Informatics 2D · Agents and Reasoning · 2019/2020

Lecture 14 · Situation Calculus

Claudia Chirita

School of Informatics, University of Edinburgh

13^{th} February 2020

Based on slides by: Jacques Fleuriot, Michael Rovatsos, Michael Herrmann, Vaishak Belle

Outline

- Planning
- Situations
- Frame problem

Using Logic to Plan

We need ways of representing

- the world
- the goal
- how actions change the world

We haven't said much about changing the world.

Difficulty After an action, new things are true, and some previously true facts are no longer true.

Situations

Situations extend the concept of a state by additional logical terms

- consist of initial situation (usually called S_0) and all situations generated by applying an action to a situation

Providing facts about situations

- by relating predicates to situations
  e.g. instead of saying just $On(A, B)$, say (somehow) $On(A, B)$ in situation S_0

Actions are thus

- performed in a situation, and
- produce new situations with new facts
- e.g. Forward and Turn(Right)

Representing Predicates Relative to a Situation

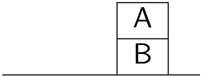
1. Can add an argument for a situation to each predicate that can change.
   - e.g. instead of $\mathtt{On}(A, B)$, write $\mathtt{On}(A, B, S_0)$

2. Alternatively, introduce a predicate Holds
   - $\mathtt{On}$ etc. become functions
   - e.g. $\mathrm{Holds}(\mathtt{On}(A, B), S_0)$
   - What do things like $\mathtt{On}(A, B)$ mean now?
     A set of situations in which A is on B.

How This Will Work

Before some action, we might have in our KB

- $\text{On}(A, B, S_0)$
- $\text{On}(B, \text{Table}, S_0)$



After an action that moves A to the table, say, we add

- $\text{Clear}(B, S_1)$
- $\text{On}(A, \text{Table}, S_1)$

All these propositions are true. We have dealt with the issue of change, by keeping track of what is true when.

Same Thing, Slightly Different Notation

Before

- $\text{Holds}(\text{On}(A, B), S_0)$
- $\text{Holds}(\text{On}(B, \text{Table}), S_0)$

After, add

- $\text{Holds}(\text{Clear}(B), S_1)$
- $\text{Holds}(\text{On}(A, \text{Table}), S_1)$

Representing Actions

We need to represent

- results of doing an action
- conditions that need to be in place to perform an action

For convenience, we will define functions to abbreviate actions

- e.g. $\mathrm{Move}(A, B)$ denotes the action type of moving A onto B
- these are action types, because actions themselves are specific to time, etc.

Now, introduce a function Result, designating "the situation resulting from doing an action type in some situation".

- e.g. $\mathrm{Result}(\mathrm{Move}(A, B), S_0)$ means "the situation resulting from doing an action of type $\mathrm{Move}(A, B)$ in situation S_0".

How This Works

Keep in mind that things like

$\text{Result}(\text{Move}(A, B), S_0)$

are terms and denote situations.
They can appear anywhere we would expect a situation.

So we can say things like

$S_1 = \text{Result}(\text{Move}(A, B), S_0)$
$\text{On}(A, B, \text{Result}(\text{Move}(A, B), S_0)) \equiv \text{On}(A, B, S_1)$

Alternatively,

$\text{Holds}(\text{On}(A, B), \text{Result}(\text{Move}(A, B), S_0))$

Axiomatising Actions

We can describe the results of actions, together with their preconditions.

e.g. "If nothing is on x and y, then one can move x to on top of y, in which case x will then be on y."

$\forall x, y, s. \mathrm{Clear}(x, s) \land \mathrm{Clear}(y, s) \rightarrow \mathrm{On}(x, y, \mathrm{Result}(\mathrm{Move}(x, y), s))$

Alternatively,

$$\forall x, y, s. \mathrm{Holds}(\mathrm{Clear}(x), s) \land \mathrm{Holds}(\mathrm{Clear}(y), s) \\ \rightarrow \mathrm{Holds}(\mathrm{On}(x, y), \mathrm{Result}(\mathrm{Move}(x, y), s))$$

This is an effect axiom.
It includes a precondition as well.

Situation Calculus

This approach is called the situation calculus.

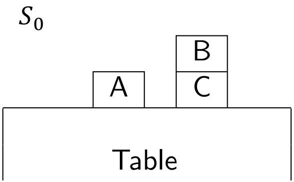
We axiomatise all our actions, then use a general theorem prover to prove that a situation exists in which our goal is true.

