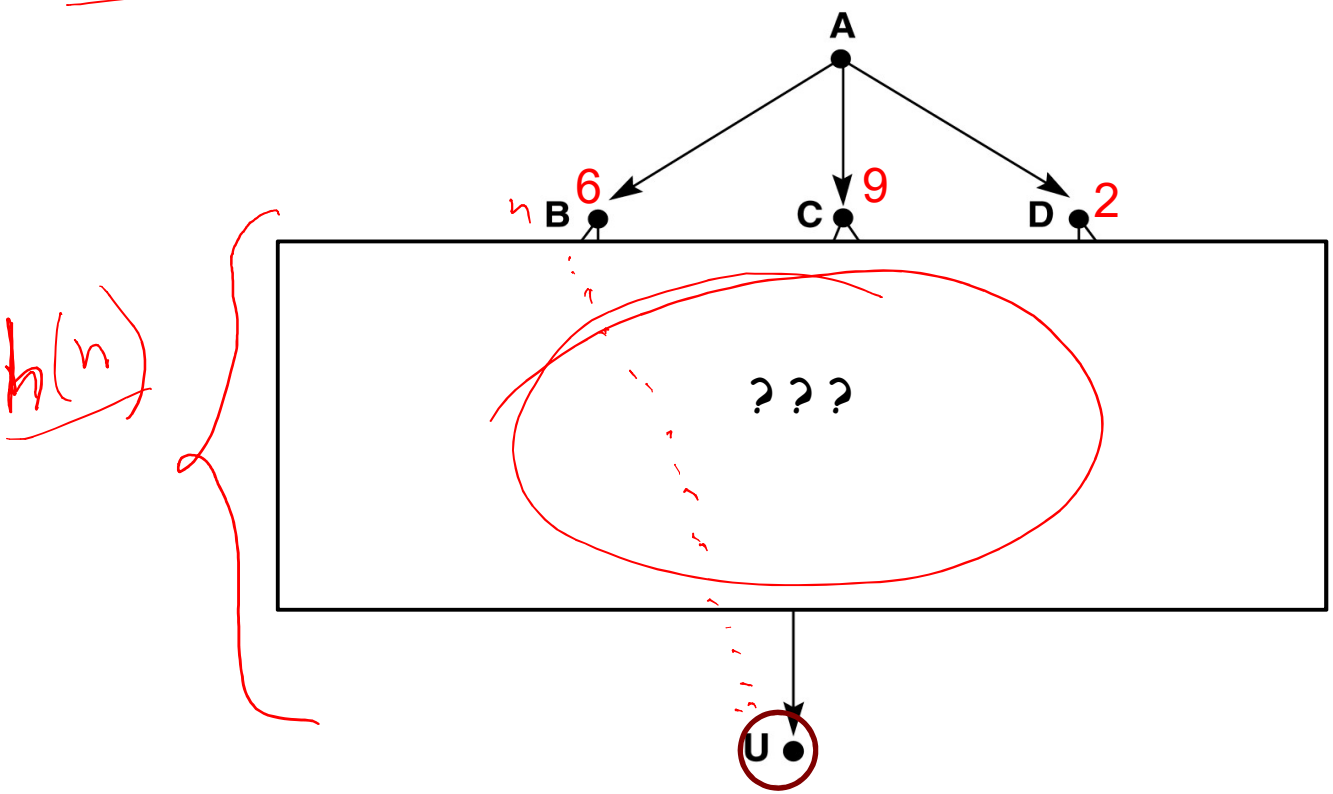
Artificial Intelligence: State Space Search part 3
Informed Search
Hill Climbing video #4

- Russell & Norvig – Sections 3.5.1, 3.5.2, 4.1.1

Today

1. State Space Representation
2. State Space Search
   a) Overview
   b) Uninformed search
      1. Breadth-first Search and Depth-first Search
      2. Depth-limited Search
      3. Iterative Deepening
      4. Uniform Cost
   c) Informed search
      1. Intro to Heuristics
      2. Hill Climbing
      3. Greedy Best-First Search
      4. Algorithms A & A*
      5. More on Heuristics
   d) Summary

h(n)



