Dynamic Interactions Between Goals and Beliefs*

Steven Shapiro[1], Gerhard Brewka[2]

[1] University of Toronto
Dept. of Mechanical and Industrial Engineering
5 King's College Road
Toronto, Ontario M5S 3G8
Canada
shapiro@mie.utoronto.ca
[2] University of Leipzig
Computer Science Inst.
Augustusplatz 10–11
04109 Leipzig, Germany
brewka@informatik.uni-leipzig.de

Abstract. Shapiro *et al.* [1,2], presented a framework for representing goal change in the situation calculus. In that framework, agents adopt a goal when requested to do so (by some agent *reqr*), and they remain committed to the goal unless the request is cancelled by *reqr*. A common assumption in the agent theory literature, e.g., [3,4], is that achievement goals that are believed to be impossible to achieve should be dropped. In this paper, we incorporate this assumption into Shapiro *et al.*'s framework, however we go a step further. If an agent believes a goal is impossible to achieve, it is dropped. However, if the agent later believes that it was mistaken about the impossibility of achieving the goal, the agent might readopt the goal. In addition, we consider an agent's goals as a whole when making them compatible with their beliefs, rather than considering them individually.

1 Introduction

Shapiro *et al.* [1,2], presented a framework for representing goal change in the situation calculus. In that framework, agents adopt a goal when requested to do so (by some agent *reqr*), and they remain committed to the goal unless the request is cancelled by *reqr*. A common assumption in the agent theory literature, e.g., [3,4], is that achievement goals that are believed to be impossible to achieve should be dropped. In this paper, we incorporate this assumption into Shapiro *et al.*'s framework, however we differ from previous approaches in two respects. If an agent believes a goal is impossible to achieve, it is dropped. However, if the agent revises its beliefs, it may later come to believe that it was mistaken about the impossibility of achieving the goal. In that case, the agent should *readopt* the goal. To our knowledge, this has not been considered in previous approaches. In addition, most frameworks[1] consider goals in isolation when checking compatibility with beliefs. However, it could be the case that each goal individually is compatible with an agent's beliefs, but the set of all goals of the agent is incompatible with its beliefs.

In Sec. 2, we present the situation calculus and Reiter's action theories, which form the basis of our framework. In Sec. 3, we present Shapiro *et al.*'s framework, and in Sec. 4, we show how to extend the framework to take into consideration the dynamic interactions between beliefs and goals. Some properties of the new framework are presented in Sec. 5. In Sec. 6, we sketch how to extend the framework further so that achievement goals that are believed to have been already achieved are dropped by the agents. We conclude in Sec. 7.

* An earlier version of this paper appeared in Proc. IJCAI-07.
[1] Bell and Huang [5] consider the compatibility of all of an agent's goals with its beliefs, but they do not consider the possibility of readopting a goal previously believed impossible.

Dagstuhl Seminar Proceedings 07351
Formal Models of Belief Change in Rational Agents
http://drops.dagstuhl.de/opus/volltexte/2007/1199

2 Representation of Action and Beliefs

The basis of our framework for goal change is an action theory [6] based on the situation calculus [7,8]. The situation calculus is a predicate calculus language for representing dynamically changing domains. A situation represents a possible state of the domain. There is a set of initial situations corresponding to the ways the agents believe the domain might be initially. The actual initial state of the domain is represented by the distinguished initial situation constant, S_0. The term $do(a, s)$ denotes the unique situation that results from the agent doing action a in situation s. Thus, the situations can be structured into a set of trees, where the root of each tree is an initial situation and the arcs are actions. The sequence of situations that precedes a situation s in its tree is called *history* of s. Predicates and functions whose value may change from situation to situation (and whose last argument is a situation) are called *fluents*. For instance, we might use the fluent $\textsc{InRoom}(Agt, R_1, S)$ to represent the fact that agent Agt is in room R_1 in situation S. The effects of actions on fluents are defined using successor state axioms [6], which provide a succinct representation for both effect axioms and frame axioms [7].

