CS440/ECE 448, Lecture 19: Search in Partially Observable Environments

Slides by Mark Hasegawa-Johnson, 2/2022

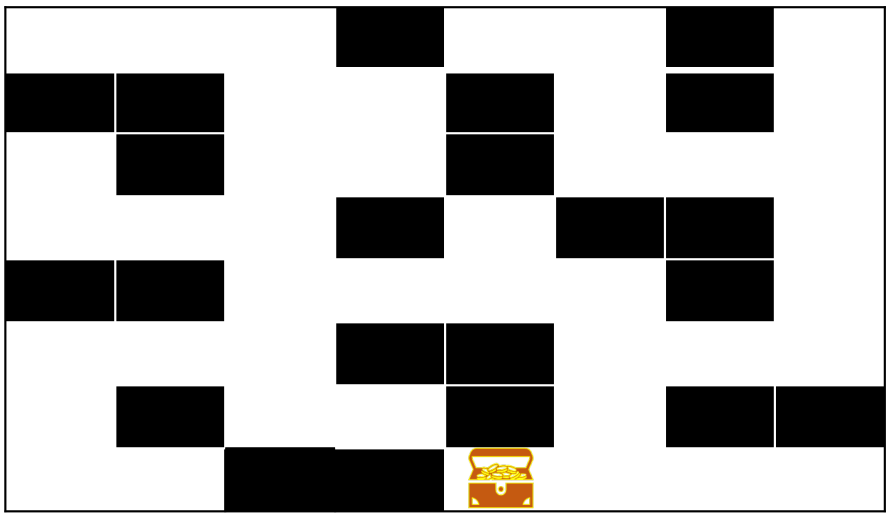
CC-BY 4.0: You are free to: copy and redistribute the material in any medium or format, remix, transform, and build upon the material for any purpose, even commercially, if you give appropriate credit.

Theodore Rombouts, The Card Players.
Public domain image, Residenzgalerie Salzburg,
https://commons.wikimedia.org/wiki/File:Theodoor_Rombouts_-_Joueurs_de_cartes.jpg

Content

- Unobservable environments: belief states
- Partially observable environments: predict and update
- Stochastic partially observable environments

Unobservable environments

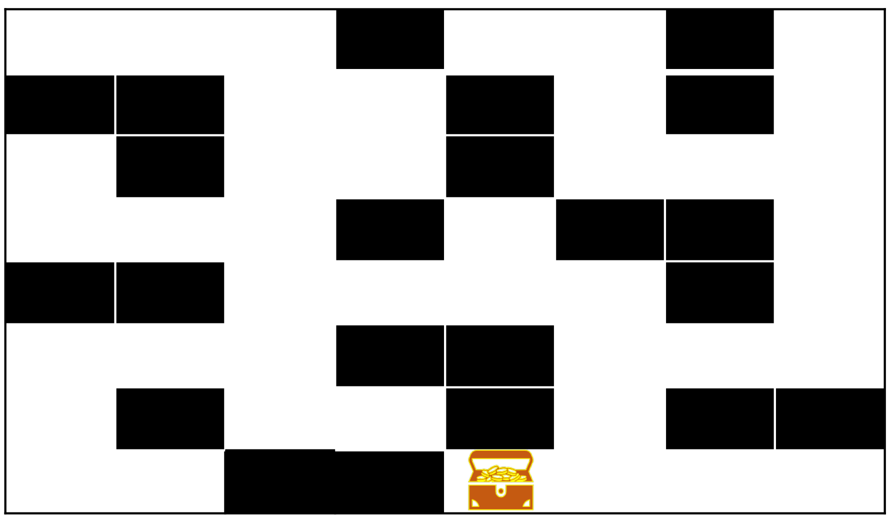


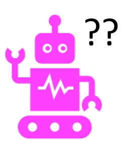
Consider an environment that’s relatively simple:

- Deterministic
- Discrete
- Known
- Single-agent
- Static
- Sequential

As we’ve learned, it should be possible to solve this problem using A* search, right?

Unobservable environments



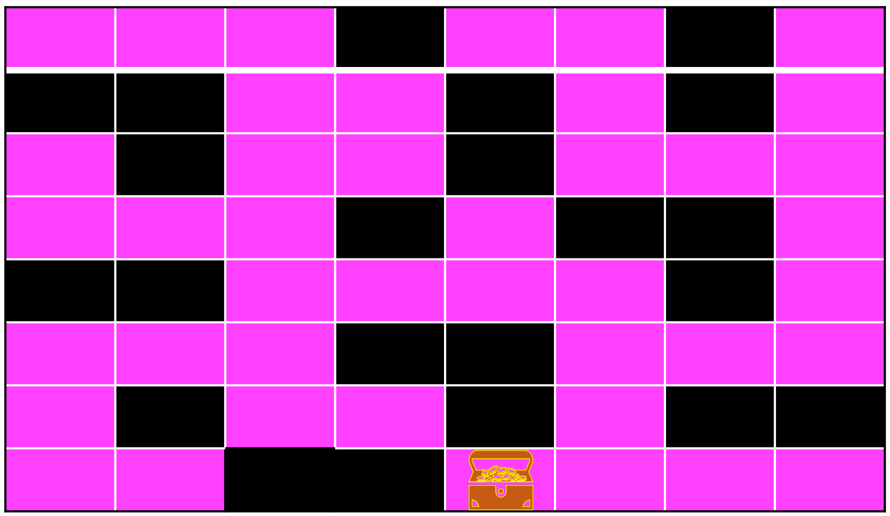


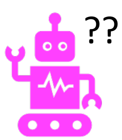
But suppose this environment is unobservable.

We know the layout of the map (it’s a known environment), but we don’t know which square we’re starting from. Once we move, we can’t tell whether we moved successfully, or ran into a wall.

Isn’t that a hopeless problem?

