COMP90054 – Week 2 tutorial

Last updated: 20 March 2023

Intro to AI and Planning

Rationally acting agents

- Knowledge & Percepts & Performance measure -> Action
- Maximise expected performance given its percept and knowledge
- **Problem -> Solver -> Solution**

Classical planning

Problem	->	Solver	->	Solution
Model	->	Planner	->	Action sequence

Planning is a **model-based approach** to autonomous behaviour:

- We use **model** to specify a **problem** P
- The system can be in one of many **states**
- Task of planning: Find **action sequence** to drive **initial state** into **goal state**
- Submit model to **planner** so it can come up with a **plan** (action sequence)

State-space model

$$\mathcal{S}(P) = \langle S, s_0, S_G, A, f, c \rangle$$

S	State space S, finite and discrete
s_0	A **known** Initial state s_0 (c.f. belief of initial state)
S_G	A set of goal states $S_G \subseteq S$
A	A set of actions, with $A(s) \subseteq A$ for each $s \in S$
f	A **deterministic** transition function $f: (a, s) \rightarrow s'$ for $a \in A(s)$
c	Positive **action cost** functions $c(a, s)$

Blind search algorithms

Why graph search?

- Model -> **Planner** -> Action sequence (Plan)
- We want to solve problem P by using **graph search algorithms** over the graph associate with $\mathcal{S}(P)$

Blind Search algorithms

- "Blind" means not requiring any input beyond problem P (c.f. Heuristics/informed search algorithms, coming up soon!)
- Algorithms
  - Breadth-First Search (BrFS)
  - Depth-First Search (DFS)
  - Iterative Deepening (ID)
- Overview

How to extract plans

By following/reversing steps of the search node expansion in search space

A search node σ contains

- $state(\sigma)$ Associated search state
- $parent(\sigma)$ Pointer to search node from which σ is reached $state(\sigma)$
- $action(\sigma)$ An action leading from $state\big(parent(\sigma)\big)$ to
- $g(\sigma)$ Cost of path from the root node to σ