The actions in the proof would comprise our plan.

Example

KB

$\mathrm{On}(A, \mathrm{Table}, S_0)$
$\mathrm{On}(B, C, S_0)$
$\mathrm{On}(C, \mathrm{Table}, S_0)$
$\mathrm{Clear}(A, S_0)$
$\mathrm{Clear}(B, S_0)$
$+$ axioms about actions



Goal

$\exists s'.\mathrm{On}(A, B, s')$

What happens?

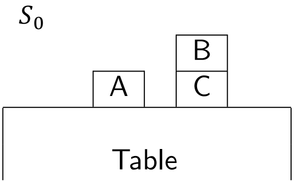
- We want to prove $\mathtt{On}(A, B, s')$ for some s'.
  - Find axiom
    $$\forall x, y, s. \text{Clear}(x, s) \land \text{Clear}(y, s) \\ \rightarrow \mathtt{On}(x, y, \text{Result}(\text{Move}(x, y), s))$$
  - Goal would be true if we could prove
    $\text{Clear}(A, s) \land \text{Clear}(B, s)$ by backward chaining.
  - But both are true in S_0, so we can conclude
    $\mathtt{On}(A, B, \text{Result}(\text{Move}(A, B), S_0))$
- We are done!

  We look at the proof and see only one action, $\text{Move}(A, B)$, which is executed in situation S_0, so this is our plan.

Example · Same Situation, Harder Goal[1]

KB

$\mathrm{On}(A, \mathrm{Table}, S_0)$
$\mathrm{On}(B, C, S_0)$
$\mathrm{On}(C, \mathrm{Table}, S_0)$
$\mathrm{Clear}(A, S_0)$
$\mathrm{Clear}(B, S_0)$
+ axioms about actions



Goal

$\exists s'.\mathrm{On}(A, B, s') \land \mathrm{On}(B, C, s')$

[1] It's not really harder, B is already on C, and we just showed how to put A on B.

With Goal $\text{On}(A, B, s') \wedge \text{On}(B, C, s')$

- Suppose we try to prove the first subgoal, $\text{On}(A, B, s')$.
  - Use same axiom
    $$\forall x, y, s. \text{Clear}(x, s) \wedge \text{Clear}(y, s) \rightarrow \text{On}(x, y, \text{Result}(\text{Move}(x, y), s))$$
  - Again, by chaining, we can conclude $\text{On}(A, B, \text{Result}(\text{Move}(A, B), S_0))$
  - Abbreviating $\text{Result}(\text{Move}(A, B), S_0)$ as S_1, we have $\text{On}(A, B, S_1)$.
- Substituting S_1 for s' in our other subgoal makes that $\text{On}(B, C, S_1)$. If this were true, we are done.
- But we have no reason to believe this is true!
- Sure, $\text{On}(B, C, S_0)$, but how does the planner know this is still true, i.e., $\text{On}(B, C, S_1)$?
- In fact, it doesn't, so it fails to find an answer!

The Frame Problem

We have failed to express the fact that everything that isn't changed by an action should really stay the same.

Can fix by adding frame axioms.

$\forall x, s.\mathrm{Clear}(x, s) \rightarrow \mathrm{Clear}(x, \mathrm{Result}(\mathrm{Paint}(x), s))$

...

There are lots of these!

Is this a big problem?

Better Frame Axioms

- Can fix with neater formulation:

$$\forall x, y, s, a. \mathrm{On}(x, y, s) \wedge (\forall z. a = \mathrm{Move}(x, z) \rightarrow y = z) \\ \rightarrow \mathrm{On}(x, y, \mathrm{Result}(a, s))$$

- Can combine with effect axioms to get successor-state axioms:

$$\forall x, y, s, a. \mathrm{On}(x, y, \mathrm{Result}(a, s)) \leftrightarrow \\ \mathrm{On}(x, y, s) \wedge (\forall z. a = \mathrm{Move}(x, z) \rightarrow y = z) \\ \vee (\mathrm{Clear}(x, s) \wedge \mathrm{Clear}(y, s) \wedge a = \mathrm{Move}(x, y))$$

How Does This Help Our Example?