Unobservable environments

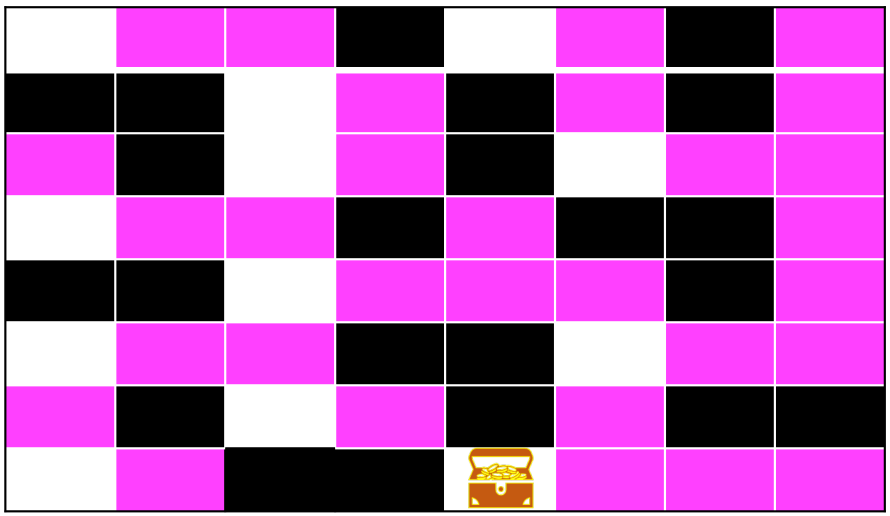




Let's use pink to color in all the squares where we might be.

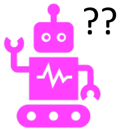
At the beginning of the search, we might be in any square, so all squares are pink.

Unobservable environments

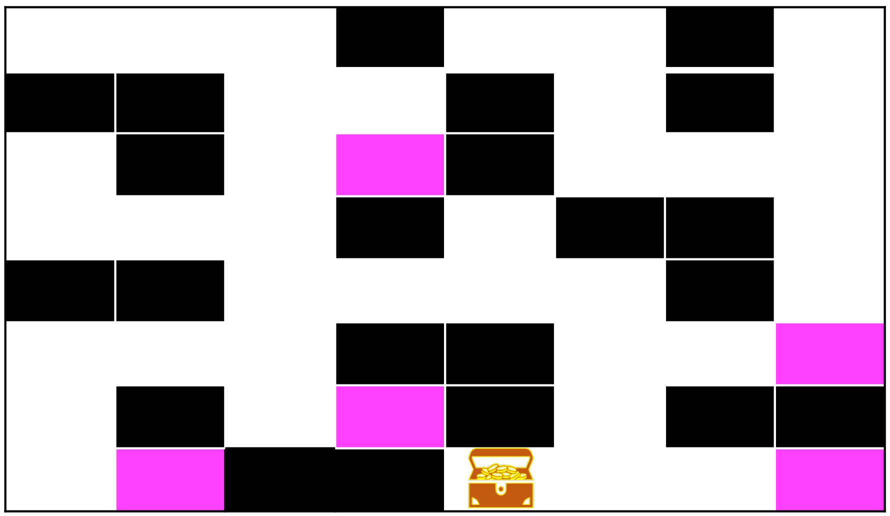


Now take 1 step to the right.

If the environment is deterministic, then we can now guarantee that we are in one of the pink squares shown here. We know we're not in any of the white squares.

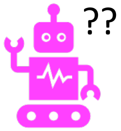


Unobservable environments



Take 4 steps right, then 4 steps down, then 4 steps right, then 4 steps down, then 2 steps right.

Now we know that we are in one of these five squares.

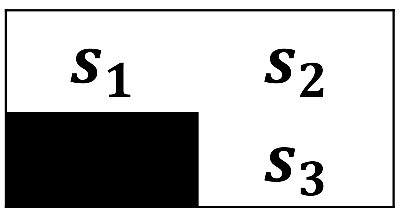


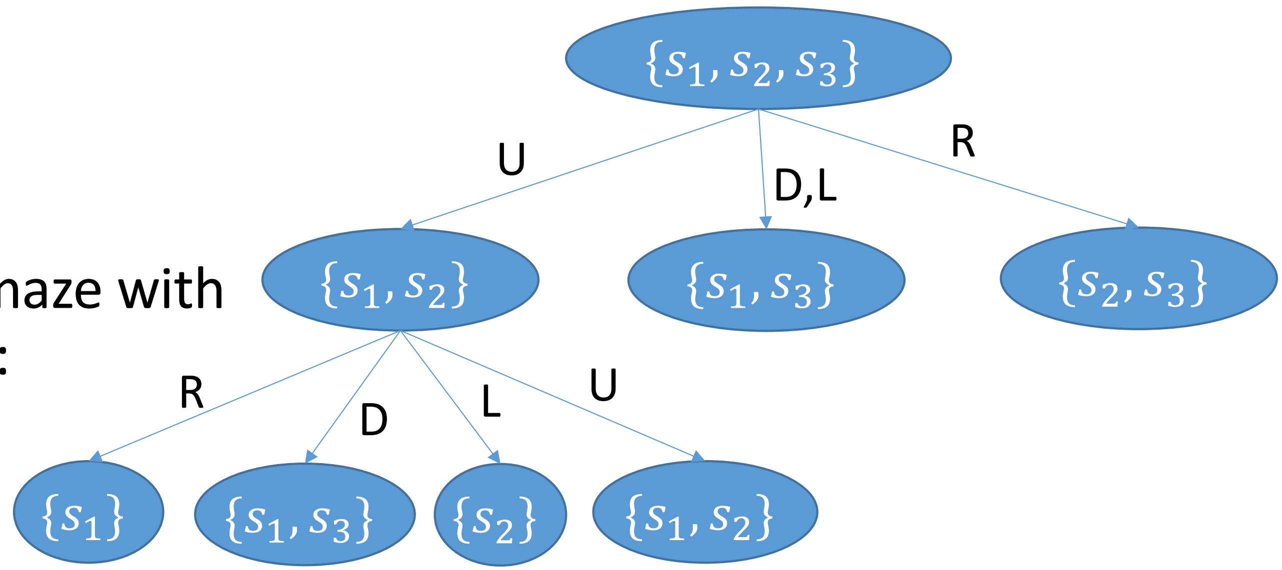
Belief state

- A **belief state** is a set of physical states (by "physical state," we mean a state of the environment).
  - In a maze, a physical state might be $s_1 = (x_1, y_1)$, the agent position.
  - The belief state is a set of physical states: $b = \{s_1, s_2, s_3, \dots\}$
- If the environment is unobservable, then we can't perform search using physical states.
- Instead, we create a search tree using belief states.

