

Hill climbing, simulated annealing and beam search

When only the final state matters, forget the search tree: keep one state, improve it, and learn when to accept a step backwards. Class Test 1 in the last 20 minutes.

01 COLD OPEN 0-3 min

Climbing Keokradong in thick fog

You are on a hill in Bandarban, in fog so thick you can only feel the slope under your feet. Goal: the highest point.

Rule: always take the step that goes up the most. Stop when every step goes down.

No map, no memory of where you have been, one position at a time. That is **local search**.

Only if you started on the right hill. You stop on the **first** top you reach, which may be a small hump with the real summit 2 km away. Today: three ways to do better, and why a little randomness helps.



POP CULTURE

Squid Game's glass bridge is the opposite problem: one wrong step and you are out, so there is no "try and step back". Local search lives on landscapes where you can always take another step.

02 CONCEPT 3-7 min

When the path doesn't matter

In 8-queens the answer is the final board, not the moves that led to it. Same for timetables, chip layouts and delivery routes. So:

- **Complete-state formulation:** every state is a full candidate solution (all 8 queens on the board).
- **Objective function:** a number to maximise (or a cost such as h = attacking pairs to minimise).
- **Neighbours:** states one small change away (move one queen in its column).
- Keep **one current state:** $O(1)$ memory, works on spaces far too big to search systematically.

LANDSCAPE VOCABULARY

Global maximum: the best state anywhere.

Local maximum: better than all its neighbours, but not the best.

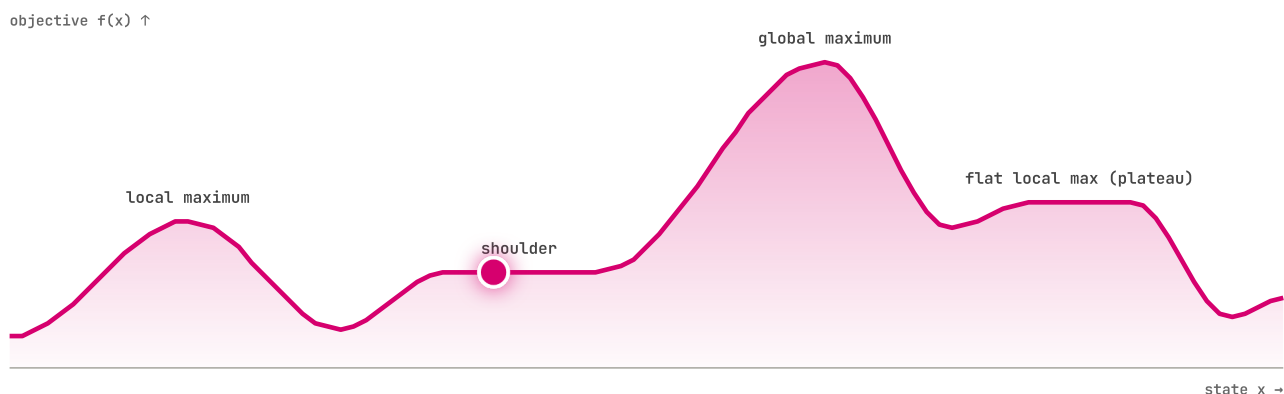
Plateau: a flat area; a **shoulder** is a plateau that still leads up.

Ridge: a sequence of local maxima that is easy to walk along in one direction but every single move goes down.

03 **LIVE LAB** 7-12 min

Live lab: a climber that only sees its feet

State-space landscape Click the ground to drop the climber (hill climbing and annealing), then Run. The climber only sees its two neighbours.



HEIGHT NOW

30

BEST SEEN (MAX 96)

30

STEPS

0

04 **CONCEPT** 12-16 min

Hill climbing and its variants

```
function HILL-CLIMBING(problem)
  current ← problem.INITIAL
  loop
    neighbour ← highest-valued successor of current
    if VALUE(neighbour) ≤ VALUE(current):
      return current          # a peak (maybe local)
    current ← neighbour
```

Steepest ascent, adapted from Russell & Norvig. Greedy: it never looks more than one step ahead.

VARIANT	CHANGE
Sideways moves	allow equal-valued moves (with a limit) to cross shoulders
Stochastic	pick a random uphill move, more likely the steeper ones
First-choice	try random neighbours, take the first that is better: good when there are thousands of neighbours
Random restart	if stuck, start again from a random state; keep the best result



PRO TIP

Random-restart hill climbing is complete *with probability approaching 1*: eventually a start lands in the global hill's basin.

