# 3 Situation Calculus

The theory in the previous section is a situation calculus theory. In short, a situation calculus theory consists of

- A set of fluents
- A set of actions
- Situations, with $S_0$ is a constant denoting the initial situation
- Function $Do$ that maps a pair of an action and a situation into a situation; $Do(a, s)$ denotes the successor situation to $s$ resulting from executing action $a$
- Predicate $Holds$ whose first parameter is a fluent and second parameter is a situation
- A set of axioms about the initial situation (what is true/false in the initial situation? e.g. $Holds(At(Agent, [1, 1]), S_0)$ etc.)
- A set of axioms that describes the effects of actions
- A set of axioms that describes the precondition of actions; for each action $a$, the theory consists of one formula of the form $Poss(a, s) \leftrightarrow Holds(\varphi, s)$ where $\varphi$ is a fluent formula.

Given a situation calculus theory (which is essential a set of first order axioms) – under certain conditions we can prove (or predict) what will be true/false after executing a sequence of actions in a situation. This is called the *projection problem.* We next discuss the assumptions needed in solving the projection problem. But first, for simplicity, let us leave the complicated 'Wumpus' world and describe the 'Block' world which will be used as the example for the discussion subsequently.

## 3.1 The Block World

Let us consider the block world in the following picture.

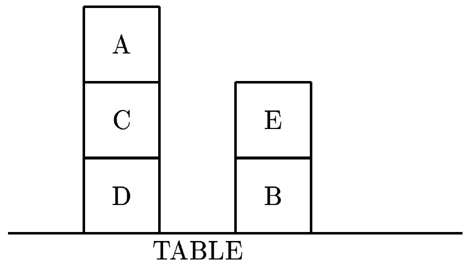


Figure 1: A Block World

We assume that the world consists of the above blocks and an agent who can pick up block and put it down on the table or on top of other blocks which are clear.

Objects of the block world are the blocks, the table, and the agent. The blocks are described by their names, $a,b,c,\ldots$. The table will be denoted by $t$. Since there is only one agent and no actions of the environment (for example, the agent will not die in this world - there is no 'Wumpus' or 'Pit'!), we do not need to have a constant to denote the agent.

We will use the predicate $Block(x)$ to denote that object $x$ is a block.

The actions of the agents are:

1. $Pickup(x)$ - pickup block $x$,
2. $Put(x, y)$ - put block $x$ on top of block $y$, and
3. $Put(x, T)$ - put block $x$ on the table.

The fluents of the block world are:

1. $On(x, y)$ - block $x$ is on the block $y$,
2. $On(x, T)$ - block $x$ is on the table,
3. $Holding(x)$ - (the agent) is holding block $x$, and
4. $Clear(x)$ - block $x$ is clear.

We note that

$$Clear(x) \leftrightarrow \neg\forall y(Block(y) \land \neg On(y, x))$$

which means that some properties of the world can be automatically deduced when we know about other properties.

The initial situation can be described by the following axioms

$$Holds(On(A, C), S_0) \tag{14}$$
$$Holds(On(C, D), S_0) \tag{15}$$
$$Holds(On(D, T), S_0) \tag{16}$$
$$Holds(On(E, B), S_0) \tag{17}$$
$$Holds(On(B, T), S_0) \tag{18}$$

Now, let us describe the set of precondition axioms and the set of effect axioms.

**Precondition axioms:** Before an action is executed, it requires that some conditions are true. This condition is called the action's *precondition*. For example, to put down a block $x$ on top of block $y$, the agent must hold $x$ and $y$ must be clear. Another example is that to pick up a block $x$, the agent must not hold another block and $x$ must be clear etc... We represent these by

$$Poss(Pickup(x), s) \rightarrow \forall y(Holds(\neg Holding(y), s)) \wedge Holds(Clear(x), s) \tag{19}$$
$$Poss(Put(x, y), s) \rightarrow Holds(Holding(x), s) \wedge Holds(Clear(y), s) \tag{20}$$
$$Poss(Put(x, T), s) \rightarrow Holds(Holding(x), s) \tag{21}$$

**Problem** The above formulas do not allow us to prove when an action, say $Pickup(x)$, is possible. I.e., when $Poss(Pickup(x), s)$ is true. Remember how to prove whether a literal is true?

Can we reverse the axioms (change $\rightarrow$ with $\leftarrow$) and then use them to prove when $Poss(Pickup(x), s)$ is true? No, this will not work since it is not correct! There might be conditions which we have not considered? For example, the block can be too heavy, the agent might be tired etc... So, we need to specify all of these conditions, called *qualifications*. But there are infinitely many conditions like this?

The above is called the *qualification problem*. It calls for the need for *nonmonotonic reasoning* in AI.

To overcome the qualification problem, we make the assumption that the antecedent of the above formula represents the sufficient and necessary conditions for the action to be executed. The *final* action precondition axioms for the above three actions are:

$$Poss(Pickup(x), s) \leftrightarrow \forall y(Holds(\neg Holding(y), s)) \wedge Holds(Clear(x), s) \tag{22}$$
$$Poss(Put(x, y), s) \leftrightarrow Holds(Holding(x), s) \wedge Holds(Clear(y), s) \tag{23}$$
$$Poss(Put(x, T), s) \leftrightarrow Holds(Holding(x), s) \tag{24}$$

We will use the above axioms instead of the axioms (19)-(21).