Belief states

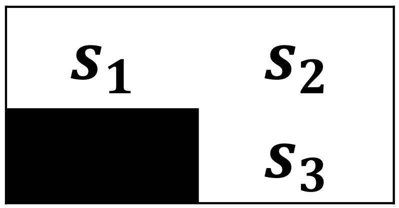
Suppose that we have a maze with only three physical states:





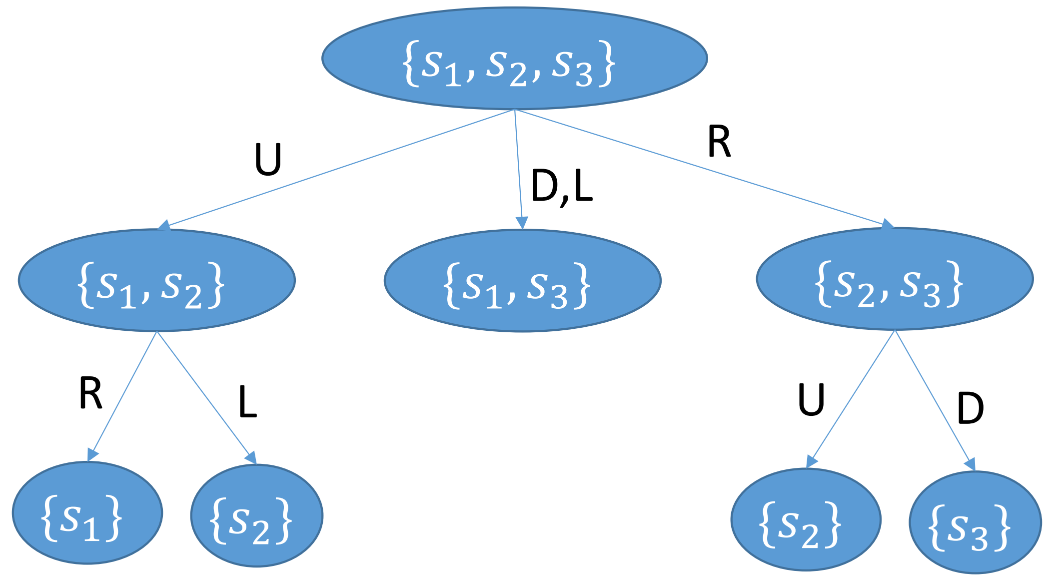
- At the beginning of search, the belief state is $b = \{s_1, s_2, s_3\}$.
- After one step to the right, the belief state is $b = \{s_2, s_3\}$

Belief states



A full breadth-first search on this maze can reach any desired physical state in just 2 steps, even if we have no idea where we started from.

(Shown here: the tree without any repeated states).



Computational Complexity

Remember that the time-complexity of BFS for an observable state space is $O\{\min(b^d, N)\}$, where b=branching factor, d=length of best path, N=# distinct states (exhaustive search). But…

- If N is the # distinct physical states, then 2^N is the number of distinct belief states. Exhaustive search is much more exhausting!!!
- The shortest path is also longer: call it d'.

Total:

$$O\{\min(b^{d\prime}, 2^N)\}$$

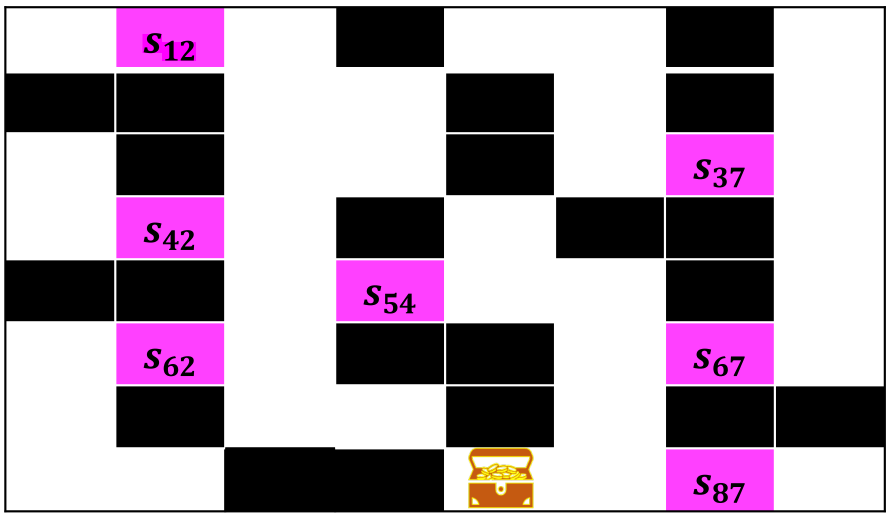
Content

- Unobservable environments: belief states
- Partially observable environments: predict and update
- Stochastic partially observable environments

Partially observable environments

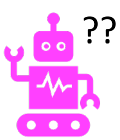
- Observable environment: the agent knows its state.
- Unobservable environment: the agent doesn’t know its state.
- Partially observable environment: the agent can observe something, but not everything.