- **h(n)** = estimate of the lowest cost from *n* to *goal*

Hill Climbing

- General idea:
  - Similar to climbing a mountain in the fog with amnesia …

健忘症

  - in the fog
    - --> only 1-step view of what is to come, so

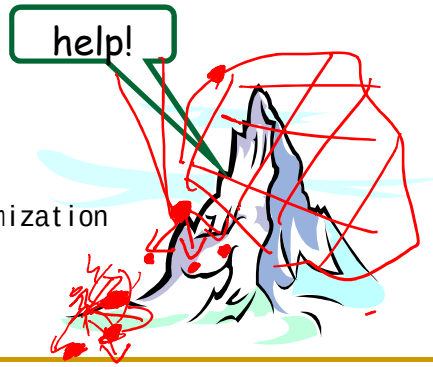
只能看见下一步，如果能爬的更高，go，不然到顶了stop

    - if next step seems higher than where you are now -> go
    - otherwise, you assume you are at the top of the mountain -> stop
  - with amnesia -->

没法backtrack,no open list

    - if you ever want to try other path, you can't because you did not keep track of where you came from

可能达到的是local optimization

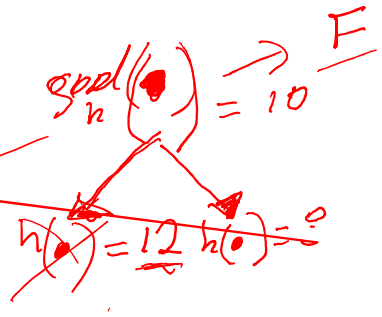


Vanilla HC vs Steepest Ascent HC

General Hill Climbing

共同点

- uses h(n)
- does not use an open list (amnesia)



1. Vanilla Hill Climbing
   - take 1st successor s with better h() than current state n
   - i.e. if lower h(n) is better, chose 1st s with h(s) < h(n) // deep diving

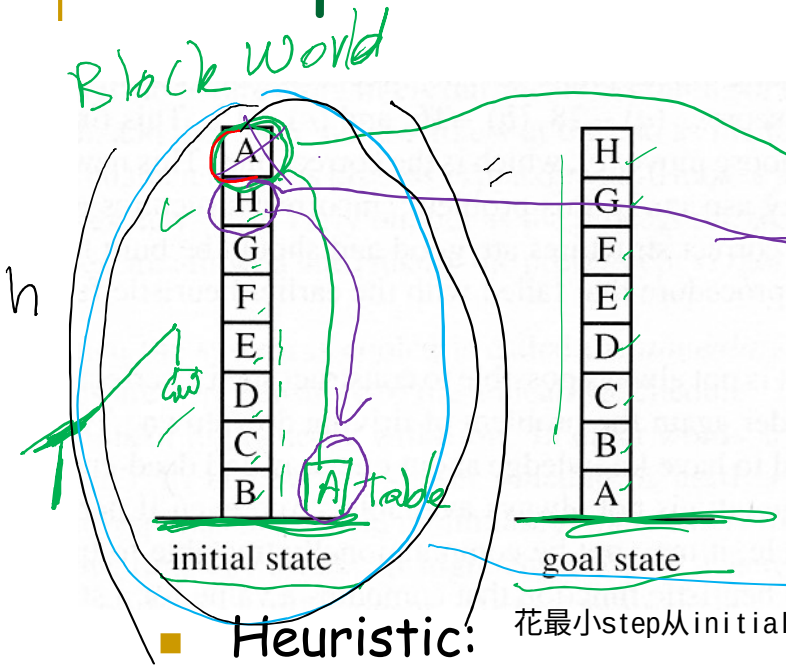
   第一个successor

   只要有比当前state好的successor，take it ，哪怕有别的备选项

2. Steepest ascent hill climbing:
   - generate all successor states S
   - run h() on all s ∈ S　　评估所有successor，找到最好的h(n)
   - among all successors s with better h() than current state n, take the successor s with the best h(n)