We will be quantifying over formulae, so we assume that we have an encoding of formulae as first-order terms. As shown by De Giacomo *et al.* [9], this is laborious but straightforward. It includes introducing constants for variables, defining substitution, introducing a *Holds* predicate to define the truth of formulae, etc. We assume we have such an axiomatization, and so we will freely quantify over formulae here (using first-order quantifiers). To simplify notation, we ignore the details of the encoding and use formulae directly instead of the terms that represent them.

We will also be using lists of formulae, so we need an axiomatization of lists. We do not present the details here but such a formalization is well known. We use the functions $car(l)$, $cdr(l)$, $cons(\psi, l)$, $reverse(l)$, and $remove(\psi, l)$; and the relation $member(\psi, l)$ with their usual meanings. *nil* denotes the empty list. We will also use lists of formulae (without repetitions) to represent finite sets of formulae, and therefore use finite sets when it is convenient, along with the usual set functions and relations.

To axiomatize a dynamic domain in the situation calculus, we use Reiter's [6] action theory, which consists of (1) successor state axioms for each fluent; (2) initial state axioms, which describe the initial state of the domain and the initial mental states of the agents; (3) unique names axioms for the actions, and domain-independent foundational axioms (given below); and (4) the axioms to encode formulae as terms, and to define lists of (terms for) formulae.[2]

Unique names axioms are used to ensure that distinct action function symbols denote different actions. For distinct action function symbols, $\mathbf{a_1}$ and $\mathbf{a_2}$, we need an axiom of the following form:[3]

Axiom 1

$$\mathbf{a_1}(\boldsymbol{x}) \neq \mathbf{a_2}(\boldsymbol{y}).$$

Also, for each action function symbol, $\mathbf{a}$, we need an axiom of the following form:

Axiom 2

$$\mathbf{a}(\boldsymbol{x}) = \mathbf{a}(\boldsymbol{y}) \supset \boldsymbol{x} = \boldsymbol{y}.$$

We want the situations to be the smallest set generated by sequences of actions starting in an initial situation. We axiomatize the structure of the situations with *foundational axioms* based on the ones listed in Levesque *et al.* [8] for the language of the "epistemic situation calculus". We first define the initial situations to be those that have no predecessors:

$$Init(s') \stackrel{\text{def}}{=} \neg\exists a, s.s' = do(a, s)$$

We declare S_0 to be an initial situation.

[2] Action theories normally also include axioms to specify the distinguished predicate $Poss(a, s)$ which is used to described the conditions under which it is physically possible to execute an action, however to simplify notation, we omit the use of *Poss* here and assume that it is always possible to execute all actions.

[3] We adopt the convention that unbound variables are universally quantified in the widest scope.

Axiom 3

$$Init(S_0)$$

We also need an axiom stating that do is injective.

Axiom 4

$$do(a_1, s_1) = do(a_2, s_2) \supset (a_1 = a_2 \wedge s_1 = s_2)$$

The induction axiom for situations says that if a property P holds of all initial situations, and P holds for all successors to situation s if it holds for s, then P holds for all situations.

Axiom 5

$$\forall P.[(\forall s.Init(s) \supset P(s)) \wedge (\forall a, s.P(s) \supset P(do(a, s)))] \supset \\ \forall s P(s).$$

We now define precedence for situations. We say that s *strictly precedes* s' if there is a (non-empty) sequence of actions that take s to s'.

Axiom 6

$$\forall s_1, s_2.s_1 \prec s_2 \equiv (\exists a, s.s_2 = do(a, s) \wedge (s_1 \preceq s)),$$

where $s_1 \preceq s_2 \stackrel{\text{def}}{=} s_1 = s_2 \vee s_1 \prec s_2$ denotes that s_1 *precedes* s_2.

