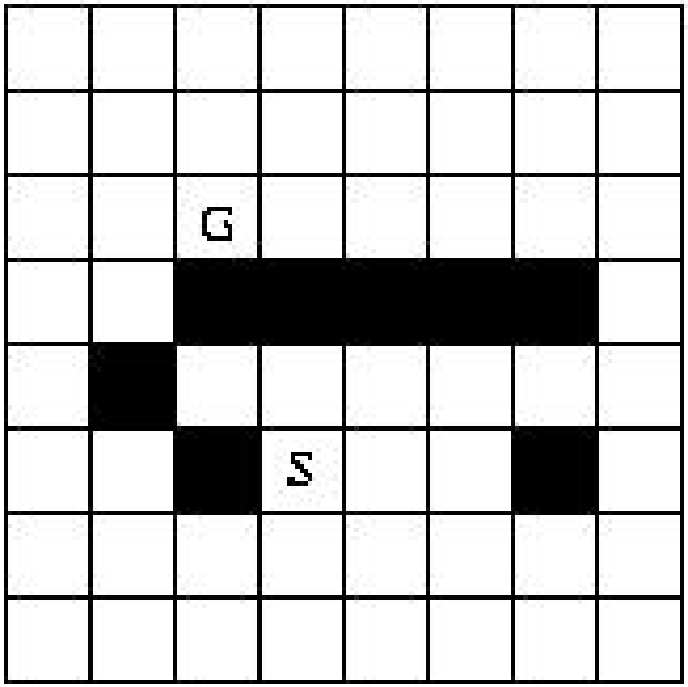
CSC 520 - Artificial Intelligence

Search - Introduction

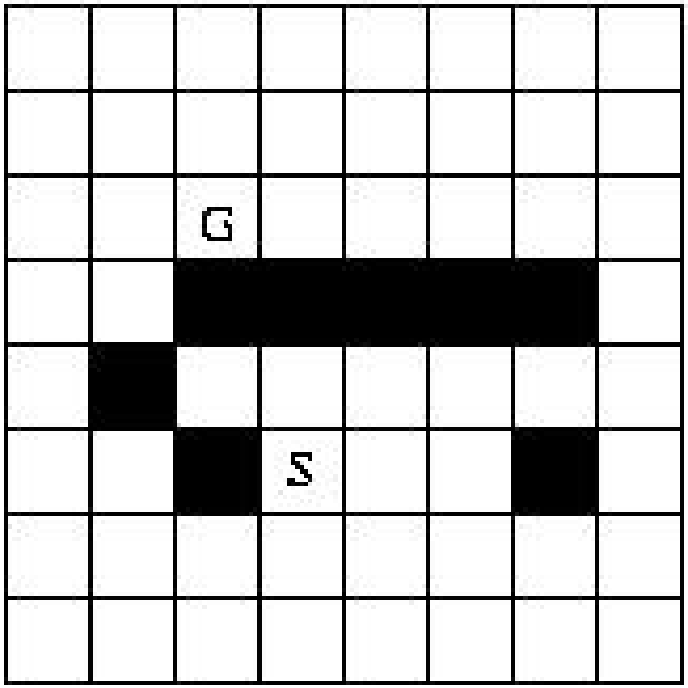
Srinath Ravindran

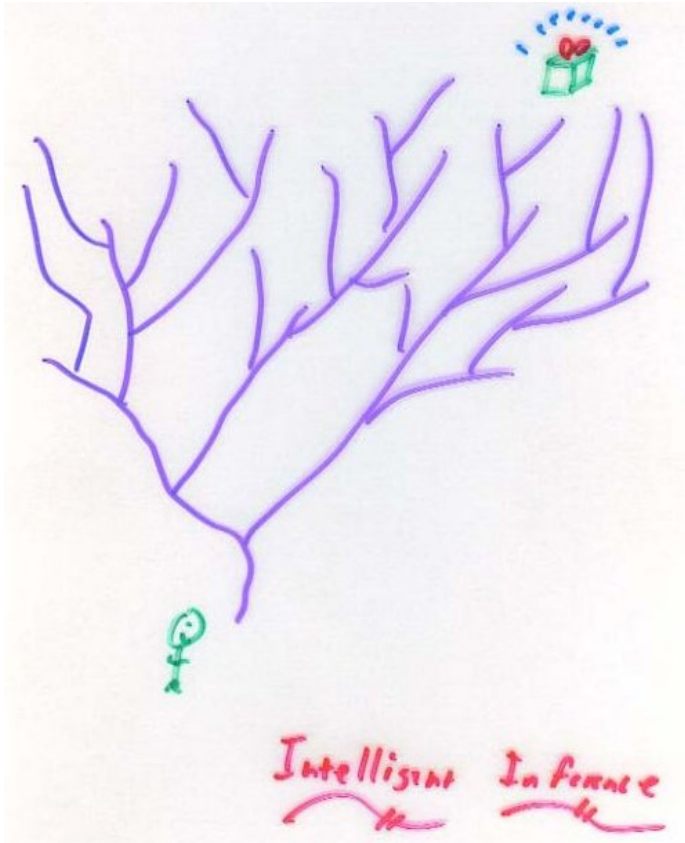
September 14, 2010

Search



Search





Search: Definition

What is search ?

- Defintion: The process by which we find a solution to a problem by trying a set of steps in "search" of a sequence that will solve the problem.
- In principle, only need to know the "rules of the game", *not* how to play well.

Why learn about search ?

- The agent in an environment searches for its goal.
- Depending on the goal and environment, search is the base to solving many complex problems
- Entire AI rests on search
  - Theorem Proving
  - Planning
  - Machine learning
  - Natural Language Processing

Search: The problem

- An agent attempts to find solutions to problems
- A solution is a sequence of actions that lead to desirable states in an environment
- To Do : Find the sequence of actions that lead to the solution, hence the solution itself.
- Search is the systematic enumeration of potential partial solutions to a problem so that they can be checked to see if they truly are solutions, or could lead to solutions.

How search works

- First, we represent the world as a collection of states that the environment+agent can be in, and to represent each action as a transition between a pair of states.
- This yields a graph with nodes representing states and edges representing actions.
- Goal: a state, or more realistically, a set of desirable states.
- The idea is to find a sequence of actions that will bring about a goal state starting from current state, the state which describes the current state of the environment+agent.
- Such a sequence corresponds to a path in the graph

State Space Search

- State Space
  - A set of all possible states in a problem solving process.

    The entire state space can be represented as a graph if it is small enough. Most realistic state spaces are too large for this.

    In each state, we maintain information relevant and required for further navigation.
  - A set of operators: steps in a problem solving process

Search Graph

- Only as much of the graph as necessary is ever explicitly represented. A good search algorithm should look at as small a subgraph of the state space as it can.
- The subgraph, which depends on the search algorithm used, is called the search graph.
