CS 440: Introduction to Artificial Intelligence

Lecture 9

Matthew Stone

September 30, 2015

Recap— Search

- Initial state
- Possible actions in each state
- Transition model:
  Takes state and action and gives new state
- Goal test
  Describes whether state is what you want
- Path cost
  Says how easy or hard action sequence is

Recap— Breadth-first search

Simple regime for exploring

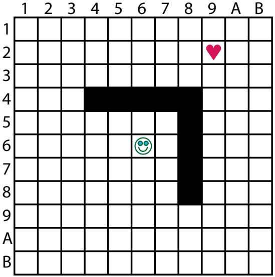
- Gradually "fan out" into the search space
- Explore level by level
- Consider all the nodes at level n first
- Then consider nodes at level $n + 1$ (and so on)

Recap—Depth-first search

- Implement frontier as a stack
- Small space requirements
- Efficient realization through function calls
- Not always shortest path first
- Related idea: “iterative deepening”

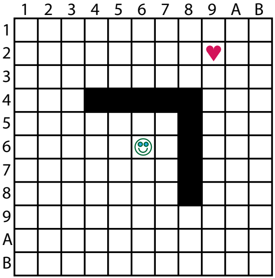
Example

- planning paths in a tiled world
- can move one square n, s, e, w
- cannot move through obstacles
- must stay on board



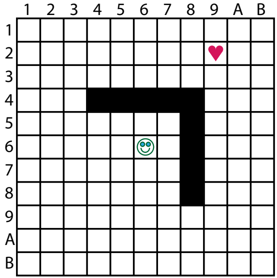
Mr Happy wants to find love

- initial state?
- goal test?
- actions in (6,6)? in (7,6)? in (7,5)? in (7,3)?



Mr Happy wants to find love

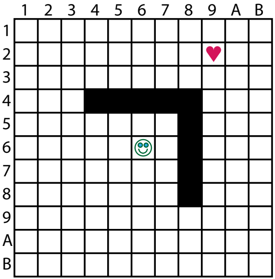
- initial state?
- goal test?
- actions in (6,6)? in (7,6)? in (7,5)? in (7,3)?



init:(6,6). goal(s):s=(9,2). (6,6):{n,s,e,w}. (7,6):{n,s,w}. (7,5):{s,w}. (7,3):{n,e,w}.

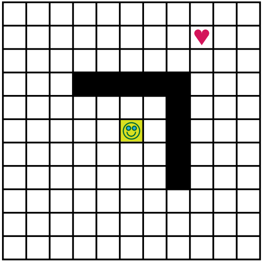
Questions

- BFS frontier after depth 0?
- After depth 1?
- After depth 2?
- After depth 3?



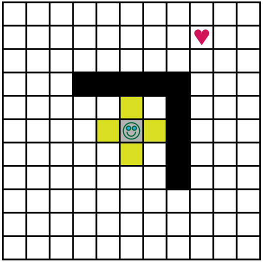
Questions

- BFS frontier after depth 0?



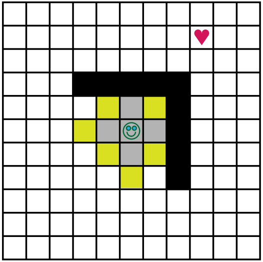
Questions

- BFS frontier after depth 1?



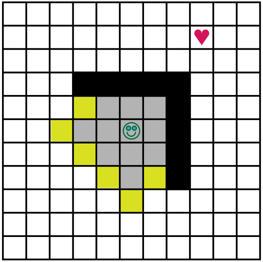
Questions

- BFS frontier after depth 2?



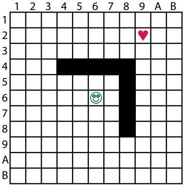
Questions

- BFS frontier after depth 3?



Questions about uninformed search

- Do you need frontier, explored list for DFS here?
- Why or why not?
- What does that say about DFS memory use?