Partially observable environment

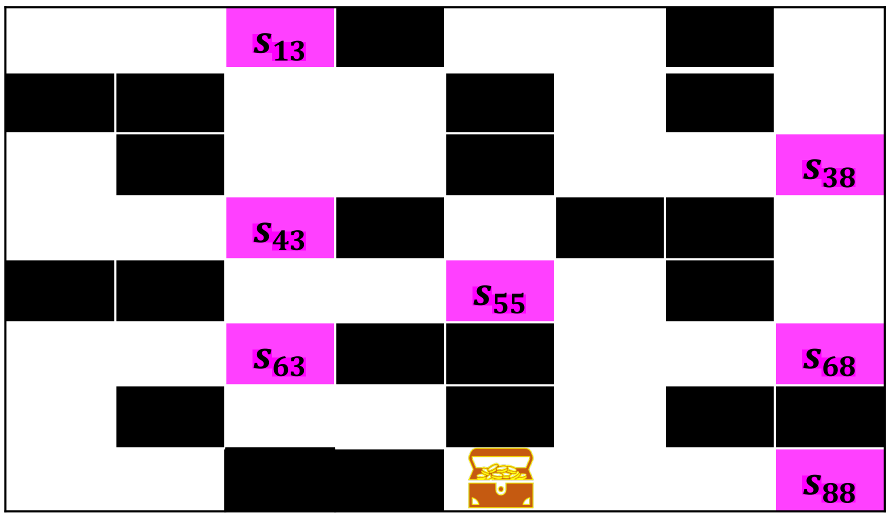


Suppose we begin by sensing the bit vector $\vec{o} = [0,1,0,1]^T$.

Then we know that our initial position must be one of these squares.

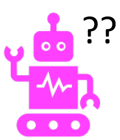


Partially observable environment

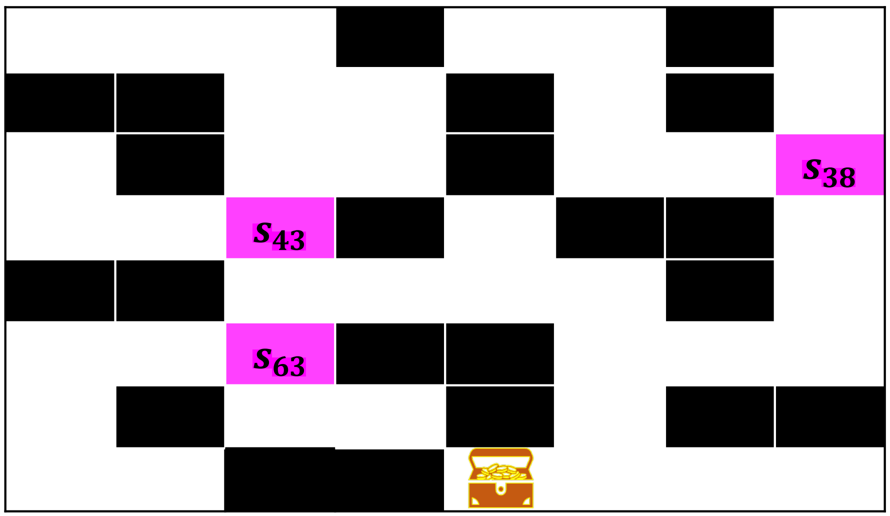


Move one step to the right.

Now we know that our position must be one of these squares.

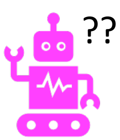


Partially observable environment



Now suppose we sense again, and measure the bit vector $\vec{o} = [1,0,0,0]^T$.

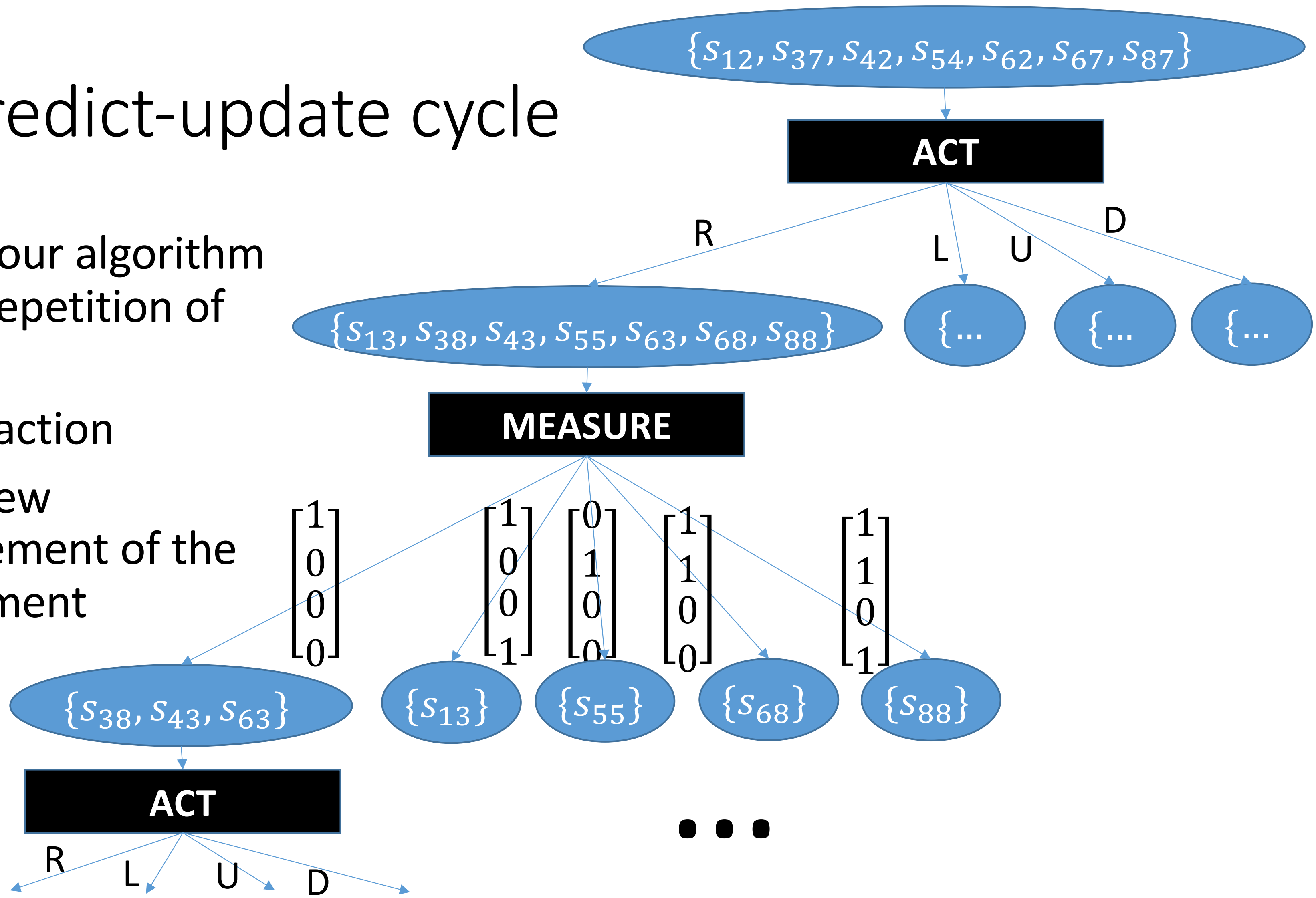
Then we know that our new position must be one of these squares.