通常来说steppest比vanilla强，但是有时候你有很多child branching(导致复数h(n)用时久)，这时选Vanilla

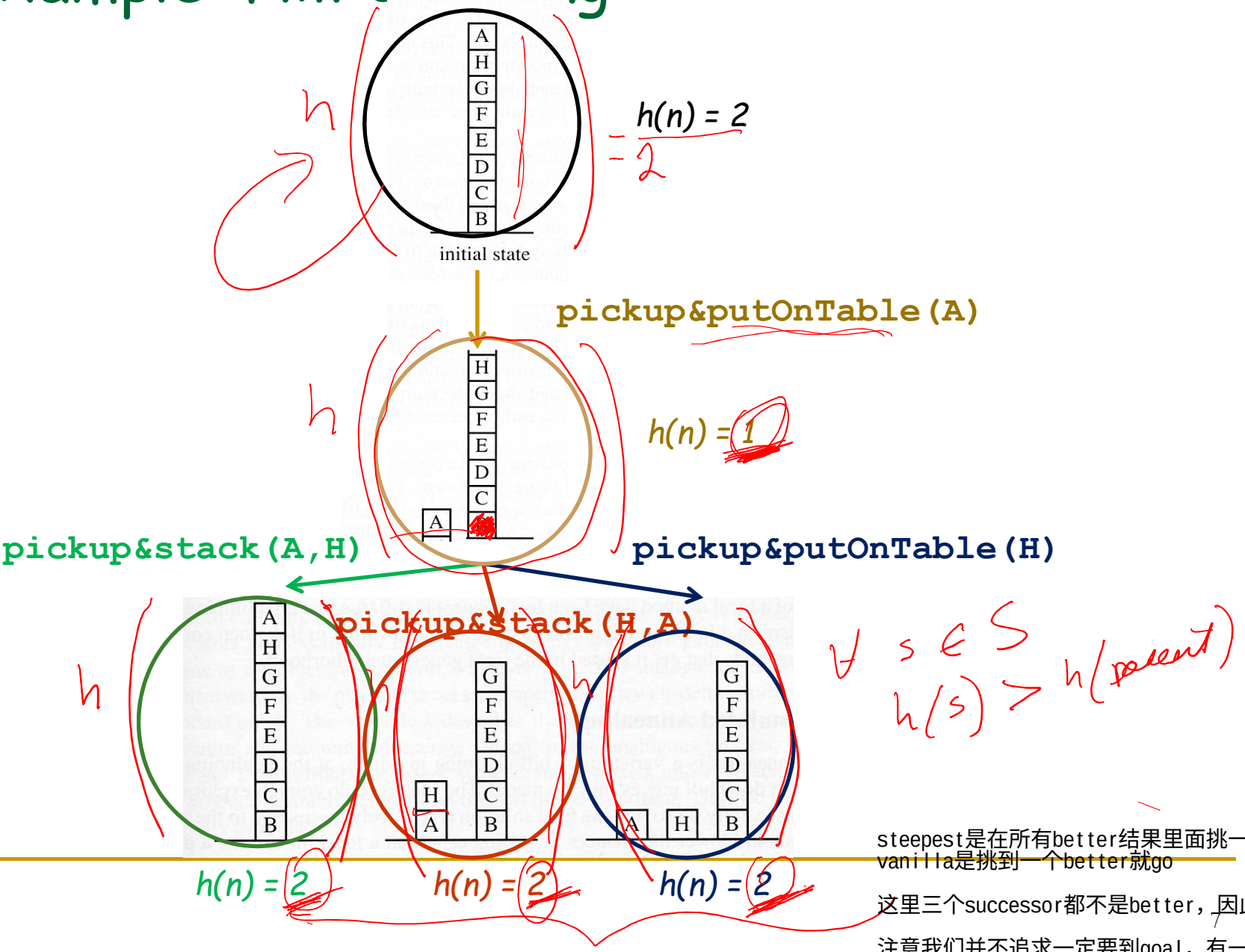
Example: Hill Climbing



- Operators:
  - pickup&putOnTable(Block) 从stack顶上取一个放到桌上
  - pickup&stack(Block1,Block2) 1顶上取一个放到2顶上
- Heuristic: 花最小step从initial state到goal state
  - 0pt if a block is sitting where it is supposed to sit
  - +1pt if a block is NOT sitting where it is supposed to sit
    +1如果不在应该在的相对位置上
  - so lower h(n) is better
    - h(initial) = 2 H在G上,G在F上.....但B不在A上，A不在地上，所以是2
    - h(goal) = 0 heurisitic的goal必须是0

source: Rich & Knight, Artificial Intelligence, McGraw-Hill College 1991.

