In *Proc. of the 7th International Workshop on Deontic Logic in Computer Science*, 2004.

[4] R. Demolombe and P. Pozos Parra. A simple and tractable extension of situation calculus to epistemic logic. In Z. W. Ras and S. Ohsuga, editors, *Proc. of 12th International Symposium ISMIS 2000.* Springer. LNAI 1932, 2000.

[5] Robert Demolombe and Pilar Pozos Parra. The chisholm paradox and the situation calculus. In *ISMIS*, pages 425–434, 2005.

[6] Marc Denecker and Eugenia Ternovska. Inductive situation calculus. *Artif. Intell.*, 171(5-6):332–360, 2007. doi: 10.1016/j.artint.2007.02.002.

[7] Virginia Dignum and Frank Dignum. A Landscape of Agent Systems for the Real World. Technical report, Institute of Information and Computing sciences, Utrecht University, 2006.

[8] F. Lin and R. Reiter. State constraints revisited. *The journal of Logic and Computation. Special issue on action and processes*, 4:655–678, 1994.

[9] D. Loveland. *Automated Theorem Proving: a logical basis.* North-Holland, 1978.

[10] J. C. McCarthy and P. Hayes. Some Philosophical Problems from the Standpoint of Artificial Intellifence. In D. Michie B. Meltzer, editor, *Machine Intelligence*, volume 4, pages 463–502. Edinburgh University Press, 1969.

[11] S. McIlraith. Integrating action and state constraints: A closed-form solution to the ramification problem (sometimes). *Artificial Intelligence*, 116, 2000.

[12] P. Pozos Parra, A. Nayak, and R. Demolombe. Theories of intentions in the framework of situation calculus. In J. Leite, A. Omicini, P. Torroni, and P. Yolum, editors, *Declarative Agent Languages and Technologies*. Springer Verlag, 2005.

[13] R. Reiter. The frame problem in the situation calculus: a simple solution (sometimes) and a completeness result for goal regression. In V. Lifschitz, editor, *Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy*, pages 359–380. Academic Press, 1991.

[14] R. Reiter. Proving Properties of States in the Situation Calculus. *Artificial Intelligence*, 64:337–351, 1993.

[15] R. Reiter. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems.* The MIT Press, 2001.