COMP219: Artificial Intelligence

Lecture 11: Search in Complex Environments

Overview

- Last time
  - Uniform cost search
  - Heuristics and heuristic search (greedy, A*)

- Today:
  - Search with partial observations
  - Belief states

- Learning outcome covered today:

  Identify, contrast and apply to simple examples the major search techniques that have been developed for problem-solving in AI;

Problem Types

- Deterministic, fully observable single-state problem
  - Agent knows exactly which state it will be in; solution is a sequence
- Non-observable sensorless problem (conformant problem)
  - Agent may have no idea where it is; solution is a sequence
- Non-deterministic and/or partially observable contingency problem
  - predicting the evolution of the environment becomes impossible
  - percepts provide new information about current state

    "For this reason, many people keep their eyes open while walking around or driving" – Russel&Norvig

  - often interleave search, execution
- Unknown state space exploration problem

Search Without Full Observation

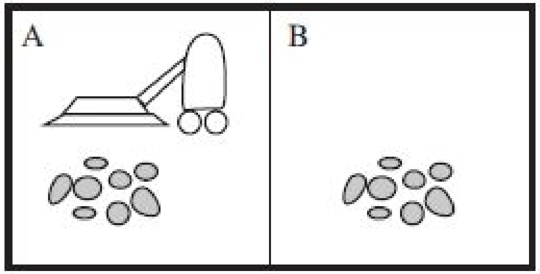
- Agent's percepts do not determine the exact state
- If agent in one of several possible states then an action may lead to one of several possible outcomes
- Belief state: agent's *current belief* about the possible *physical states* it might be in (given history – sequence of actions + percepts)

Search with No Observations

- Sensorless problem
- No need to rely on sensors working properly, e.g. manufacturing
- Less costly
- Search in belief states not physical states

Example:
Robot Vacuum Cleaner World

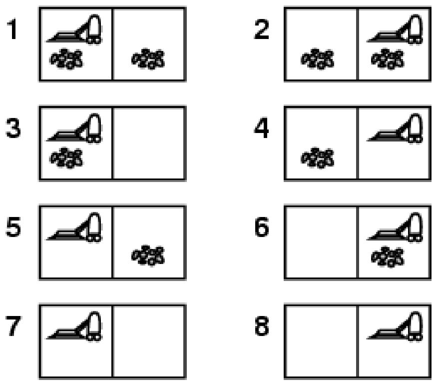
- Actions:
  – Right
  – Left
  – Suck



Example: vacuum world – 1

Single-state:
- Start in #5

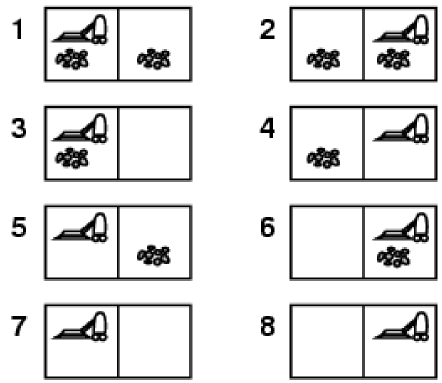
Solution?



Answer: vacuum world – 1

Single-state:
- Start in #5

Solution?
- *[Right, Suck]*

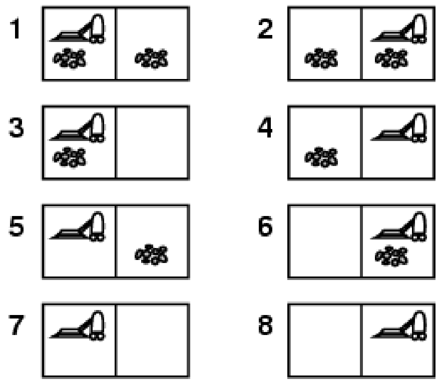


Example: vacuum world – 2

Sensorless:

- Start in {*1,2,3,4,5,6,7,8*}
- e.g. *Right* goes to {*2,4,6,8*}

Solution?



