Assignent 3: The monkey's escape
Sequential action scenarios in TAL and SitCalc

Lars Karlsson

February 11, 1998

1 The assignment

This is an assignment about simple, sequential scenarios, which involve some comparison between TAL and situation calculus. You should encode a scenario both in TAL and in situation calculus a la Reiter. In doing the TAL parts, you should use the PMON software, which is available as a java applet at **http://anton.ida.liu.se:8081**. The best way to run it is to set it up as a Marimba channel (Instructions can be found at the site).

The scenario involves the monkey from Green's paper, who is now fed up with life as an AI test animal. Not that it is very painful; it would have been much worse if he was a turkey. Anyhow, the monkey wishes to widen his intellectual horizons. Therefore, the clever monkey plans and performs an escape. The conditions are as follows.

1. The monkey lives in a cage.
2. The cage is locked and can only be opened with the keys.
3. The keys hang on a hook on a wall close to the cage. The monkey can not reach the hook.
4. However, the monkey can reach the hook and take the keys if he uses a stick.
5. Somebody (who obviously is underestimating the intelligence of the monkey) has left a stick in the cage.
6. The cage door exits into a garden, and the garden is surrounded by a fence. The monkey, being a very agile monkey, can climb the fence.
7. Once on the other side of the fence, the monkey is free!

Encode the scenario including the approriate action laws in TAL and in situation calculus, and device a plan for the monkey's escape. For the situation calculus version of the scenario, compute the successor state axioms and use them to prove that the plan works. For TAL, use the VITAL tool to implement the scenario and show that the plan actually works. Present the scenario description, the extension of the *Occlude* predicate on the form $\forall f,t[\mathit{Occlude}(f,t) \equiv \ldots]$, and the preferred models. You can assume that you know exactly when each action starts and ends.

Then extend the TAL version of the scenario as follows.

8. The time it takes to climb the fence depends on whether the fence is high or low. If the fence is high, it takes twice as long time as if the fence is low (you can chose the exact numbers yourself). Further, if the fence is high, there is a risk that the monkey sprains his wrist when jumping down from a high fence.

If we observe that the monkey hasn't got a sprained wrist before the escape but has one after, what conclusion can we draw? Show it.

How can the original and the extended versions of these scenarios be classified using Sandewall's epistemological and ontological taxonomies?

This extension is not possible to implement in situation calculus with successor state axioms, for two reasons. The first reason is obviously the lack of explicit time. What is the second reason? Motivate.

2 Successor state axioms in Situation Calculus

Recall that in situation calculus, there is a function that defines what situation is the result when an action is executed. For instance, $\mathit{Result}(\mathit{paint\text{-}green}(x), s)$ denotes the situation after painting the object x green in situation s. The effects of actions can be defined as in the following example:[1]

$$\begin{aligned}&\mathit{Holds}(\mathit{has\text{-}green\text{-}paint}, s) \Rightarrow\\&\quad \mathit{Holds}(\mathit{green}(x), \mathit{Result}(\mathit{paint\text{-}green}(x), s)) \wedge\\&\quad \neg \mathit{Holds}(\mathit{red}(x), \mathit{Result}(\mathit{paint\text{-}green}(x), s))\end{aligned} \tag{1}$$

$$\begin{aligned}&\mathit{Holds}(\mathit{has\text{-}red\text{-}paint}, s) \Rightarrow\\&\quad \mathit{Holds}(\mathit{red}(x), \mathit{Result}(\mathit{paint\text{-}red}(x), s)) \wedge\\&\quad \neg \mathit{Holds}(\mathit{green}(x), \mathit{Result}(\mathit{paint\text{-}green}(x), s))\end{aligned}$$

Usually, there is also a $Poss(a, s)$ predicate that defines when the action a is possible to execute. In the examples above, for instance, one could have associated the preconditions with the $Poss$ predicate, like in $Poss(\mathit{paint\text{-}green}(x), s) \equiv \mathit{Holds}(\mathit{has\text{-}red\text{-}paint}, s)$, and replaced the precondition with $Poss(\mathit{paint\text{-}green}(x), s) \Rightarrow$ ... above. However, we will not use $Poss$ here.

Given a number of action laws, one can construct the *successor state axiom* for each fluent as follows. Let $\gamma_f^+(a, s)$ be the conditions under which f becomes true in the situation $result(a, s)$ according in the action laws for all different actions a. In our case, we have:

$$\gamma^+_{\mathit{green}(x)}(a, s) = \quad (a = \mathit{paint\text{-}green}(x) \wedge \mathit{Holds}(\mathit{has\text{-}green\text{-}paint}, s)) \tag{2}$$

$$\gamma^+_{\mathit{red}(x)}(a, s) = \quad (a = \mathit{paint\text{-}red}(x) \wedge \mathit{Holds}(\mathit{has\text{-}red\text{-}paint}, s)) \tag{3}$$

Likewise, let $\gamma_f^-(a, s)$ be the conditions under which f becomes false in the situation $result(a, s)$.

$$\gamma^-_{\mathit{green}(x)}(a, s) = \quad (a = \mathit{paint\text{-}red}(x) \wedge \mathit{Holds}(\mathit{has\text{-}red\text{-}paint}, s)) \tag{4}$$

$$\gamma^-_{\mathit{red}(x)}(a, s) = \quad (a = \mathit{paint\text{-}green}(x) \wedge \mathit{Holds}(\mathit{has\text{-}green\text{-}paint}, s)) \tag{5}$$

[1] We assume that all free variables, in this case x and s, are universally quantified.

Now, one can easily construct the successor state axiom for f as follows:

$$Holds(f, Result(a, s)) \equiv [\gamma_f^+(a, s) \vee (Holds(f, s) \wedge \neg\gamma_f^-(a, s))] \tag{6}$$

In short, the successor state axiom states that f holds in $Result(a, s)$ if and only if a is an action that makes f true when executed in s, or if f already was true in s and a is not an action that makes f false when executed in s. The successor state axiom in the example will be as follows.

$$\begin{aligned}&Holds(green(x), Result(a, s)) \equiv \\ &[(a = paint\text{-}green(x) \wedge Holds(has\text{-}green\text{-}paint, s)) \vee \\ &\ (Holds(green(x), s) \wedge \neg(a = paint\text{-}red(x) \wedge Holds(has\text{-}red\text{-}paint, s)))]\end{aligned} \tag{7}$$

$$\begin{aligned}&Holds(red(x), Result(a, s)) \equiv \\ &[(a = paint\text{-}red(x) \wedge Holds(has\text{-}red\text{-}paint, s)) \vee \\ &\ (Holds(red(x), s) \wedge \neg(a = paint\text{-}green(x) \wedge Holds(has\text{-}green\text{-}paint, s)))]\end{aligned} \tag{8}$$