Although belief change plays a large role in this paper, the focus is on the goal change framework. Belief change frameworks in Reiter's action theory framework have been developed [10,11,12], however we will not assume a particular framework here. Instead, we will make a few general assumptions about the belief change framework as needed. In particular, we assume a possible worlds approach (with B as the accessibility relation) using situations as possible worlds. The accessible situations are the ones that are used to determine the beliefs of the agent, as usual. These would correspond to, e.g., the most plausible accessible situations of Shapiro *et al.* [10] or simply the situations that the agent considers possible in a framework without plausibilities over situations. Therefore, we assume that the language contains a distinguished predicate $B(agt, s', s)$. We also assume that the agents' beliefs are always consistent:

Axiom 7

$$\forall s \exists s' B(agt, s', s).$$

The beliefs of the agent are defined as those formulae true in all the accessible situations:

$$\mathbf{Bel}(agt, \phi, s) \stackrel{\text{def}}{=} \forall s'.B(agt, s', s) \supset \phi[s'].$$

Here, ϕ is a formula that may contain the distinguished constant Now instead of its (final) situation argument. $\phi[s]$ denotes the formula that results from substituting s for Now in ϕ. When the intended meaning is clear, we may suppress this situation argument of ϕ.

3 Goal Change

Shapiro *et al.* [1,2], presented a framework for representing goal change in the situation calculus. In that framework, agents adopt goals when requested to do so (by some agent $reqr$) and they remain committed to their goals unless the request is cancelled by $reqr$. One problem with this approach is that an agent will retain a goal even if it believes the goal is impossible to achieve. We address this problem here. We first introduce Shapiro *et al.*'s [2] framework, and then show how it can be changed to better reflect the intuitive interactions between beliefs and goals.

An agent's goals are future oriented. For example, an agent might want some property to hold eventually, i.e., the agent's goal is of the form $\mathbf{Ev}(\psi)$, for some formula ψ. We evaluate formulae such as these with respect to a path of situations rather than a single situation, and we call such formulae *goal formulae*. Cohen and Levesque [3] used infinite time-lines to evaluate such formulae, but for simplicity, we evaluate goal formulae with respect to finite paths of situations which we represent by pairs of situations, (*Now*, *Then*), such that $Now \preceq Then$. *Now* corresponds to the "current time" on the path of situations defined by the sequence of situations in the history of *Then*. Goal formulae may contain two situation constants, *Now* and *Then*. For example, $\exists r.\textsc{InRoom}(\textsc{John}, r, Now) \wedge \neg\textsc{InRoom}(\textsc{John}, r, Then)$ could be used to denote the goal that John eventually leaves the room he is in currently. If ψ is a goal formula then $\psi[s, s']$ denotes the formula that results from substituting s for *Now* and s' for *Then*. When the intended meaning is clear, we may suppress these situation arguments of goal formulae.

Following Cohen and Levesque [3], Shapiro *et al.* model goals using an accessibility relation over possible worlds (situations, in our case). The accessible worlds are the ones that are compatible with what the agent *wants* to be the case. Shapiro *et al.*'s W accessibility relation, like the B relation, is a relation on situations. Intuitively, $W(agt, s', s)$ holds if in situation s, *agt* considers that in s' everything that it wants to be true is actually true. For example, if the agent wants to become a millionaire in a situation S, then in all situations W-related to S, the agent is a millionaire, but these situations can be arbitrarily far in the future.

Following Cohen and Levesque [3], the goals of the agent should be compatible with what it believes. The situations that the agent wants to actualize should be on a path from a situation that the agent considers possible. Therefore, the situations that will be used to determine the goals of an agent will be the W-accessible situations that are also compatible with what the agent believes, in the sense that there is B-accessible situation in their history. We will say that s' $B_{agt,s}$-*intersects* s'' if $B(agt, s'', s)$ and $s'' \preceq s'$. We will suppress *agt* or s if they are understood from the context. Shapiro *et al.* define the goals of *agt* in s to be those formulae that are true in all the situations s' that are W-accessible from s and that B-intersect some situation, s'':

$$\mathbf{Goal}_{SLL}(agt, \psi, s) \stackrel{\text{def}}{=} \forall s', s''.W(agt, s', s) \wedge B(agt, s'', s) \wedge s'' \preceq s' \supset \psi[s'', s'].$$

Note that s'' corresponds to the "current situation" (or the current time) in the path defined by s'. We define a similar accessibility relation C below and define **Goal** in the same way but using C instead of W.