Example: Hill Climbing



steepest是在所有better结果里面挑一个best
vanilla是挑到一个better就go

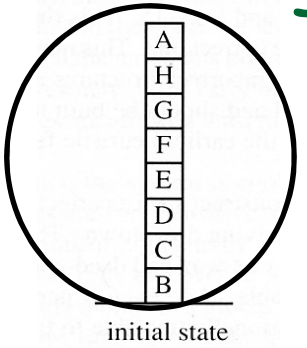
这里三个successor都不是better，因此会stop

注意我们并不追求一定要到goal，有一定优化也是好的

Example: Hill Climbing

hill-climbing，会停下

但有的算法，可以让你做到即使当前结果更差
但只是暂时的temperarily，为了达到最终goal
例如拧魔方，一面全红，为了六面完美
要打乱一面全红

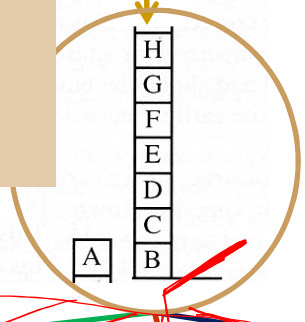


initial state

$h(n) = 2$

hill-climbing will stop, because all children have higher h(n) than the parent… --> local minimum

pickup&putOnTable(A)

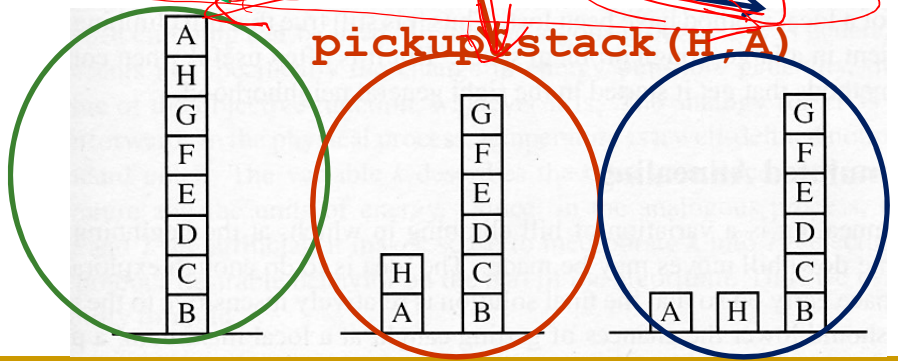


$h(n) = 1$

pickup&stack(A,H)

pickup&putOnTable(H)

pickup&stack(H,A)