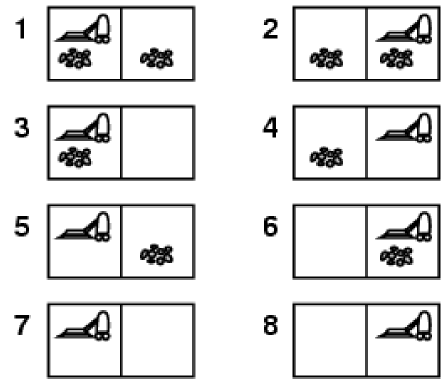
Answer: vacuum world – 2

Sensorless:

- Start in {*1,2,3,4,5,6,7,8*}
- e.g. *Right* goes to {*2,4,6,8*}

Solution?

- *[Right,Suck,Left,Suck]*

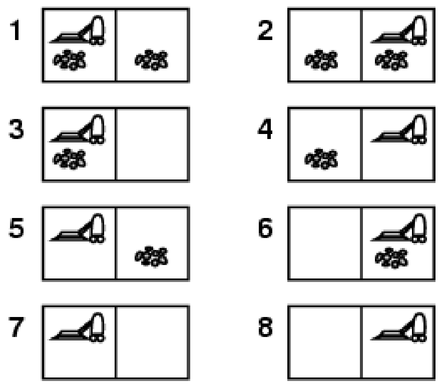


Example: vacuum world – 3

Contingency:

- *Non-deterministic*: suck may dirty a clean carpet
- *Partially observable*: location, dirt at current location
- Percept: [A, Clean] i.e. start in #5 or #7

Solution?



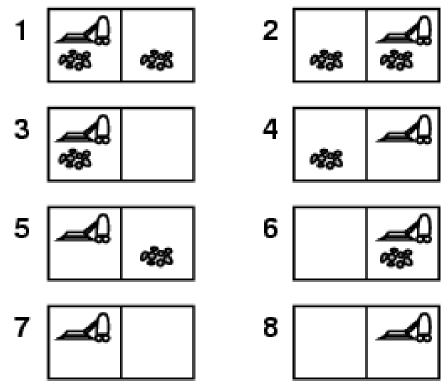
Answer: vacuum world – 3

Contingency:

- *Non-deterministic*: suck may dirty a clean carpet
- *Partially observable*: location, dirt at current location
- Percept: [A, Clean] i.e. start in #5 or #7

Solution?

- *[Right, **if** dirt **then** Suck]*



No Observations: Constructing a Belief State Search Problem

- Physical problem P is defined by $Actions_p$, $Result_p$, $Goal\text{-}Test_p$, $Step\text{-}Cost_p$
- Sensorless problem is defined by:
  - Belief States – every possible set (B) of physical states
  - Initial State – set of all states in P
  - Actions
  - Transition model
  - Goal test – all states in belief state must satisfy goal
  - Path cost

Determining Possible Actions

- Depends on whether it's safe to execute actions in states where they are not applicable
  - have an effect on the environment?
- If safe, union of all actions:

$$Actions(b) = \bigcup_{s \in b} Actions_P(s)$$

- Otherwise, only allow the intersection, i.e. actions which are legal in ***all*** states

Transition Model

- For deterministic actions, the set of states that might be reached is:

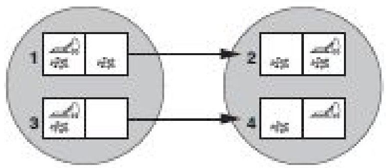
$$b' = Result(b, a) = \{s' : s' = Result_P(s, a) \text{ and } s \in b\}$$

  - b' is never larger than b
- For non-deterministic actions: b' may be larger
- Prediction step is process of generating the new belief state after the action

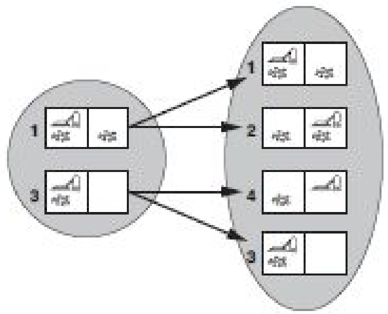
$$b' = Predict_P(b, a)$$

Prediction Step: Vacuum World

- Action: [Right]