Shapiro *et al.* specify how actions change the goals of agents. They do not give a successor state axiom for W directly, instead they use an auxiliary predicate, Requested. Requested records which goals have been requested of and adopted by an agent, as well as which situations should be dropped from W to accommodate these requests. When a request is cancelled, the corresponding goal (and dropped situations) are removed from the Requested relation. A requested goal is adopted by an agent if the agent does not currently have a conflicting goal. This maintains consistency of goals. $\textsc{Requested}(agt, \psi, s', s)$ holds if some agent has requested that *agt* adopt ψ as a goal in situation s and this request causes *agt* to cease to want situation s'. Here is the successor state axiom for Requested:

Axiom 8

$$\begin{aligned}
&\textsc{Requested}(agt, \psi, s', do(a, s)) \equiv\\
&((\exists reqr.a = \textsc{request}(reqr, agt, \psi) \wedge\\
&\qquad W^-(agt, \psi, a, s', s)) \vee\\
&\quad(\textsc{Requested}(agt, \psi, s', s) \wedge\\
&\quad\neg\exists reqr.a = \textsc{cancelRequest}(reqr, agt, \psi))),
\end{aligned}$$

where W^- is defined below. An agent's goals are expanded when it is requested to do something by another agent. After the $\textsc{request}(requester, agt, \psi)$ action occurs, *agt* should adopt the goal that ψ, unless it currently has a conflicting goal (i.e., we assume agents are maximally cooperative). Therefore, the $\textsc{request}(requester, agt, \psi)$ action should cause *agt* to drop any paths in W where ψ does not hold. This is taken into account in the definition of W^-:

$$\begin{aligned}
&W^-(agt, \psi, a, s', s) \stackrel{\text{def}}{=}\\
&\quad\exists s''.B(agt, s'', s) \wedge s'' \preceq s' \wedge \neg\psi[do(a, s''), s'].
\end{aligned}$$

According to this definition, s' will be dropped from W, due to a request for ψ, if s' B-intersects some s'' such that ψ does not hold on the path $(do(a, s''), s')$. The reason that we check whether $\neg\psi$ holds at $(do(a, s''), s')$ rather than at (s'', s') is to handle goals that are relative to the current time. If, for example, ψ states that the very next action should be to get some coffee, then we need to check whether the next action after the request is getting the coffee. If we checked $\neg\psi$ at (s'', s'), then the next action would be the REQUEST action.

We also have to assert that initially no requests have been made. We do so with the following initial state axiom:

Axiom 9

$$Init(s) \supset \neg\textsc{Requested}(agt, \psi, s', s).$$

Shapiro *et al.* defined W in terms of REQUESTED. s' is W-accessible from s iff there is no outstanding request that caused s' to become inaccessible.

$$W(agt, s', s) \stackrel{\text{def}}{=} \forall\psi.\neg\textsc{Requested}(agt, \psi, s', s))$$

4 Dynamic interactions between goals and beliefs

A common assumption in the agent theory literature [3,4] is that achievement goals that are believed to be impossible to achieve should be dropped. However, we go a step further. If some time later, an agent revises its beliefs and decides that the goal is achievable after all, the agent should reconsider and possibly readopt the goal. Also, the previous focus has been on individual goals that are incompatible with an agent's beliefs. However, it could be the case that each goal is individually compatible with an agent's beliefs but the set of goals of the agent is incompatible, so some of them should be dropped.

First, we make precise the notion of a finite set of goal formulae being compatible with an agent's beliefs. We say that a finite set of goal formulae α is *B-consistent* in situation s, if there exists a path (s'', s') such that s'' is B-accessible from s, and none of the goals in α caused s' to be dropped from W.

$$\begin{aligned} &BCons(agt, \alpha, s) \stackrel{\text{def}}{=} \\ &\quad \exists s', s''.B(agt, s'', s) \wedge s'' \preceq s' \wedge \\ &\qquad\qquad \forall\psi.\psi \in \alpha \supset \neg\textsc{Requested}(agt, \psi, s', s). \end{aligned}$$

If α is a singleton, we may replace it with its element.

