Informatics 2D: Reasoning and Agents

Alex Lascarides

School of informatics

The University of Edinburgh

Lecture 16a: Introduction to Planning

Where are we?

The first two blocks of the course dealt with . . .

- Basic notions of agency
- Intelligent problem-solving
- Heuristic search, constraints
- Logic & logical reasoning
- Reasoning about actions and time

In the remainder of the course we will talk about . . .

- Planning
- Uncertainty

Where are we?

The first two blocks of the course dealt with ...

- Basic notions of agency
- Intelligent problem-solving
- Heuristic search, constraints
- Logic & logical reasoning
- Reasoning about actions and time

In the remainder of the course we will talk about ...

- Planning
- Uncertainty

What is planning?

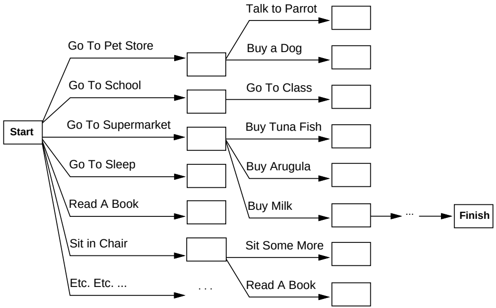
- **Planning** is the task of coming up with a sequence of actions that will achieve a goal
- We are only considering **classical planning** in which environments are
  - fully observable (accessible),
  - deterministic,
  - finite,
  - static (up to agents' actions),
  - discrete (in actions, states, objects and events).
- (Lifting some of these assumptions will be the subject of the "uncertainty" part of the course)

Why planning?

- So far we have dealt with two types of agents:
    1. Search-based problem-solving agents
    2. Logical planning agents
- Do these techniques work for solving planning problems?

Why planning?

- Consider a search-based problem-solving agent in a robot shopping world
- Task: Go to the supermarket and get milk, bananas and a cordless drill
- What would a search-based agent do?



Problems with search

- No goal-directedness.
- No problem decomposition into sub-goals that build on each other
  - May undo past achievements
  - May go to the store 3 times!
- Simple goal test doesn't allow for the identification of milestones
- How do we find a good heuristic function?
  How do we model the way humans perceive complex goals and the quality of a plan?

How about logic & deductive inference?

- Generally a good idea, allows for “opening up” representations of states, actions, goals and plans
- If $Goal = Have(Bananas) \wedge Have(Milk)$ this allows achievement of sub-goals (if independent)
- Current state can be described by properties in a compact way (e.g. $Have(Drill)$ stands for hundreds of states)
- Allows for compact description of actions, for example

$$Object(x) \Rightarrow Can(a, Grab(x))$$

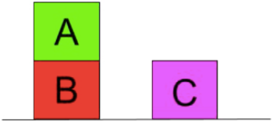
- Allows for representing a plan hierarchically, e.g. $GoTo(Supermarket) = Leave(House) \wedge ReachLocationOf(Supermarket) \wedge Enter(Supermarket)$ then decompose further into sub-plans

How about logic & deductive inference?

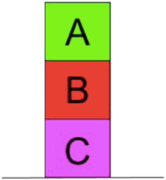
Problems:

1. In its general form either awkward (propositional logic) or tractability problems (first-order logic)
2. If p is a sequence that achieves the goal, then so is $[a, a^{-1}|p]$!
3. (Logically independent) subgoals may need to be undone to achieve other goals.

Goal: $on(A, B) \wedge on(B, C)$



Initial State



Goal

What next?

Solutions: We need

1. To reduce complexity to allow scaling up.
2. To allow reasoning to be guided by plan 'quality'/efficiency.

Do 1. next, and 2. after that.

Summary

- Planning is the task of identifying a sequence of actions to achieve your goal.
- Both search and logic are important tools, but need adapting to tackle:
  - efficient search and inference
  - discriminating the variable quality of valid plans
- Next time: Planning Domain Definition Language (PDDL) Formal representation of states, goals and plans.