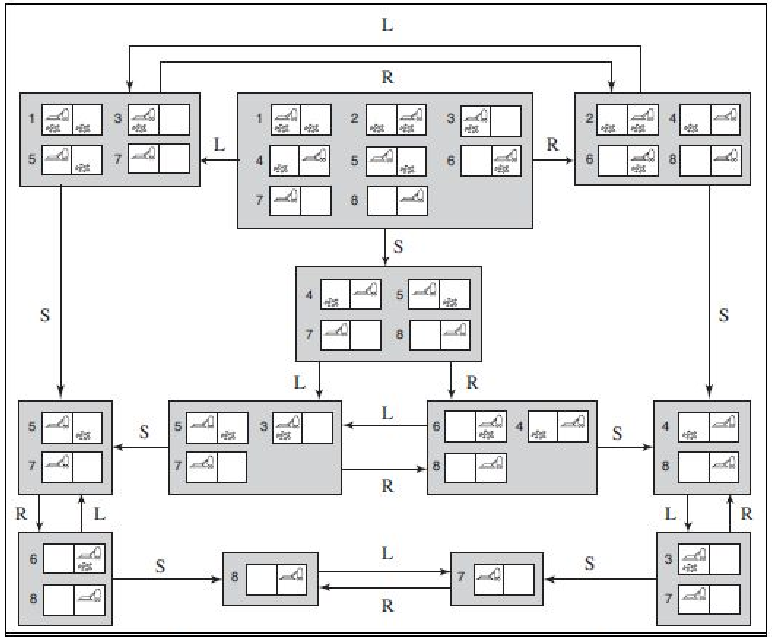


Deterministic



Non-deterministic 'slippery'

Belief State Problem: Vacuum World



Pruning the Belief State Graph

- Repeated belief states: don't add to agenda
  - E.g., *[Suck,Left,Suck]* reaches the same *b* as *[Right,Left,Suck] = {5,7}*
- Moreover, if an action sequence is a solution for a belief state *b*, it is also a solution for any subset of *b*
  - Suppose *{5,7}* has already been generated
  - Consider *[Left]*: *b={1,3,5,7}*
  - Can discard that path as its subset *{5,7}* will lead to solution whenever *{1,3,5,7}* does

Sensorless Problem Solving

- Problem: Size of each belief state
- e.g.
  10 x 10 vacuum world
  100 x 2^{100} = 10^{32} physical states

Incremental Belief State Search

- Unlike standard search algorithms, look inside *b*
- Build up solution one physical state at a time
- One solution for all states
- Find action sequence that works for all states:
  - Find solution for state 1
  - Check if it works for state 2, then state 3, etc.
  - If not, find different solution for state 1, etc.
- Advantage: detects failure quickly
  - If *b* unsolvable, usually the small subset of first few states examined is also unsolvable

Searching with Partial Observations

- Specify how the environment generates percepts
  - e.g. vacuum agent has position and local dirt sensor but no dirt sensor for other squares
  - Percept(state 1) = [A,Dirty]
- Usually several states could have produced the same percept
  - e.g. [A,Dirty] = {1, 3}

Searching with Partial Observations – Transitions

- 3 stages:
  - Prediction stage – same as sensorless problems

$$\hat{b} = \text{Predict}(b, a)$$

  - Observation prediction stage determines set of percepts o that could be observed in predicted belief state

$$\text{Possible - Percepts}(\hat{b}) = \{o : o = \text{Percept}(s) \text{ and } s \in \hat{b}\}$$

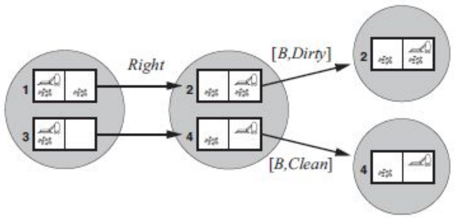
  - Update stage – for each possible percept, determine which belief states could result from the percept

$$b_o = \text{Update}(\hat{b}, o) = \{s : o = \text{Percept}(s) \text{ and } s \in \hat{b}\}$$