To make its goals compatible with its beliefs, an agent takes the set of requested formulae which may be B-inconsistent and chooses a maximally B-consistent set to be its goals. We assume that each agent has a preorder $(\leq)$ over goal formulae corresponding to a prioritization of goals. $\psi \leq \psi'$ indicates that ψ has equal or greater priority than ψ'. This ordering could be used to, e.g., represent that an agent gives different priorities to requests from different sources, or to give higher priority to emergency requests. The agent chooses a maximally B-consistent subset of the requested formulae respecting this ordering. To simplify notation, we fix here a single such ordering for all agents, but in practice different agents will have different orderings, and it is not difficult to generalize the definitions to accommodate this.

Let:

$$reqs(agt, s) \stackrel{\text{def}}{=} \{\psi \mid \exists s' \textsc{Requested}(agt, \psi, s', s)\},$$

denote the set of formulae that have been requested for *agt* in situation s. Since there are no requests initially, and an action adds at most one goal formula to the set of requests, it is easy to see that this set is finite in any situation. Therefore, we can consider the set of requests in a situation to be a list. The list is sorted according to the priority ordering $(\leq)$, using the recursively defined function $sort(\alpha)$, which takes a finite set α and returns a list of elements of α sorted according to $\leq$:[4]

[4] **if** P **then** A **else** B is an abbreviation for $(P \supset A) \wedge (\neg P \supset B)$

Axiom 10

$$\begin{aligned}&\mathit{sort}(\alpha) = l \equiv\\ &\quad \textbf{if } \alpha = \mathit{nil} \textbf{ then } l = \mathit{nil} \textbf{ else}\\ &\qquad l = \mathit{cons}(\mathit{chooseMin}(\alpha),\\ &\qquad\qquad \mathit{sort}(\mathit{remove}(\mathit{chooseMin}(\alpha), \alpha)),\end{aligned}$$

where *chooseMin* is a function which takes a finite set of formulae and returns an element of the set that is minimal in $\leq$:

Axiom 11

$$\mathit{chooseMin}(\alpha) = x \supset \forall y \in \alpha . y \leq x \supset x \leq y.$$

After the set of requests is sorted, a maximally B-consistent sublist is selected that respects the ordering, using the function *makecons*, which is defined using a recursively defined auxiliary function *makecons*′.[5]

Axiom 12

$$\begin{aligned}&\mathit{makecons}'(\mathit{agt}, l, s) = l' \equiv\\ &\quad \textbf{if } l = \mathit{nil} \textbf{ then } l' = \mathit{nil} \textbf{ else}\\ &\qquad \textbf{if } \mathit{BCons}(\mathit{agt}, \mathit{cons}(\mathit{car}(l),\\ &\qquad\qquad \mathit{makecons}'(\mathit{agt}, \mathit{cdr}(l), s)), s) \textbf{ then}\\ &\qquad\quad l' = \mathit{cons}(\mathit{car}(l), \mathit{makecons}'(\mathit{agt}, \mathit{cdr}(l), s)) \textbf{ else}\\ &\qquad\quad l' = \mathit{makecons}'(\mathit{agt}, \mathit{cdr}(l), s).\end{aligned}$$

$$\mathit{makecons}(\mathit{agt}, l, s) = \mathit{reverse}(\mathit{makecons}'(\mathit{agt}, \mathit{reverse}(l), s)).$$

In other words, the list α is checked starting from the end to see if the last element is B-consistent with the result of recursively making the rest of the list B-consistent. If it is B-consistent, then it is added to the result, otherwise it is left out. Finally, the resulting list is reversed to restore the ordering.