Given the above axioms and some axioms about the , we can now deduce whether the action $Pickup(A)$ is possible in the initial situation, i.e., $Poss(Pickup(A), S_0)$ is true. How?

**Effect axioms:** Actions have effects. For example, if the agent pickups a block he will hold it in the successor situation.

$$Poss(Pickup(x), s) \rightarrow Holds(Holding(x), Do(Pickup(x), s)) \tag{25}$$

$$Poss(Put(x, y), s) \rightarrow Holds(On(x, y), Do(Put(x, y), s)) \tag{26}$$

$$Poss(Put(x, T), s) \rightarrow Holds(On(x, T), Do(Put(x, T), s)) \tag{27}$$

$$\begin{aligned} Poss(Pickup(x), s) \rightarrow [&Holds(On(x, y), s) \\ &\rightarrow Holds(\neg On(x, y), Do(Pickup(x), s))] \end{aligned} \tag{28}$$

But, the above equations only represent the 'positive effects' of actions (what becomes true) and 'negative effects' of actions (what becomes false) as well. In (28), for example, if the agent pickups block $x$ which is on a block $y$ in the situation $s$ then $On(x, y)$ will not hold in $Do(Pickup(x), s)$; or if the agent puts block $x$ on a block $y$ in the situation $s$ then $Clear(y)$ will not hold in $Do(Put(x, y), s)$, etc. **Try to complete the set of effect axioms.**

Furthermore, there are fluents whose values do not change after the execution of an action. For example, if $x$ is on $y$ in situation $s$ ($Holds(On(x, y), s)$ is true) and the agent puts block $z$ on top of $x$, then $x$ is still on $y$ in situation $Do(Put(z, x), s)$ ($Holds(On(x, y), Do(Put(z, x), s))$ is true). Representing the unchange effects of actions require axioms of the form

$$Holds(On(x, y), s) \rightarrow Holds(On(x, y), Do(Pickup(z), s))$$

These axioms are called *frame axioms.*

If we have $n$ actions, $m$ fluents, we would need $2 \times n \times m$ frame axioms. This is too many!!!

Representing frame axioms in a compact way is a challenge for quite sometime. To date, there are many solutions to the frame problems. Under reasonable assumptions, we can write, for each fluent $f$, one *successor state axiom* in the following form

$$Holds(f, Do(a, s)) \leftrightarrow \gamma_f^+(a, s) \vee (Holds(f, s) \wedge \gamma_f^-(a, s)) \tag{29}$$

where $\gamma_f^+(a, s)$ (resp. $\gamma_f^-(a, s)$) summarizes the conditions for the fluent $f$ to become true (resp. false).

For example, in the block world we have

$$Holds(On(x, y), Do(a, s)) \leftrightarrow \gamma^+_{On(x,y)}(a, s) \vee (Holds(On(x, y), s) \wedge \gamma^-_{On(x,y)}(a, s)) \tag{30}$$

where

$\gamma^+_{On(x,y)}(a, s) = (a = Put(x, y)) \wedge Poss(a, s)$ and

$\gamma^-_{On(x,y)}(a, s) = (a = Pickup(x)) \wedge Poss(a, s)$.

**Complete the description!**

## 3.2 Projection Problem

Let us denote the situation calculus theory for the block world by $\mathcal{T}$. We can prove that

$$\mathcal{T} \models Holds(On(A, C), S_0)$$

$$\mathcal{T} \models Holds(\neg On(A, C), Do(Pickup(A), S_0))$$

In general, we can prove that

$$\mathcal{T} \models Holds(f, Do(a_n, \ldots, Do(a_1, S_0))) \tag{31}$$

for a fluent $f$ and a sequence of actions $a_1, \ldots, a_n$.

We also write

$$\mathcal{T} \models Holds(f, Do(\alpha, S_0))) \tag{32}$$

$\alpha = [a_1, \ldots, a_n]$.

## 3.3 Planning in Situation Calculus

The projection shows that we could formulate a planning task by asking for a sequence of actions ($\alpha$) that makes the goal (say, a fluent $f$) true in the resulting situation of executing $\alpha$ in $S_0$. That is, for a planning task of achieving the goal $\varphi$ (a fluent formula), we ask for a sequence of actions $\alpha$ such that

$$\mathcal{T} \models Holds(\varphi, Do(\alpha, S_0))) \tag{33}$$

**Example 3.1** *Find a sequence of actions that achieves the goal of having A on D. That is, we need to find* $\alpha$ *such that* $\mathcal{T} \models Holds(On(A, D), Do(\alpha, S_0))$.

*It is easy to see that one of the possibility is*

$\alpha = [Put(A, E), Put(C, T), Put(A, D)]$. **Why?**

*Can you find another sequence of actions?*

Next: How to find a plan?