The predict-update cycle

Notice that our algorithm is now the repetition of two steps:

1. Take an action
2. Take a new measurement of the environment



The predict-update cycle

Given an initial belief state, e.g., $b = \{s_{12}, s_{37}, s_{42}, s_{54}, s_{62}, s_{67}, s_{87}\}$, we repeat the following two steps:

1. ACT (action a), and then **predict** what will be the result of our action. For example, if our action is $a = R$, then our predicted outcome is $b = \{s_{13}, s_{38}, s_{43}, s_{55}, s_{63}, s_{68}, s_{88}\}$
2. MEASURE the environment again (observation $\vec{o}$), and then **update** our belief state (by deleting any physical states that are incompatible with the new measurement). For example, if our measurement is $\vec{o} = [1,0,0,0]^T$, then our new belief state is $b = \{s_{38}, s_{43}, s_{63}\}$

The predict-update cycle

Given an initial belief state, e.g., $b = \{s_{12}, s_{37}, s_{42}, s_{54}, s_{62}, s_{67}, s_{87}\}$, we repeat the following two steps:

1. ACT (action a), and then **predict** what will be the result of our action.

$$b \leftarrow \mathrm{PREDICT}(b, a)$$

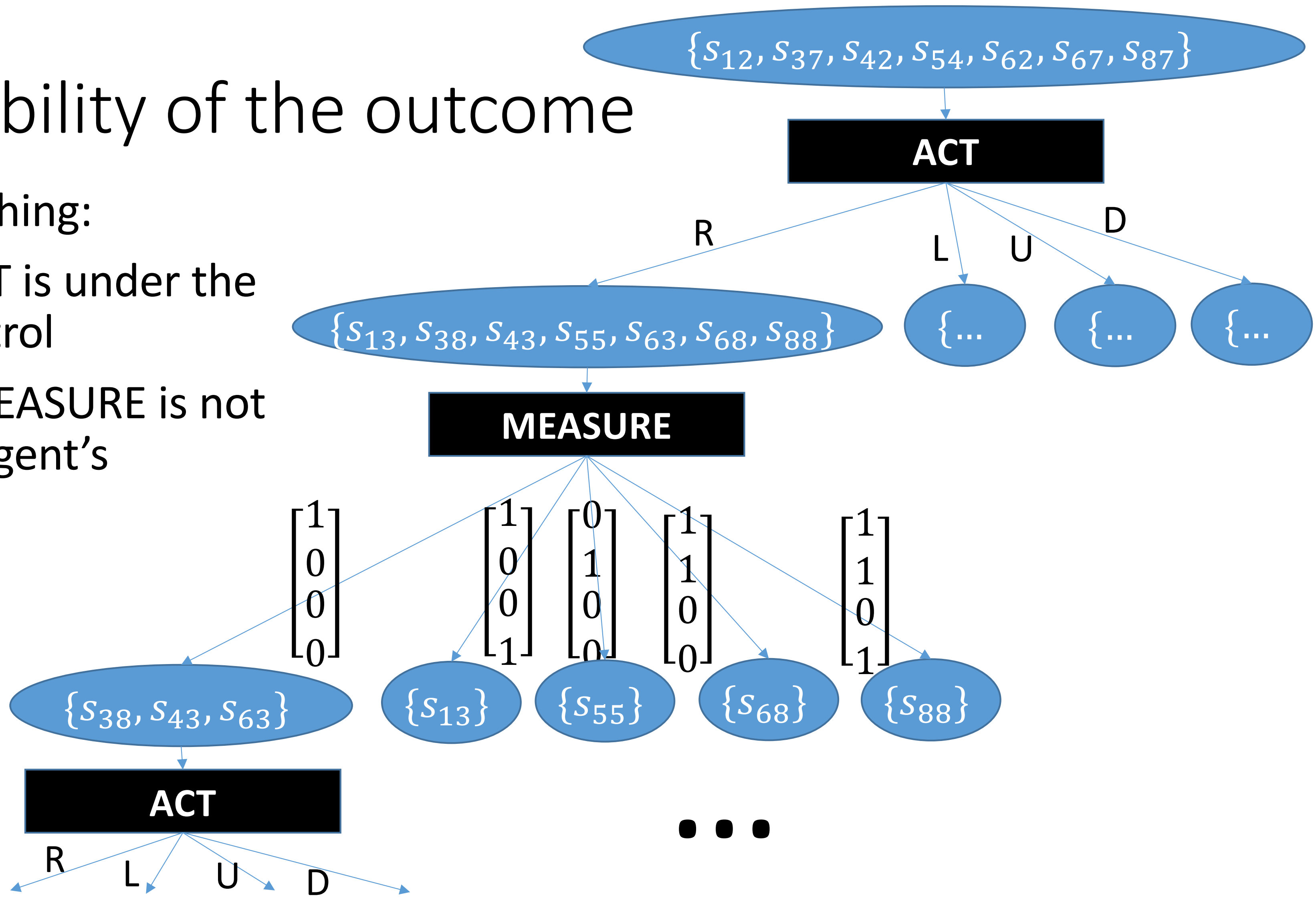
2. MEASURE the environment again (observation $\vec{o}$), and then **update** our belief state (by deleting any physical states that are incompatible with the new measurement).