This list is used to define our accessibility relation for goals. First, we define $\text{Chosen}(\mathit{agt}, \psi, s', s)$ (in analogy to Shapiro *et al.*'s Requested), which holds if ψ was chosen by *agt* and that choice should cause s' to be dropped from the accessibility relation (i.e., $\text{Requested}(\mathit{agt}, \psi, s', s)$ holds).

$$\begin{aligned}&\text{Chosen}(\mathit{agt}, \psi, s', s) \stackrel{\text{def}}{=}\\ &\quad \mathit{member}(\psi, \mathit{makecons}(\mathit{sort}(\mathit{reqs}(\mathit{agt}, s)), s)) \wedge\\ &\quad \text{Requested}(\mathit{agt}, \psi, s', s).\end{aligned}$$

We define a new accessibility relation for goals, $C(\mathit{agt}, s', s)$, based on the chosen set of goal formulae rather than the requested set. Intuitively, s' is a situation that the agent wants to realize in situation s. We say that $C(\mathit{agt}, s', s)$ holds if s' is a situation that was not caused to be dropped by any chosen goal formula ψ:

$$C(\mathit{agt}, s' s) \stackrel{\text{def}}{=} \forall \psi \neg \text{Chosen}(\mathit{agt}, \psi, s', s).$$

Finally, the goals of the agent are defined analogously to the way it was done by Shapiro *et al.*, but using C instead of W:

$$\begin{aligned}&\mathbf{Goal}(\mathit{agt}, \psi, s) \stackrel{\text{def}}{=}\\ &\quad \forall s', s''. C(\mathit{agt}, s', s) \wedge B(\mathit{agt}, s'', s) \wedge s'' \preceq s' \supset\\ &\qquad\qquad \psi[s'', s'].\end{aligned}$$

[5] A similar function was defined in Booth and Nittka [13]. This way of handling preferences can also be viewed as a special case of [14].

5 Properties

We now consider some properties of goal change. Let Σ consist of the encoding axioms, the axioms defining lists, and Axioms 1–12. Our first result is that the agents' goals are always (simply) consistent.

Theorem 1.

$$\Sigma \models \forall agt, s. \neg \mathbf{Goal}(agt, \mathit{FALSE}, s).$$

As we have discussed, it should be the case that if an agent believes a goal ψ is impossible to achieve then the agent should drop the goal. For this theorem, we assume that ψ is an achievement goal, i.e., of the form eventually ψ' for some goal formula ψ'. The theorem states that if an agent believes that ψ is impossible to achieve, then the agent does not have the goal $\mathbf{Ev}(\psi)$. We need to give a definition for **Ev** to be used both inside the **Bel** operator and the **Goal** operator. Since belief formulae take a situation as an argument and goal formulae take a path as an argument, we need two definitions in order to use them in the two contexts, therefore, we overload the definition.

In the belief context, $\mathbf{Ev}(\psi, s)$ takes a single situation argument. It holds if there exists a path (s'', s') in the future of s such that $\psi[s'', s']$ holds.

$$\mathbf{Ev}(\psi, s) \stackrel{\text{def}}{=} \exists s'', s'. s \preceq s'' \wedge s'' \preceq s' \wedge \psi[s'', s'].$$

In the goal context, $\mathbf{Ev}(\psi, s, s')$ takes a path (a pair of situations) as an argument. It holds if there is a situation s'' in the future of s such that $\psi[s'', s']$ holds.

$$\mathbf{Ev}(\psi, s, s') \stackrel{\text{def}}{=} \exists s''. s \preceq s'' \wedge s'' \preceq s' \wedge \psi[s'', s'].$$

Note that we suppress the situation arguments of **Ev** when it is passed as an argument to **Bel** or **Goal**.

Theorem 2.

$$\begin{aligned}\Sigma \models \forall agt, \psi, s. &\mathbf{Bel}(agt, \neg \mathbf{Ev}(\psi), s) \supset \\ &\neg \mathbf{Goal}(agt, \mathbf{Ev}(\psi), s).\end{aligned}$$

As a corollary, we have a result about belief contraction. If an agent has $\mathbf{Ev}(\psi)$ as a goal in s but after an action a occurs, the agent believes ψ is impossible to achieve, then the agent drops the goal that $\mathbf{Ev}(\psi)$.