- The state space depends on the problem and how states are represented; the search graph depends on the state space and the algorithm used to search the state space.

State Space - Problem Instance

- Problem Instance
  - State Space
  - Initial State
  - Set of Goal States
    - May be enumerated
    - may be described implicitly with a measurable property
  - Path Cost
    - Assumed to be Step-wise
- Example: Chess
  - State Space: All possible configurations of the board
  - Initial State: opening configuration
  - Goal States: Checkmate / Draw (?)
  - Path Cost: Uniform

State Space Search: Assumptions

- Several Assumptions used for these algorithms.
  - Static Environment
  - Fully Observable
  - Discrete
  - Deterministic
- Some searches relax these constraints.

State Space Search: Challenges

- Finding a sequence of operators (Job of search)
- Representing states & operators (AI scientist's job)
- Dealing with combinatorial explosion
  - For practical interesting problems, numbers of possibilities is incredibly large.

Goal: Develop intelligence that can restrict search to promising regions of search spaces!

Types of Searches

- Major types of search
  - Forward vs Backward
  - Stochastic vs Deterministic
  - Informed vs Uninformed
  - Feasible vs Optimal
- "Best" depends on context.

Types of Search: Forward vs Backward

- Forward Search: Start at the initial state, search for goal. Useful if there's little branching
- Backward Search: Start at goal, search backwards to the beginning
  - Have to reverse operators
  - Useful when there are few paths to a goal.
- Bidirectional Search: Start at both, grow trees until they intersect.
  - Compromise between two methods.
  - Sometimes requires heuristics to guarantee they'll meet.

Types of Search: Stochastic vs Deterministic

- Deterministic Search: The search space is explored in a systematic, pre-determined order.
  - Some searches can guarantee an answer.
  - Can be slow to find the answer.
- Stochastic Search: Randomly choose which direction to search in.
  - Randomness sometimes affects correctness
  - Quicker than deterministic search.

Types of Search: Informed vs Uninformed

- Uninformed, or blind search, systematically explores the space. Systematic Search will eventually visit every state.
- Informed, or heuristic search, uses estimates of path cost to guide search.