$h(n) = 2$ $h(n) = 2$ $h(n) = 2$

Don't be confused…
a lower h(n) is better…

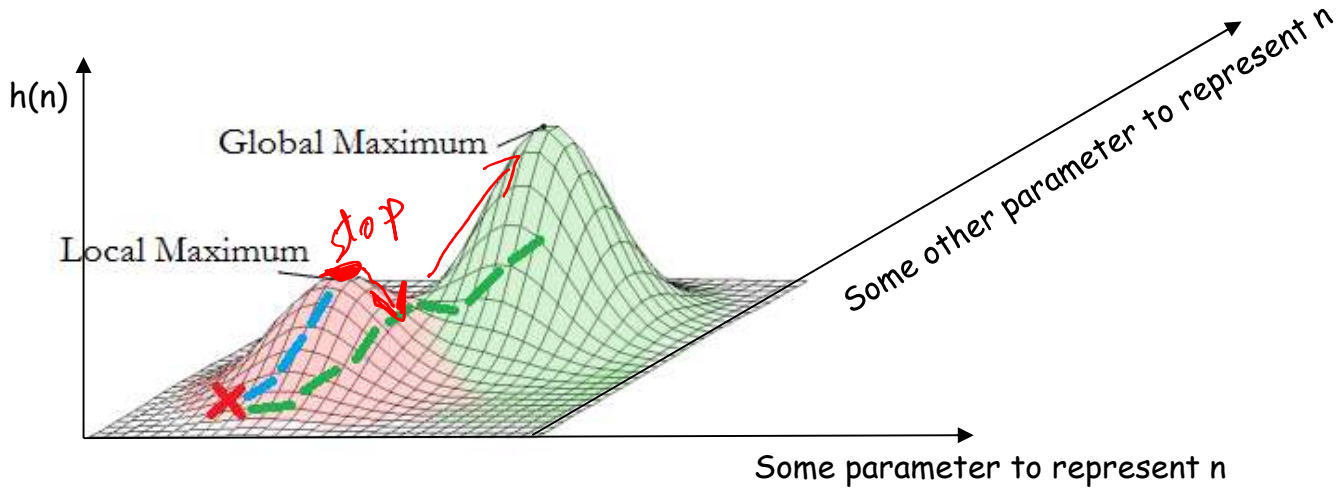
Steepest Ascent Hill Climbing

```
currentNode = startNode;
    loop do
        L = CHILDREN(currentNode);
        nextEval = +INFINITY;
        nextNode = NULL;

        for all c in L
           if (HEURISTIC-VALUE(c) < nextEval) // lower h is better
                nextNode = c;
                nextEval = HEURISTIC-VALUE(c);

          if nextEval >= HEURISTIC-VALUE(currentNode)
             // Return current node since no better child state exist
             return currentNode;

          currentNode = nextNode;
```

