Reasoning About Action

Many of the environments about which we wish to reason are dynamic in nature. For example, consider a robot whose job it is to deliver packages in an office environment. In this because we shall consider some of the issues that arise when reasoning about a changing world.

Reasoning About Action

| Situations
| States/Situations
| Domain Constraints
| Actions
| The Frame Problem
| Morals

Where to next?

Reasoning About Action
Reasoning About Action

One method to reason about action is to simply change the agent's knowledge base. Erase some sentence(s) that should no longer be true and add sentences that will now be true (i.e., after performing an action). However, we can only answer questions about the current state. The state of the world may change since we have not considered the past or future.

Situational Calculus

Situation calculus is a way of describing change in first-order logic. A situation is a snapshot of the world at a particular point in time. Situations include the state of the world (i.e., the current state) and the time.

The Blocks World

The blocks world is a famous AI domain. In this domain, there are blocks that can be stacked on each other. We shall consider one of the more famous AI domains — the blocks world.

Table

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The Blocks World

The blocks world is a famous AI domain. In this domain, there are blocks that can be stacked on each other. We shall consider one of the more famous AI domains — the blocks world.

Reasoning About Action

Can you think of anything else? Constraints on legal scenarios. Other actions that change state of the world and when changes they occur. The state of the world is a snapshot of the world at a particular point in time.
There are of course other ways to formulate this scenario. Another common way uses the situation calculus, as follows:

\[
\begin{align*}
\text{if } x \text{ is a block and there is another block on it, then } y \text{ is not clear} \quad &\forall x, y, z \left( \text{on}(x, y) \land \text{on}(y, z) \rightarrow \neg \text{clear}(z) \right) \\
\text{if } x \text{ is clear and there is no block on top of it} \quad &\forall x, y \left( \neg \text{clear}(x) \rightarrow \forall y \left( \neg \text{on}(x, y) \right) \right) \\
\end{align*}
\]

Also known as state constraints.

**Domain Constraints**

Also known as state constraints.

**Effect Axioms for clear**

Positive effect axiom for clear:

\[
\begin{align*}
&\forall x, y, z \left( \text{on}(x, y) \land \text{on}(y, z) \rightarrow \text{clear}(z) \right) \\
&\text{Negative effect axiom for clear}:
\end{align*}
\]

We need positive and negative effect axioms for every fluent (predicate) that can change value.

We need positive and negative effect axioms for every fluent (predicate).

How many such axioms do you need to specify?

Methods:

- **Method 2:**

Positive effect axiom for clear:

\[
\begin{align*}
&\forall x, y, z \left( \text{on}(x, y) \land \text{on}(y, z) \rightarrow \text{clear}(z) \right) \\
&\text{Negative effect axiom for clear}:
\end{align*}
\]

We will need to specify, when each action does to the world.

For example, "move block x from y to z", and "clear x".

\[
\begin{align*}
&\forall x, y, z \left( \text{on}(x, y) \land \text{on}(y, z) \rightarrow \text{clear}(z) \right) \\
&\forall x, y, z \left( \text{on}(x, y) \land \text{on}(y, z) \rightarrow \text{clear}(z) \right)
\end{align*}
\]

In fact, do is a function from a situation to the result of performing (some) action y in situation x (if that is possible).

In terms of do, we have action do in situation S, which is the result of

\[
\begin{align*}
&\text{do} (\forall y \left( \text{on}(x, y) \rightarrow \neg \text{clear}(x) \right) \\
&\text{Another common way to use the situation calculus is called}
\end{align*}
\]

**Actions**
The Frame Problem

Action descriptions are not complete:
They describe what changes
They do not specify what stays the same
Action descriptions are not complete

The famous Frame Problem:
The problem of characterizing those aspects of the state
can be (e.g.)
Particular effects are very difficult (in principle)

Ramification Problem

Ramification Problem

can be (e.g.)
Particular effects are very difficult (in principle)
The list of preconditions
action and their effects

Qualification Problem

Qualification Problem

can be (e.g.)
Particular effects are very difficult (in principle)

Summary

Summary

Reasoning about actions is a very interesting area of artificial intelligence and often makes use of belief change and nonmonotonic reasoning.