Types of Search: Feasible vs Optimal

- Feasible Search: Just want to reach goal.
  - Path Cost can be uniform or ignored.
- Optimal Search: Minimize path cost
  - Optionally, maximize reward
- Most blind searches are feasible, informed searches are optimal

Example: Missionaries and Cannibals

- 3 missionaries and 3 cannibals seek to cross a river
- The only available boat holds 2 people
- If m's on a bank outnumbered by c's on a bank, c's will eat m's.
- Find simplest schedule of crossings that will permit "non-consumption".

Missionaries and Cannibals: State Space

- State Space: Set of states - position of missionaries and cannibals
- Question: What features do we represent?
  - # in boat
  - # on shore
  - boat position
  - direction of boat travel
  - color of boat
  - temperature
  - day of week
  - Russia's prime minister
  - manufacturer of cannibals' watch
- Solution Strongly affected by representation

Missionaries and Cannibals: Problem Definition

- States
  - # M's on left [L = left bank]
  - # C's on left [R = right bank]
  - boat position (L, R)
  - etc . . . (?)
- Operators
  - MM: 2 M's cross the river
  - MC: 1 M and 1 C cross the river
  - CC: 2 C's cross the river
  - M: 1 M crosses the river
  - C: 1 C crosses the river
- Initial State: (3,3,L)
- Goal State: (0,0,R)

Missionaries and Cannibals: Solution

State	Action
3,3,L	MC
2,2,R	M
3,2,L	CC
3,0,R	C
3,1,L	MM
1,1,R	MC
2,2,L	MM
0,2,R	C
0,3,L	CC
0,1,R	C
0,2,L	CC
0,0,R	GOAL

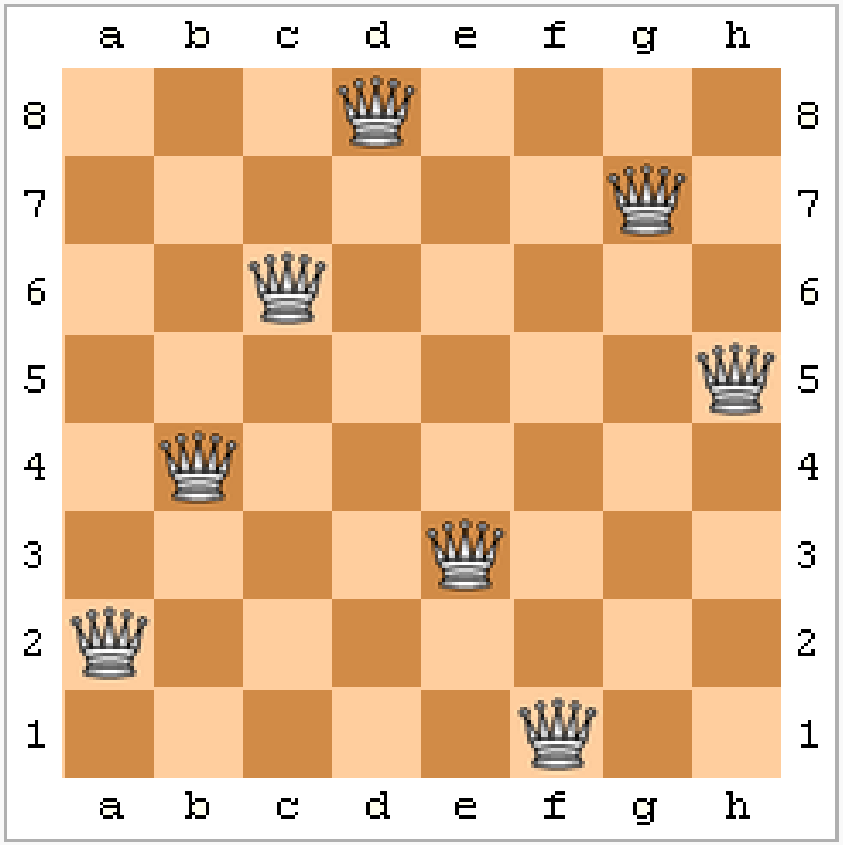
8 Queens Puzzle

- Consider a standard 8x8 chess board
- We wish to place 8 queens on the board
- Each queen should not threaten any other queen.

8 Queens: Problem Instance

- State Space:
  - States: Any possible configuration in which 8 queens are on the 8x8 board
  - Operators: Place a queen on a square
- Initial State: Blank board
- Goal State: All 8 queens placed, none threatening another.

8 Queens: Solution



One solution.