

Searching: Intro

CPSC 322 – Search 1

Textbook §3.0 – 3.3

Lecture Overview

- 1 Agent Design
- 2 Example Problems
- 3 State Spaces

Agents and Representations

- Recall that an agent is something that **acts** in an environment
- The agent also receives **observations** about the environment
 - this could be observations from sensors such as cameras, laser rangefinders, etc.
 - can also include “observations” of the agent’s own past actions
- In a deterministic environment, the agent can perfectly predict the outcome of an action
 - doesn’t need sensors: just needs to remember its own past actions

The Table-Lookup Agent

- An agent can be thought of as a **mapping** from observations to the new action that the agent will take
- How should agents be constructed? One choice:
 - agent takes in the sequence of observations
 - agent looks up the correct action for this sequence of observations based on an internal representation (e.g., a table)
- Such an agent could indeed behave rationally. What's the problem?

The Table-Lookup Agent

- An agent can be thought of as a **mapping** from observations to the new action that the agent will take
- How should agents be constructed? One choice:
 - agent takes in the sequence of observations
 - agent looks up the correct action for this sequence of observations based on an internal representation (e.g., a table)
- Such an agent could indeed behave rationally. What's the problem?
 - too many sequences of observations are possible!
 - e.g., 10 possible observations, 10 timesteps $\rightarrow 10^{10}$ different entries in the table
 - compare this to e.g., the number of different move sequences that are possible in chess

Lecture Overview

- 1 Agent Design
- 2 Example Problems**
- 3 State Spaces

Example Problems

- To make things more concrete, let's think about some example problems:
 - solving a Sudoku
 - solving an 8-puzzle
 - the delivery robot planning the route it will take

What's an 8-Puzzle?

5	4	
6	1	8
7	3	2

Start State

1	2	3
8		4
7	6	5

Goal State

Example Problems

- To make things more concrete, let's think about some example problems:
 - solving a Sudoku
 - solving an 8-puzzle
 - the delivery robot planning the route it will take
- All of these problems are deterministic; thus, there's no need for any observations from sensors.
- Are these single or sequential decision problems?

Example Problems

- To make things more concrete, let's think about some example problems:
 - solving a Sudoku
 - solving an 8-puzzle
 - the delivery robot planning the route it will take
- All of these problems are deterministic; thus, there's no need for any observations from sensors.
- Are these single or sequential decision problems?
 - as discussed before, the distinction isn't really useful here; problems can be seen both ways
 - **CSPs**: settings where there's nothing meaningfully sequential about the decision
 - **Planning**: decisions are always sequential
 - **Now**: we're going to define the underlying tools that allow us to solve both

Lecture Overview

- 1 Agent Design
- 2 Example Problems
- 3 State Spaces**

State Spaces

- Idea: sometimes it doesn't matter what sequence of observations brought the world to a particular configuration; it just matters how the world is arranged now.
 - called the **Markov** assumption
- Represent the different configurations in which the world can be arranged as different **states**
 - which numbers are written in cells of the Sudoku and which are blank?
 - which numbers appear in which slots of the 8-puzzle?
 - where is the delivery robot?
- States are assignments of values to one or more **variables**
 - a single variable called "state"
 - x and y coordinates; etc...
- From each state, one or more **actions** may be available, which would move the world into a new state
 - write a new number in a blank cell of the Sudoku
 - slide a tile in the 8-puzzle
 - move the delivery robot to an adjacent location

Agent Design

- An agent can be thought of as a **mapping** from the given **state** to the new action that the agent will take
- However, there's a problem... often, we don't understand the domain well enough to build the mapping
 - we'd need to be able to tell the agent how it should behave in every state
 - that's why we want **intelligent** agents: they should decide how to act for themselves
 - in order for them to do so, we need to give them **goals**

State Spaces

- Represent the different configurations in which the world can be arranged as different **states**
 - which numbers are written in cells of the Sudoku and which are blank?
 - which numbers appear in which slots of the 8-puzzle?
 - where is the delivery robot?
- States are assignments of values to one or more **variables**
- From each state, one or more **actions** may be available, which would move the world into a new state
 - write a new number in a blank cell of the Sudoku
 - slide a tile in the 8-puzzle
 - move the delivery robot to an adjacent location
- Some states are **goal states**
 - A Sudoku state in which all numbers are different in each box, row and column
 - The single 8-puzzle state pictured earlier
 - The state in which the delivery robot is located in room 123