Graph Search

CPSC 322 – Search 2

Textbook §3.4

Lecture Overview

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

Lecture Overview

Search

- What we want to be able to do:
  - find a solution when we are not given an algorithm to solve a problem, but only a specification of what a solution looks like
  - idea: search for a solution

Definition (search problem)

A search problem is defined by

- A set of states
- A start state
- A goal state or goal test
  - a boolean function which tells us whether a given state is a goal state
- A successor function
  - a mapping from a state to a set of new states

Abstract Definition

How to search

- Start at the start state
- Consider the different states that could be encountered by moving from a state that has been previously expanded
- Stop when a goal state is encountered

To make this more formal, we'll need to talk about graphs...

Search Graphs

Definition (graph)

A graph consists of

- a set N of nodes;
- a set A of ordered pairs of nodes, called arcs or edges.

- Node n_2 is a neighbor of n_1 if there is an arc from n_1 to n_2.
    - i.e., if $\langle n_1, n_2 \rangle \in A$

Definition (path)

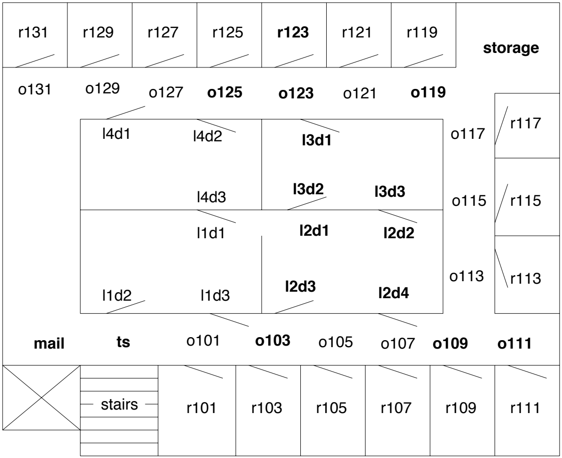
A path is a sequence of nodes $\langle n_0, n_1, \ldots, n_k \rangle$ such that $\langle n_{i-1}, n_i \rangle \in A$.

Definition (solution)

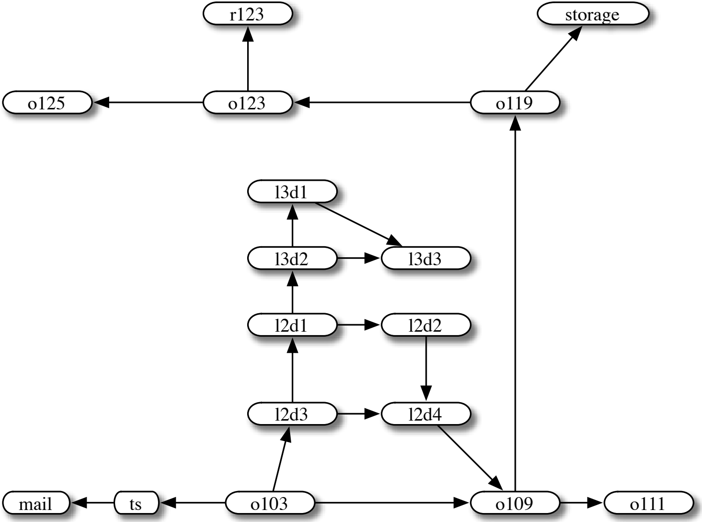
Given a start node and a set of goal nodes, a solution is a path from the start node to a goal node.

Example Domain for the Delivery Robot

The agent starts outside room 103,
and wants to end up inside room 123.



Example Graph for the Delivery Robot



Example Sudoku Problem

3		5					6	
6					2		3	
2			5					
4	3		1			2	5	
	5	8			6		1	4
					8			1
	4		3					5
	6					9		8

Let's define this as a search problem. What are:

- the set of states?

Example Sudoku Problem

3		5					6	
6					2		3	
2			5					
4	3		1			2	5	
	5	8			6		1	4
					8			1
	4		3					5
	6					9		8

Let's define this as a search problem. What are:

- the set of states?
- the start state?