$$b \leftarrow \mathrm{UPDATE}(b, \vec{o})$$

Controllability of the outcome

Notice something:

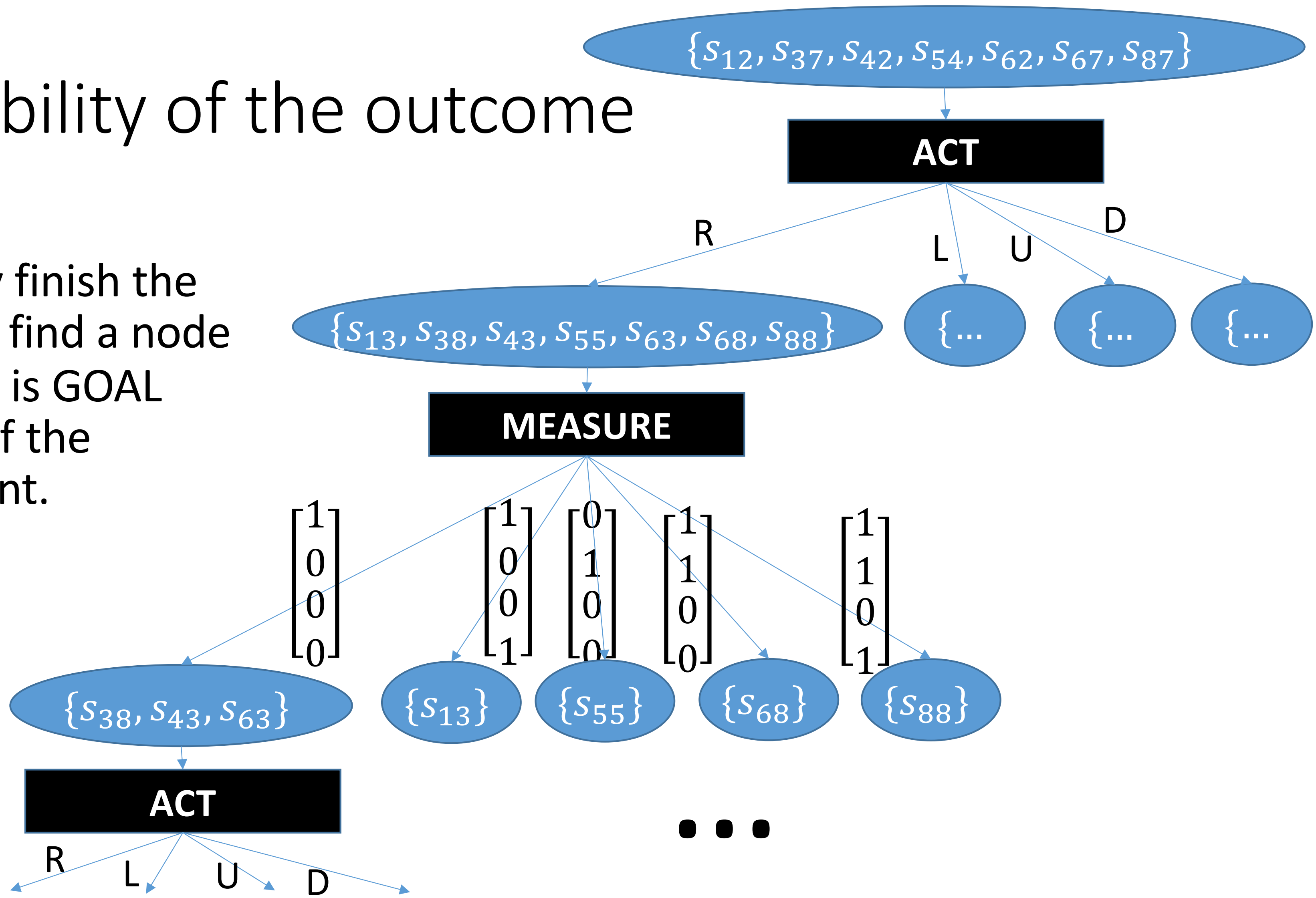
- How we ACT is under the agent's control
- What we MEASURE is not under the agent's control!



Controllability of the outcome

For BFS:

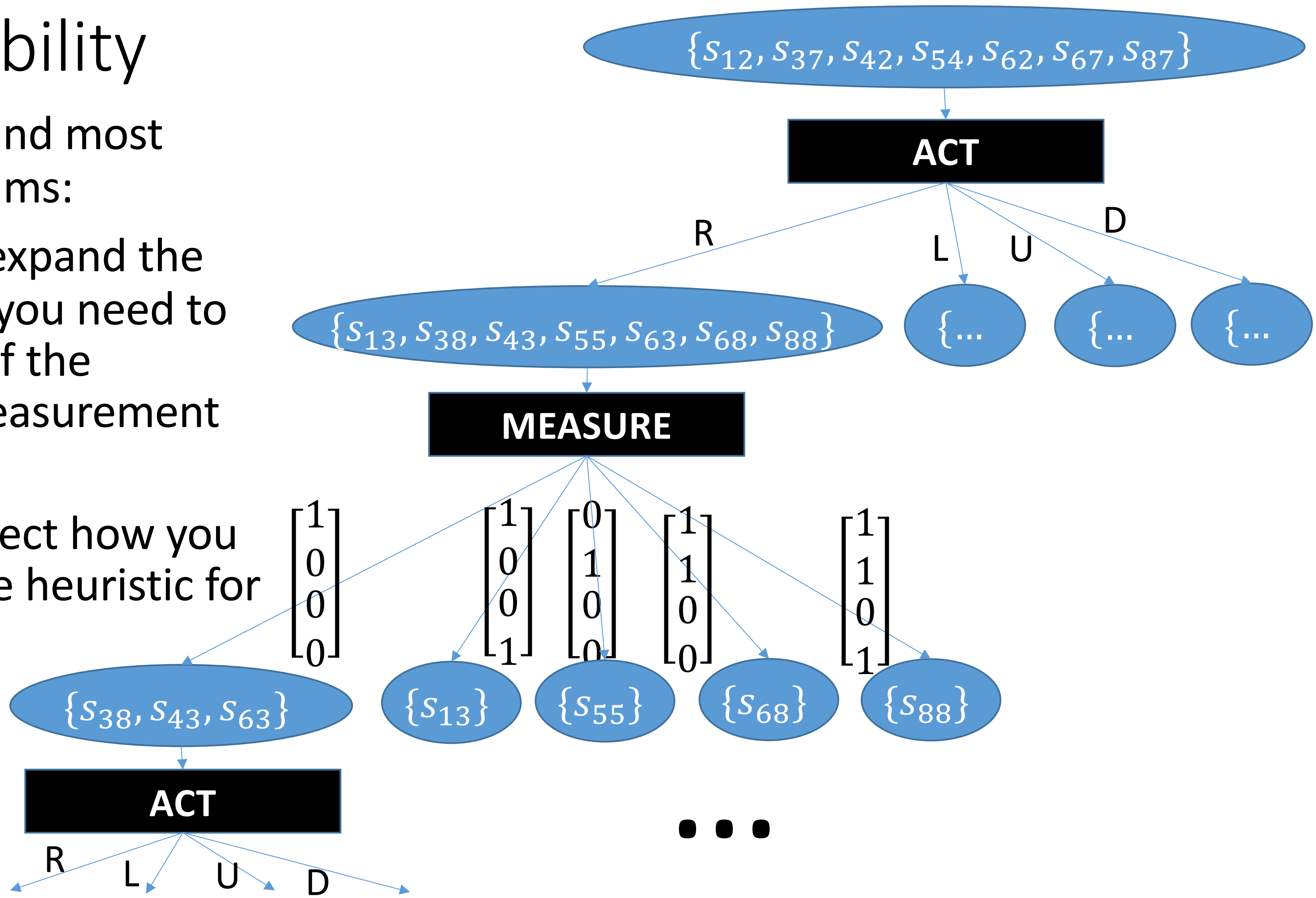
- We can only finish the search if we find a node whose state is GOAL regardless of the measurement.



Controllability

For DFS, A*, and most other algorithms:

- In order to expand the action a=R, you need to expand all of the resulting measurement states.
- This may affect how you calculate the heuristic for A*.



Content

- Unobservable environments: belief states
- Partially observable environments: predict and update
- Stochastic partially observable environments

Stochastic partially observable environments

- Observable environment: the agent knows its state.
- Unobservable environment: the agent doesn't know its state.
- Partially observable environment: the agent can observe something, but not everything.

- Deterministic environment: action a, in state s, always leads to the same successor state: $s' = \mathrm{TRANSITION}(s, a)$
- Stochastic environment: action a, in state s, leads to a probability distribution over possible successor states

$$P(s'|s, a) = \mathrm{TRANSITION}(s, a)$$

Stochastic partially observable environments

1. **Predict:** given the current belief state $b = \{s_1, s_2, s_3, \ldots\}$, and the action a, **expand** the belief state to $b' = \{s_1{}', s_2{}', s_3{}', \ldots\}$ that have ***nonzero probability*** $P(s_j{}'|s_i, a)$ for any $s_i \in b$.

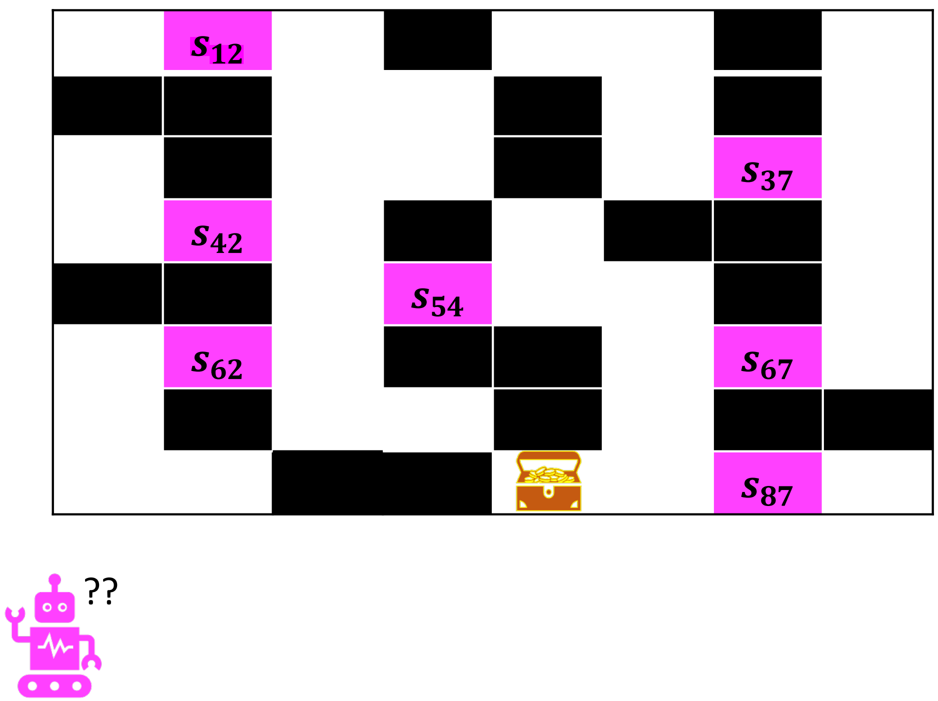
$$b' \leftarrow \mathrm{PREDICT}(b, a)$$

2. **Update:** **contract** the belief state by deleting any physical states that are incompatible with the new measurement.

$$b \leftarrow \mathrm{UPDATE}(b, \vec{o})$$

The randomness is dealt with here.