Corollary 1.

$$\begin{aligned}\Sigma \models \forall agt, a, \psi, s. &\mathbf{Goal}(agt, \mathbf{Ev}(\psi), s) \wedge \\ &\mathbf{Bel}(agt, \neg \mathbf{Ev}(\psi), do(a, s)) \supset \\ &\neg \mathbf{Goal}(agt, \mathbf{Ev}(\psi), do(a, s)).\end{aligned}$$

We also have a result concerning the expansion of goals. If an agent gets a request for ψ, it will not necessarily adopt ψ as a goal, for example, if it has a conflicting higher priority goal. But if ψ is the highest priority goal formula, and it is B-consistent, it should be adopted as a goal. We say that a goal formula ψ is highest priority among a finite set of goal formulae α, if the priority of ψ is at least as high as the priority of any goal formula in the set, and any goal formula in the set whose priority is at least as high as ψ is equal to ψ.

$$\begin{aligned}\mathit{Hp}(\psi, \alpha) \stackrel{\text{def}}{=} &(\forall \psi' \in \alpha. \psi \leq \psi') \wedge \\ &(\forall \psi'' \in \alpha. \psi'' \leq \psi \supset \psi'' = \psi').\end{aligned}$$

For this theorem, we need an assumption about the belief change framework. Namely, it must be the case that request actions are not "belief producing". More precisely, if a situation s'' is accessible after a request action was executed in situation s, then s'' came about by executing the same request action in a situation s' accessible from s. In other words, successor situations are not dropped from the B relation after a request action is executed.

Axiom 13

$$B(agt, s'', do(\text{REQUEST}(reqr, agt, \psi), s)) \supset$$
$$\exists s'.s'' = do(\text{REQUEST}(reqr, agt, \psi), s') \wedge B(agt, s', s).$$

Theorem 3.

$$\Sigma \cup \{Axiom\ 13\} \models \forall agt, \psi, reqr, s.$$
$$BCons(agt, \psi, do(\text{REQUEST}(reqr, agt, \psi), s)) \wedge$$
$$Hp(\psi, (\{\psi\} \cup reqs(agt, s))) \supset$$
$$\textbf{Goal}(agt, \psi, do(\text{REQUEST}(reqr, agt, \psi), s)).$$

6 Future Work

Another interaction between achievement goals and beliefs is that once an agent believes that an achievement goal has been realized, it should drop that goal. We have not addressed this yet, but it will not be difficult to add it to our framework, as described in the following. In the context of belief change, the agent may believe that a goal has been achieved but later change its mind. In this case, the agent should first drop its achievement goal, but later readopt it after the mind change. Therefore, we need to check whether it is the case that agent believes that an achievement goal ψ has been false continuously since the last request for ψ. If not ψ, should be dropped. This can be formalized in the situation calculus as follows. We first define a predicate $\textbf{Prev}(a, s', s)$, which holds iff the last occurrence of a in the history of s occurs immediately before situation s'.

$$\textbf{Prev}(a, s', s) \stackrel{\text{def}}{=}$$
$$\exists s''.s' = do(a, s'') \wedge s' \preceq s \wedge$$
$$\forall s^*, a^*.s' \prec do(a^*, s^*) \preceq s \supset a \neq a^*.$$

Then, we say that ψ is live in situation s, if the agent believes that ψ has been continuously false since that last request for ψ:

$$Live(\psi, s) \stackrel{\text{def}}{=}$$
$$\textbf{Bel}(agt,$$
$$(\exists s'', reqr.\textbf{Prev}(request(reqr, agt, \psi), s'', Now) \wedge$$
$$\forall s^*, s_1^*.s'' \preceq s^* \preceq Now \wedge s'' \preceq s_1^* \preceq s'' \supset$$
$$\neg\psi[s_1^*, s^*]),$$
$$s).$$

If ψ is believed to have been already achieved, it should be dropped regardless of the agent's other goals and should therefore not be taken into account when determining the maximally B-consistent set. Therefore, for all goals ψ s.t. $\neg Live(\psi, s)$, we remove ψ from $reqs(agt, s)$ before it is passed to *makecons*. This ensures that ψ will not be chosen and will therefore not be a goal of the agent.