Adapted from https://en.wikipedia.org/wiki/Hill_climbing

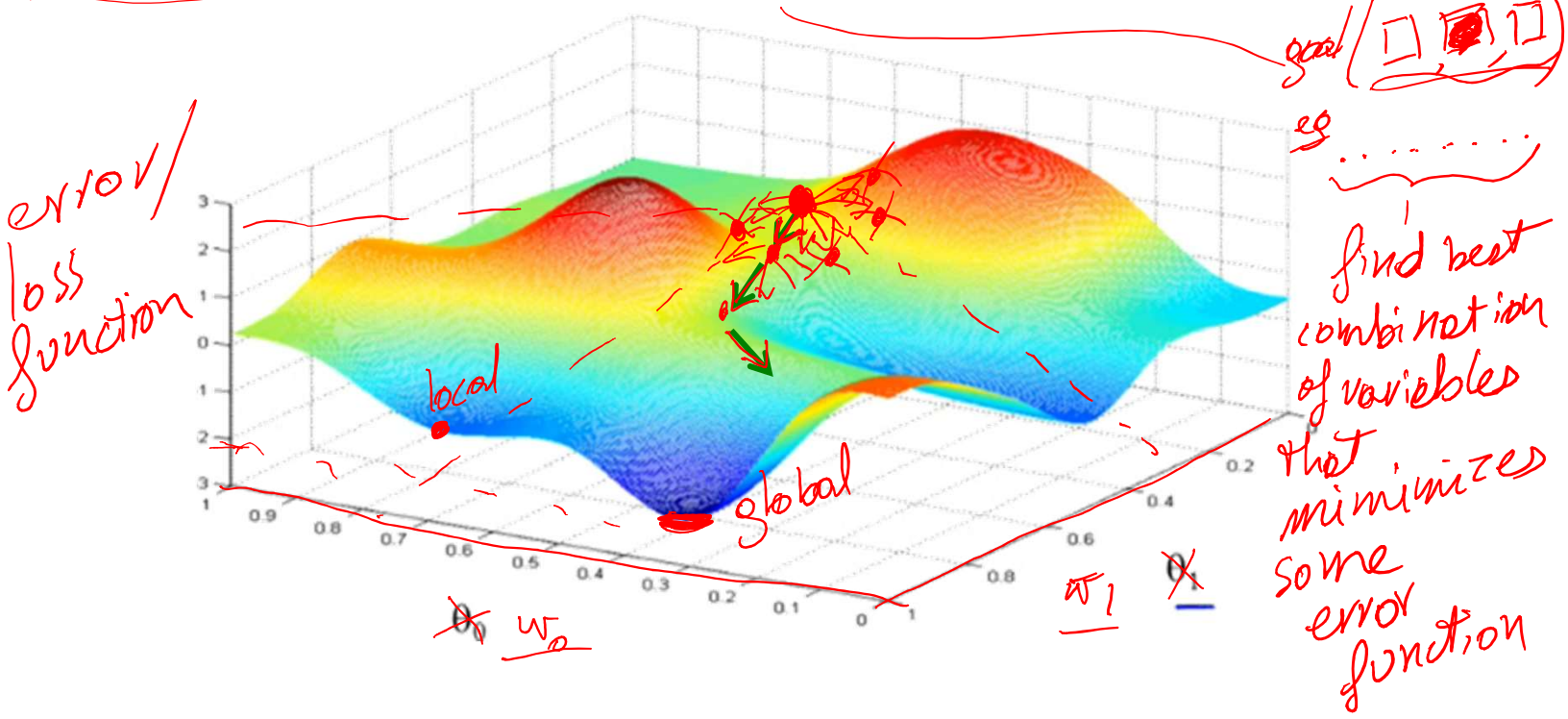
Problems with Hill Climbing

- Foothills (or local maxima)
  - reached a local maximum, not the global maximum
  - a state that is better than all its neighbors but is not better than some other states farther away.
  - at a local maximum, all moves appear to make things worse.
  - ex: 8-puzzle: we may need to move tiles temporarily out of goal position in order to place another tile in goal position



Use of Hill Climbing

- mostly for optimization problems
- i.e. goal defined not as a function of the state alone, but with respect of other states (eg. the best state I can reach)



Source: Andrew Ng

Today

1. State Space Representation ✓
2. State Space Search ✓
   a) Overview ✓
   b) Uninformed search ✓
      1. Breadth-first Search and Depth-first Search ✓
      2. Depth-limited Search ✓
      3. Iterative Deepening ✓
      4. Uniform Cost ✓
   c) Informed search ✓
      1. Intro to Heuristics ✓
      2. Hill climbing ✓
      3. Greedy Best-First Search
      4. Algorithms A & A*
      5. More on Heuristics
   d) Summary

h(n)

Problem with Hill-Climbing

- used mostly for optimization problems
  - where the goal state is defined with respect to other states
  - ex. shortest path, longest....
- if goal state is independent of other states
  - we should be able to backtrack, and find another path to the goal
  - i.e. we should use an OPEN list
  - i.e. Greedy Best First Search

Up Next

1. State Space Representation
2. State Space Search
   a) Overview
   b) Uninformed search
      1. Breadth-first and Depth-first
      2. Depth-limited Search
      3. Iterative Deepening
      4. Uniform Cost
   c) Informed search
      1. Intro to Heuristics
      2. Hill climbing
      3. Greedy Best-First Search
      4. Algorithms A & A*
      5. More on Heuristics
   d) Summary