LSS 2000, Friday 15 December, 2000
Applied Logic I: Reasoning about Action
12

Reasoning with the Situation Calculus

Reasoning with the Situation Calculus

If we would like to determine a plan to achieve goal \( T(s,') \), prove

\[
\text{Cred: } \neg \text{on}(g, \text{Table}').
\]

\( \text{Goal: } \neg \text{on}(g, \text{Table}'). \)

\( \text{Given: } \text{on}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

One solution — Frame Axioms

description that are not changed by an action
The problem of characterizing those aspects of the state

The Frame Problem

They do not specify what stays the same
They describe what changes
Action descriptions are not complete

<table>
<thead>
<tr>
<th>Cred:</th>
<th>on(g, Table')</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal:</td>
<td>( \neg \text{on}(g, \text{Table}') )</td>
</tr>
</tbody>
</table>

Chopra, Samir and Pagnucco, Maurice, 2000
Generated: 28 July 2000

<table>
<thead>
<tr>
<th>Goal:</th>
<th>( \neg \text{on}(g, \text{Table}') )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cred:</td>
<td>on(g, Table')</td>
</tr>
</tbody>
</table>

Ramification Problem

Ramification Problem

What are the ramifications (direct and indirect effects) of performing an action

Qualification Problem

Qualification Problem

Recent approaches have investigated the use of explicit notions of causality in an attempt to solve this problem efficiently

<table>
<thead>
<tr>
<th>Goal:</th>
<th>( \neg \text{on}(g, \text{Table}') )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cred:</td>
<td>on(g, Table')</td>
</tr>
</tbody>
</table>

Ramification Problem

Ramification Problem

What are the ramifications (direct and indirect effects) of performing an action

Qualification Problem

Qualification Problem

Recent approaches have investigated the use of explicit notions of causality in an attempt to solve this problem efficiently

<table>
<thead>
<tr>
<th>Goal:</th>
<th>( \neg \text{on}(g, \text{Table}') )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cred:</td>
<td>on(g, Table')</td>
</tr>
</tbody>
</table>

Reasoning with the Situation Calculus

Reasoning with the Situation Calculus

If we would like to determine a plan to achieve goal \( T(s,') \), prove

\[
\text{Cred: } \neg \text{on}(g, \text{Table}').
\]

\( \text{Goal: } \neg \text{on}(g, \text{Table}'). \)

\( \text{Given: } \text{on}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

One solution — Frame Axioms

description that are not changed by an action
The problem of characterizing those aspects of the state

The Frame Problem

They do not specify what stays the same
They describe what changes
Action descriptions are not complete

<table>
<thead>
<tr>
<th>Cred:</th>
<th>on(g, Table')</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal:</td>
<td>( \neg \text{on}(g, \text{Table}') )</td>
</tr>
</tbody>
</table>

Ramification Problem

Ramification Problem

What are the ramifications (direct and indirect effects) of performing an action

Qualification Problem

Qualification Problem

Recent approaches have investigated the use of explicit notions of causality in an attempt to solve this problem efficiently

<table>
<thead>
<tr>
<th>Goal:</th>
<th>( \neg \text{on}(g, \text{Table}') )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cred:</td>
<td>on(g, Table')</td>
</tr>
</tbody>
</table>

Reasoning with the Situation Calculus

Reasoning with the Situation Calculus

If we would like to determine a plan to achieve goal \( T(s,') \), prove

\[
\text{Cred: } \neg \text{on}(g, \text{Table}').
\]

\( \text{Goal: } \neg \text{on}(g, \text{Table}'). \)

\( \text{Given: } \text{on}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{clear}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

\( \text{on}(g, \text{Table}'). \)

One solution — Frame Axioms

description that are not changed by an action
The problem of characterizing those aspects of the state

The Frame Problem

They do not specify what stays the same
They describe what changes
Action descriptions are not complete

<table>
<thead>
<tr>
<th>Cred:</th>
<th>on(g, Table')</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal:</td>
<td>( \neg \text{on}(g, \text{Table}') )</td>
</tr>
</tbody>
</table>
Some Morals?

The way you use logic is important. It can lead to more complex structures and beliefs. For example, using AGM belief change operators can lead to more refined and sophisticated reasoning about knowledge. This is important in fields like philosophy, artificial intelligence, and computer science.

Where to Next?

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change

Belief Change


Raymond Reiter, The Frame Problem in the Situation Calculus: A Simple Solution (Sometimes), and a Completeness Result for Central Simple Solution (Sometimes) and a Completeness Result for Central Knowledge about Action, in Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy, pp 359-380.

Raymond Reiter, The Frame Problem in the Situation Calculus: A Simple Solution (Sometimes) and a Completeness Result for Central Simple Solution (Sometimes) and a Completeness Result for Central Knowledge about Action, in Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy, pp 359-380.