7 Conclusion

In this paper, we extended a previous framework for goal change so that agents drop goals that are believed impossible to achieve. To our knowledge this is the first framework to take into account the possibility that the agents change their minds about the impossibility of their goals. When this happens the agents may readopt goals that were previously believed to be impossible to achieve. Some properties about goal consistency, contraction, and expansion were shown. We also sketched how to further extend the framework so that agents will drop achievement goals that are believed to be already achieved. Again, agents might later change their mind about whether the goal has been already achieved and possibly readopt the goal.

8 Acknowledgments

We would like to thank M. David Sadek for suggesting this avenue of research. We also thank Philip R. Cohen, Andreas Herzig, Yves Lespérance, Hector J. Levesque, and Sebastian Sardina for useful discussions pertaining to this paper.

References

1. Shapiro, S., Lespérance, Y., Levesque, H.J.: Goal change. In Kaelbling, L.P., Saffiotti, A., eds.: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05), Denver, CO, Professional Book Center (2005) 582–588
2. Shapiro, S., Lespérance, Y., Levesque, H.J.: Goal change in the situation calculus. Journal of Logic and Computation, to appear (2006)
3. Cohen, P.R., Levesque, H.J.: Intention is choice with commitment. Artificial Intelligence **42** (1990) 213–261
4. Rao, A.S., Georgeff, M.P.: Asymmetry thesis and side-effect problems in linear time and branching time intention logics. In Mylopoulos, J., Reiter, R., eds.: Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91), Morgan Kaufmann (1991) 498–504
5. Bell, J., Huang, Z.: Dynamic goal hierarchies. In Cavedon, L., Rao, A.S., Wobcke, W., eds.: Intelligent Agent Systems, Theoretical and Practical Issues, Based on a Workshop Held at PRICAI'96, Cairns, Australia, August 26-30, 1996. Number 1209 in Lecture Notes in Computer Science, Springer-Verlag (1997) 88–103
6. Reiter, R.: Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems. MIT Press, Cambridge, MA (2001)
7. McCarthy, J., Hayes, P.J.: Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, B., Michie, D., eds.: Machine Intelligence 4. Edinburgh University Press (1969)
8. Levesque, H.J., Pirri, F., Reiter, R.: Foundations for a calculus of situations. Electronic Transactions of AI (ETAI) **2** (1998) 159–178
9. De Giacomo, G., Lespérance, Y., Levesque, H.J.: ConGolog, a concurrent programming language based on the situation calculus. Artificial Intelligence **121** (2000) 109–169
10. Shapiro, S., Pagnucco, M., Lespérance, Y., Levesque, H.J.: Iterated belief change in the situation calculus. In Cohn, A.G., Giunchiglia, F., Selman, B., eds.: Principles of Knowledge Representation and Reasoning: Proceedings of the Seventh International Conference (KR2000), San Francisco, CA, Morgan Kaufmann (2000) 527–538
11. Shapiro, S., Pagnucco, M.: Iterated belief change and exogenous actions in the situation calculus. In López de Mántaras, R., Saitta, L., eds.: Proceedings of the 16th European Conference on Artificial Intelligence (ECAI-04), Amsterdam, IOS Press (2004) 878–882
12. Shapiro, S.: Belief change with noisy sensing and introspection. In Morgenstern, L., Pagnucco, M., eds.: Working Notes of the IJCAI-05 on Nonmonotonic Reasoning, Action, and Change (NRAC 2005). (2005) 84–89
13. Booth, R., Nittka, A.: Reconstructing an agent's epistemic state from observations. In Kaelbling, L.P., Saffiotti, A., eds.: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05), Denver, CO, Professional Book Center (2005) 394–399
14. Brewka, G.: Preferred subtheories: An extended logical framework for default reasoning. In Sridharan, N.S., ed.: Proceedings 11th International Joint Conference on Artificial Intelligence (IJCAI-89), Morgan Kaufmann (1989) 1043–1048