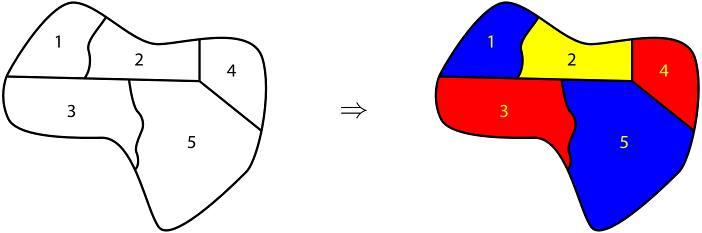
Questions about uninformed search

- Do you need frontier, explored list for DFS here?
    - Yes, otherwise DFS will search many redundant paths
- What does that say about DFS memory use?
    - No way to use tree model of DFS search with cheap memory

Map Coloring

- each “country” gets a color
- neighboring countries must get different colors
- use at most k colors
  interesting cases: k is 3 or 4

Example



Representation for DFS

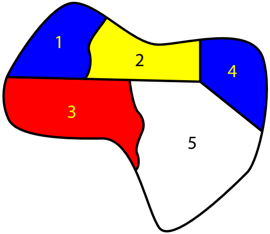
- Node is assignment of colors to first i countries
- Action is assign consistent color to country $i+1$
- Goal is all countries colored

Demo at `http://www.mathcove.net/petersen/lessons/get-lesson?les=14`

Search example

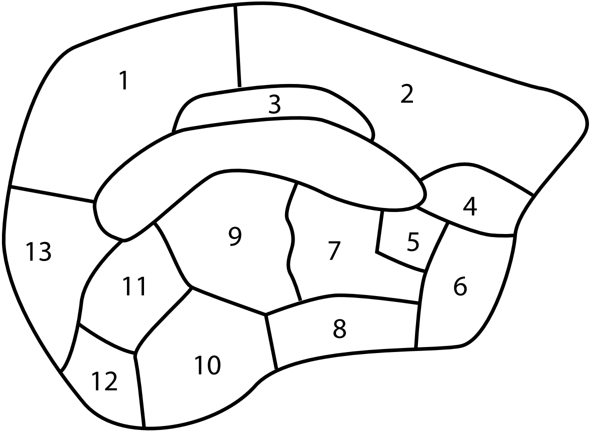
3 color our map

- Assign color 1 to country 1
- Assign color 2 to country 2
- Assign color 3 to country 3
- Assign color 1 to country 4
- Dead end: must backtrack



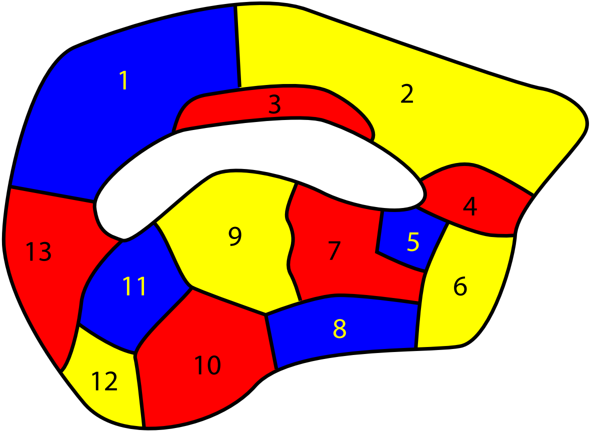
Aside: making search plausible

3 color "donutland"



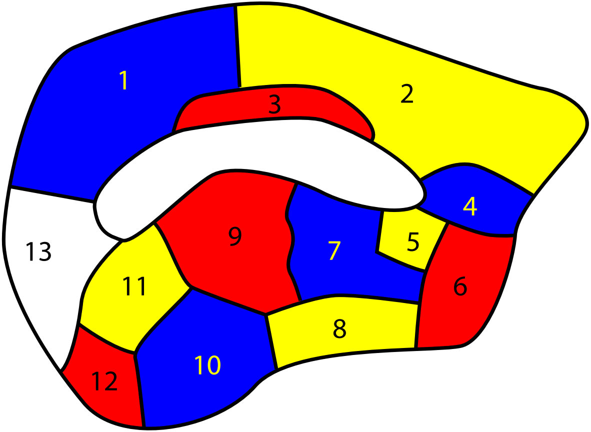
Aside: making search plausible

Answer



Aside: making search plausible

Serious dead end



Properties of representation

- Solution lies at depth n for n countries
- Search space is a tree

Alternative representation

- Node is assignment of colors to all *n* countries (not necessarily consistent)
- Action is change color of any country *i*
- Goal is consistent coloring

Properties of representation

- Solution may be very close to random initial point
- Search space is graph
- Many paths between any two states
- Suitable representation for local search (see next week)
  - hill climbing
  - simulated annealing
  - genetic search

Demo at `http://www.ff.iij4u.or.jp/~kanada/ccm/coloring/`