Knowledge, action, and the frame problem

Reasoning about Actions

Francesco Petri, Flavio Maiorana and Charlotte Primiceri

May 28, 2024

Sapienza Università di Roma

Introduction

Table of Contents

Situation calculus provides a framework for reasoning about actions.
This work presents an expansion to handle the *knowledge* possessed or acquired by the agent, and allow it to shape the agent's decisions.

- Knowledge is represented by one additional fluent
- Uniform axiomatization with the rest of sitcalc
- Ordinary actions and knowledge-producing ones are strictly separated
- Easy expansion of regression as defined in [Reiter2001]
- Desirable properties are direct consequences of the axiomatization (e.g. knowledge persistence / memory)

Opzionale
Un paio di azioni ordinarie e un paio di azioni di conoscenza di esempio, giusto per inquadrare il discorso

The K fluent

Knowledge as a fluent

$$K(s', s)$$

Defines an accessibility relation between situations.

(Informal) definition

$K(s', s)$ is true if an agent in situation s could mistake the current situation for the other s', given its current knowledge.

Knowledge

Knowledge as a fluent

Definition of knowledge

A fluent is known to be true (false) in a situation s if and only if it is true (false) in all situations accessible from s.

Shorthand notation: $\textbf{Knows}(\phi, s) \stackrel{\text{def}}{=} \forall s'\, \mathrm{K}(s', s) \rightarrow \phi(s')$

Knowledge-producing actions

Knowledge as a fluent

Actions that have an effect on the agent's knowledge

SENSE actions

Learn the truth value of a formula. Example: check if a door is open or closed.

$$\textbf{Knows}(\mathrm{P}, \mathrm{DO}(\mathrm{SENSE}_{\mathsf{P}}, s)) \vee \textbf{Knows}(\neg \mathrm{P}, \mathrm{DO}(\mathrm{SENSE}_{\mathsf{P}}, s))$$

READ actions

Learn the value of a term. Example: read a number on a sheet of paper.

$$\exists x\, \textbf{Knows}(\tau = x, \mathrm{DO}(\mathrm{READ}_{\tau}, s))$$

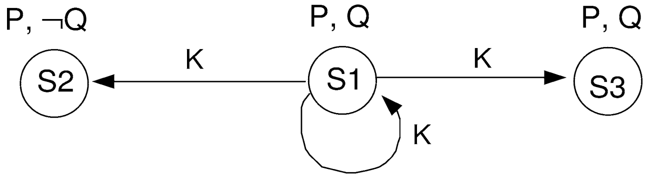
Assumption: ordinary and knowledge-producing actions are strictly separated.

Towards a successor state axiom for K

Knowledge effects

In order to complete the specification of the K fluent, we need to define its successor state axiom, determining how ordinary actions and knowledge-producing actions affect it.

Consider this case study with three accessible situations. The agent is in S1.



$$\textbf{Knows}(\mathrm{P}, S1) \wedge \neg\textbf{Knows}(\mathrm{Q}, S1)$$

Ordinary actions

Knowledge effects

Pippo

Knowledge-producing actions
Knowledge effects

Pippo

The successor state axiom for K

Knowledge effects

Pippo

<varie ed eventuali>
Knowledge effects

Pippo

Table of Contents

The Problem

- A robot has to manage some plants in a garden.
- The robot can perform an action on one plant at a time and only if it's near the plant.
- A plant can be watered only if it is dry. The plant's humidity can be inspected.
- The robot has a water tank with unlimited capacity.

Fluents and Actions

Axiomatization

Fluents

- $Near(x, s) \rightarrow$ Robot is near plant x in situation s
- $Humidity(x, s) \rightarrow$ Humidity of plant x in situation s
- $Infested(x, s) \rightarrow$ Plant x is infested by bugs in situation s
- $Healthy(x, s) \rightarrow$ Plant x is healthy in situation s

Actions

- $Treat(x, s) \rightarrow$ Treat plant x in situation s
- $Water(x, s) \rightarrow$ Water plant x in situation s
- $GoTo(x) \rightarrow$ Go to plant x

Successor State Axioms

Axiomatization

In general $F(x, do(\alpha, s)) \equiv \Phi_F^+(x, a, s) \vee (F(x, s) \wedge \neg\Phi_F^-(x, a, s))$

- $Near(x, do(\alpha, s)) \equiv \alpha = GoTo(x) \vee (Near(x, s) \wedge \neg\exists y.\alpha = GoTo(y))$
- $Healthy$?
- $Humidity(x, do(\alpha, s)) = h \equiv Humidity(x, s) = h$
- $Infested(x, do(\alpha, s)) = h \equiv Infested(x, s) = i$

Initial State

Axiomatization

-

Precondition Axioms

Axiomatization

Sensing Result Axioms

Axiomatization

Boh

Considerations on theorems

Boh

Golog

Table of Contents

Knowledge, action, and the frame problem

Thank you for listening!
Any questions?