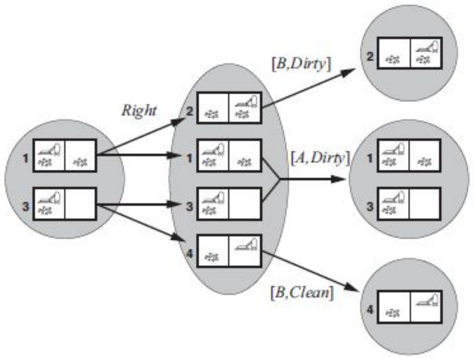
- Percepts reduce uncertainty

Transitions: Vacuum World

Deterministic



Non-deterministic 'slippery'

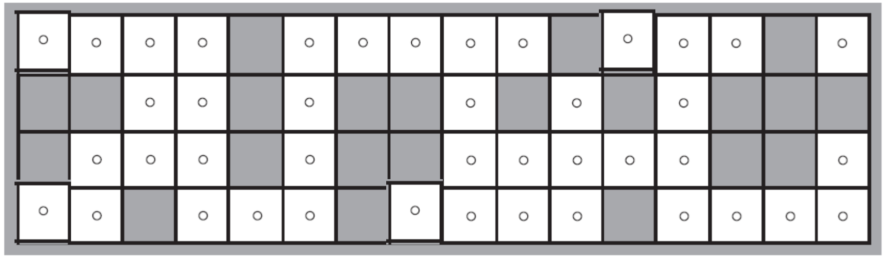


Exercise

- A robot has the task of localisation, i.e. working out where it is given a map of the world and a sequence of percepts and actions
- 4 sonar sensors tell the robot whether there is an obstacle in each compass direction (NESW)
- MOVE action moves randomly to an adjacent square

Exercise: Localisation

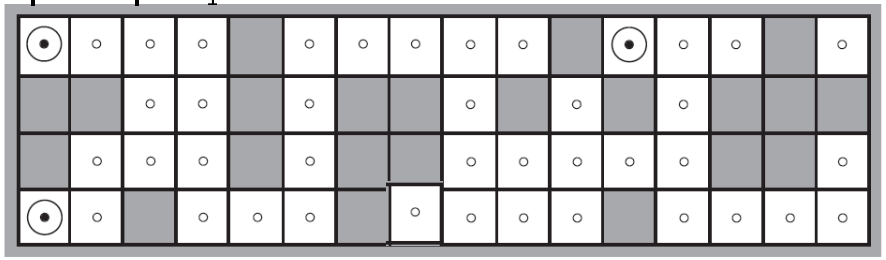
- Where is the robot?



- After one percept P_1=NSW?

Exercise: Localisation

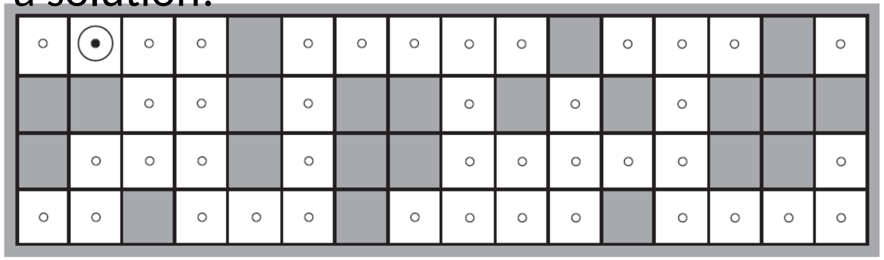
- Possible locations of the robot after one percept P_1=NSW



- After MOVE and next percept P_2=NS?

Exercise: Localisation

- After MOVE and next percept P_2=NS we have a solution!



Summary

- Search with no observation
  - Belief states
  - Pruning the belief state graph
  - Incremental belief state search
- Search with partial observation
  - Transitions: prediction; observation prediction; update

- Next time
  - Applying search to games