Example Sudoku Problem

3		5					6	
6					2		3	
2			5					
4	3		1			2	5	
	5	8			6		1	4
					8			1
	4		3					5
	6					9		8

Let's define this as a search problem. What are:

- the set of states?
- the start state?
- the goal state or goal test?

Example Sudoku Problem

3		5					6	
6					2		3	
2			5					
4	3		1			2	5	
	5	8			6		1	4
					8			1
	4		3					5
	6					9		8

Let's define this as a search problem. What are:

- the set of states?
- the start state?
- the goal state or goal test?
- the successor function?

Example Sudoku Problem

3		5					6	
6					2		3	
2			5					
4	3		1			2	5	
	5	8			6		1	4
					8			1
	4		3					5
	6					9		8

Let's define this as a search problem. What are:

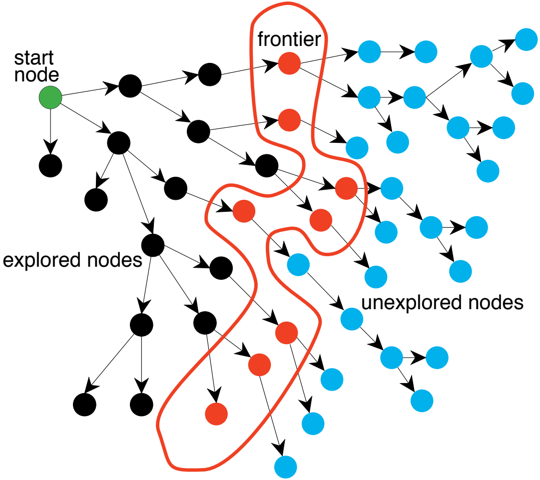
- the set of states?
- the start state?
- the goal state or goal test?
- the successor function?

Note: here only the goal matters, not the path to it.

Graph Searching

- Generic search algorithm: given a graph, start nodes, and goal nodes, incrementally explore paths from the start nodes.
- Maintain a frontier of paths from the start node that have been explored.
- As search proceeds, the frontier expands into the unexplored nodes until a goal node is encountered.

Problem Solving by Graph Searching



Graph Searching

- Generic search algorithm: given a graph, start nodes, and goal nodes, incrementally explore paths from the start nodes.
- Maintain a frontier of paths from the start node that have been explored.
- As search proceeds, the frontier expands into the unexplored nodes until a goal node is encountered.
- The way in which the frontier is expanded defines the search strategy.

Lecture Overview

Graph Search Algorithm

Input: a graph,
a set of start nodes,
Boolean procedure $goal(n)$ that tests if n is a goal node.
$frontier := \{\langle s \rangle : s$ is a start node$\}$;
while $frontier$ is not empty:
  select and **remove** path $\langle n_0, \ldots, n_k \rangle$ from $frontier$;
  if $goal(n_k)$
    return $\langle n_0, \ldots, n_k \rangle$;
  for every neighbor n of n_k
    add $\langle n_0, \ldots, n_k, n \rangle$ to $frontier$;
end while

- After the algorithm returns, it can be asked for more answers and the procedure continues.
- Which value is selected from the frontier defines the search strategy.
- The $neighbor$ relationship defines the graph.
- The $goal$ function defines what is a solution.

Branching Factor

Definition (forward branching factor)

The forward branching factor of a node is the number of arcs going out of that node.

- If the forward branching factor of every node is b and the graph is a tree, how many nodes are exactly n steps away from the start node?

Branching Factor

Definition (forward branching factor)

The forward branching factor of a node is the number of arcs going out of that node.

- If the forward branching factor of every node is b and the graph is a tree, how many nodes are exactly n steps away from the start node?
  - b^n nodes.
- We'll assume that all branching factors are finite.

Comparing Algorithms

Definition (complete)

A search algorithm is complete if, whenever at least one solution exists, the algorithm is guaranteed to find a solution within a finite amount of time

Definition (time complexity)

The time complexity of a search algorithm is an expression for the worst-case amount of time it will take to run, expressed in terms of the maximum path length m and the maximum branching factor b.

Definition (space complexity)

The space complexity of a search algorithm is an expression for the worst-case amount of memory that the algorithm will use, expressed in terms of m and b.