- We want to prove $\mathrm{On}(B, C, \mathrm{Result}(\mathrm{Move}(A, B), S_0))$ given that $\mathrm{On}(B, C, S_0)$.
- Axiom says $\forall x, y, s, a. \mathrm{On}(x, y, \mathrm{Result}(a, s)) \leftrightarrow$
  $\mathrm{On}(x, y, s) \wedge (\forall z. a = \mathrm{Move}(x, z) \rightarrow y = z)$
  $\vee (\mathrm{Clear}(x, s) \wedge \mathrm{Clear}(y, s) \rightarrow a = \mathrm{Move}(x, y))$
- So we need to show
  $\mathrm{On}(B, C, S_0) \wedge (\forall z. \mathrm{Move}(A, B) = \mathrm{Move}(B, z) \rightarrow C = z)$ is true:
  - The first conjunct is in the KB.
  - The second one is true since actions are the same iff they have the same name and involve the exact same objects*:
    $A(x_1, \dots, x_m) = A(y_1, \dots, y_m)$ iff $x_1 = y_1, \dots, x_m = y_m$.
    So $\mathrm{Move}(A, B) = \mathrm{Move}(B, z)$ is false.

*Another assumption in KB: $A(x_1, \dots, x_m) \neq B(y_1, \dots, y_n)$.

These are known as Unique Action Axioms.

Refutation Theorem Proving
(Dual) Skolemisation

Suppose $\forall x.\exists y.G(x, y)$ is the goal in resolution refutation.

We need to negate the goal:

$\neg\forall x.\exists y.G(x, y) \equiv \exists x.\forall y.\neg G(x, y)$

Then skolemise (i.e drop the existential quantifier):

$\neg G(X_0, y)$

Intuition

y is to be unified to construct witness.
X_0 must not be instantiated.

KB and Axioms as Clauses

Variables a, x, y, z, s

Constants A, B, C, S_0

Initial State

$\mathrm{On}(A, \mathrm{Table}, S_0)$
$\mathrm{On}(B, C, S_0)$
$\mathrm{On}(C, \mathrm{Tabl}e, S_0)$
$\mathrm{Clear}(A, S_0)$
$\mathrm{Clear}(B, S_0)$

(neg.) Goal

$\neg\mathrm{On}(A, B, s') \vee \neg\mathrm{On}(B, C, s')$

KB and Axioms as Clauses

Effect Axiom

$\neg \text{Clear}(x, s) \lor \neg \text{Clear}(y, s) \lor \text{On}(x, y, \text{Result}(\text{Move}(x, y), s))$

Skolem function
↙

Frame Axioms

$\neg \text{On}(x, y, s) \lor a = \text{Move}(x, Z(x, y, s, a)) \lor \text{On}(x, y, \text{Result}(a, s))$

$\neg \text{On}(x, y, s) \lor \neg y = Z(x, y, z, s, a) \lor \text{On}(x, y, \text{Result}(a, s))$

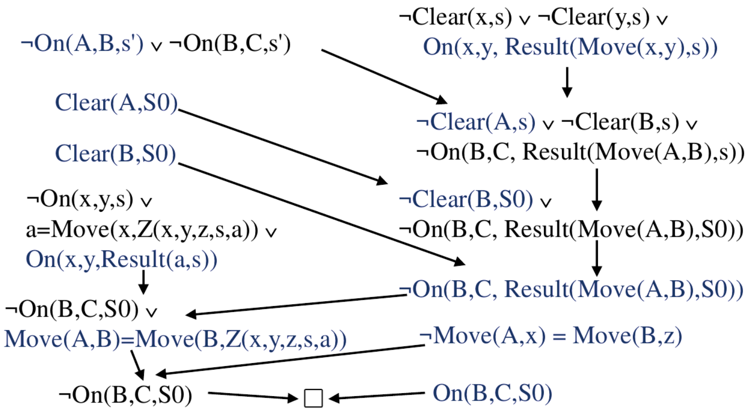
Unique Action Axioms

$\neg \text{Move}(A, B) = \text{Move}(B, z)$

Unique Name Axiom

$\neg C_i = C_j$ for every pair of distinct constants C_i and C_j in KB.

Resolution Refutation



Frame problem partially solved

- This solves the representational part of the frame problem.
- Still have to compute that everything that was true and wasn't changed is still true.
- Inefficient (as is general theorem proving).
- Solution: Special purpose representations, special purpose algorithms, called planners.

Summary

- Planning
- Situations
- Frame problem