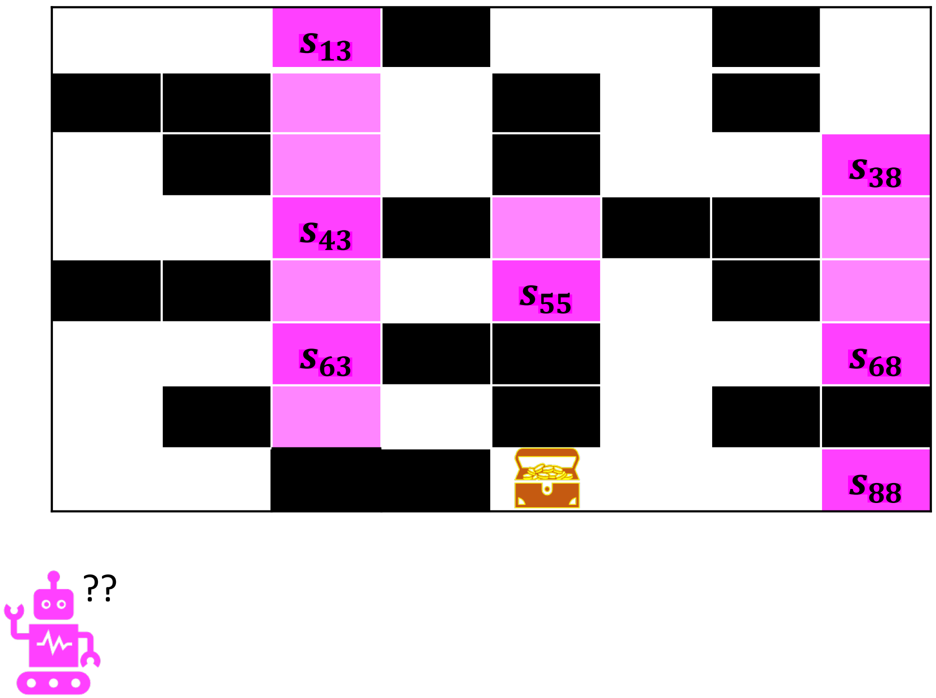
Stochastic partially observable environment



Suppose we begin by sensing the bit vector $\vec{o} = [0,1,0,1]^T$, so we know that the initial state is one of these squares.

However, this is a stochastic environment. If we try to move right, we might accidentally move diagonally (right+up or right+down).

Stochastic partially observable environment



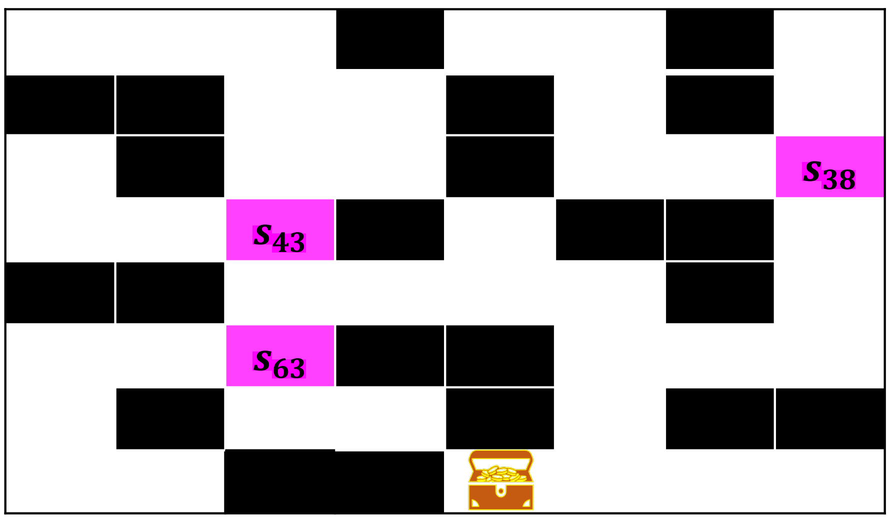
… so when we try to move one step to the right, our new belief state includes not only the original belief state (darker) but also the physical states that result from a random mis-step (lighter).

$$b' \leftarrow \text{PREDICT}(b, a)$$

is the list of all physical states shown here.

The belief state has **<u>expanded</u>**, from 7 to 14 physical states.

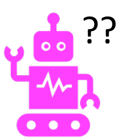
Stochastic partially observable environment



Now suppose we sense again, and measure the bit vector $\vec{o} = [1,0,0,0]^T$.

$$b'' \leftarrow \mathrm{UPDATE}(b', \vec{o})$$

includes only the three states shown here. All others have been eliminated by the measurement.



The belief state has **contracted**, from 14 to 3 physical states.

Content

- Unobservable environments
  - A belief state is a set of physical states: all the physical states that are possible
  - If there are N physical states, there are 2^N belief states
- Partially observable environments
  - $b' \leftarrow \mathrm{PREDICT}(b, a)$: usually b’ and b have the same size
  - $b'' \leftarrow \mathrm{UPDATE}(b', \vec{o})$: usually b’’ is smaller than b’
- Stochastic partially observable environments
  - $b' \leftarrow \mathrm{PREDICT}(b, a)$: usually b’ is larger than b
  - $b'' \leftarrow \mathrm{UPDATE}(b', \vec{o})$: usually b’’ is smaller than b’