05 WORKED EXAMPLE 16-21 min

Worked example: h for an 8-queens board

GIVEN

An 8-queens board in the complete-state formulation, one queen per column. Columns 1-8 from left to right, rows 1-8 from bottom to top. The queens are on rows (5, 6, 7, 4, 5, 6, 7, 6).

FIND

h (number of attacking pairs, counted once) and the best move for steepest-ascent hill climbing.

Method: Russell & Norvig's 8-queens hill-climbing example

SOLUTION

1. Write the state as rows, column 1 → 8 (row 1 at the bottom): **(5, 6, 7, 4, 5, 6, 7, 6)**. One queen per column, so only rows and diagonals can clash.
2. **Same row** (equal numbers): 1-5, 2-6, 2-8, 6-8, 3-7 → **5 pairs**.
3. **Same diagonal** ($|\text{row difference}| = \text{column distance}$): 1-2, 1-3, 2-3, 2-4, 3-5, 4-5, 4-6, 4-7, 5-6, 5-7, 6-7, 7-8 → **12 pairs**.

$$h = 5 + 12 = 17$$

4. Successors: move one queen within its column, $8 \times 7 = \mathbf{56}$ of them. Computing h for each (the numbers on the board in the lab), the lowest is **12**, reached by 8 different moves.
5. Steepest ascent picks one of the eight at random, e.g. column 2 to row 1: **(5, 1, 7, 4, 5, 6, 7, 6)**, $h = 12$. Repeat until no move lowers h.

We minimise h, so "hill climbing" here walks downhill. Same algorithm with value = -h.

ANSWER

$h = 17$ (5 row pairs + 12 diagonal pairs). Best successors have $h = 12$ (8 of them), e.g. column 2 $\rightarrow$ row 1.

06 LIVE LAB 21-27 min

Live lab: 8-queens, one move at a time

8-queens hill climbing h = attacking pairs. Click any square to move that column's queen; or let steepest ascent pick the lowest number.

8	14	14	13	17	12	14	12	18
7	18	14		15	15	14		16
6	17		16	18	15		15	
5		14	17	15		14	16	16
4	15	14	14		13	16	13	16
3	14	12	18	13	15	12	14	14
2	14	16	13	15	12	14	12	16
1	18	12	14	13	13	12	14	14
	1	2	3	4	5	6	7	8

h **17** attacking pairs

- ☐ allow sideways moves (≤ 100)
- ☒ show h of every move
- ☐ draw attacking pairs

STEPS

0

BEST MOVE GIVES

$h = 12$

The textbook board: $h = 17$. Every number is the h you would get by moving that column's queen there.

Your turn • 1: how many restarts?

1 EXAM CQ

Steepest-ascent hill climbing solves a random 8-queens board with probability $p = 0.14$, taking 4 steps when it succeeds and 3 when it gets stuck.

- (a) With random restarts, how many runs do you expect? (b) How many steps in total? (c) Redo (b) with sideways moves: $p = 0.94$, 21 steps when it succeeds, 64 when it fails.

SOLUTION

1. Each run succeeds with $p = 0.14$, so the expected number of runs is

$$\frac{1}{p} = \frac{1}{0.14} \approx 7.1 \text{ runs} \quad (\approx 6.1 \text{ failures} + 1 \text{ success})$$

2. Expected steps = failures $\times 3$ + the successful run $\times 4$:

$$6.14 \times 3 + 4 \approx 22 \text{ steps}$$

3. With sideways moves $p = 0.94$: about 1.06 runs, so about $0.06 \times 64 + 21 \approx 25$ steps. Similar work, but almost every single run succeeds.

ANSWER

≈ 7 runs and ≈ 22 steps without sideways moves; ≈ 25 steps with them. Either way 8-queens is solved in a few dozen moves.

Simulated annealing: sometimes step down

```
function SIMULATED-ANNEALING(problem, schedule)
  current ← problem.INITIAL
  for t = 1, 2, 3, ...
    T ← schedule(t)
    if T = 0: return current
    next ← a random successor of current
    ΔE ← VALUE(next) - VALUE(current)
    if ΔE > 0: current ← next
    else current ← next with probability eΔE/T
```

THE IDEA

Early on (T high) accept many bad moves: explore, jump out of small hills. Later (T low) accept almost none: settle on a peak.

If T is lowered **slowly enough**, SA finds the global optimum with probability approaching 1.



POP CULTURE

Named after annealing in metallurgy: heat a metal, then cool it slowly so its atoms settle into a low-energy crystal. A village kamar (blacksmith) quenching a da does the opposite: fast cooling makes it hard but brittle.

09 WORKED EXAMPLE 35–38 min

Worked example: how likely is a step down?

GIVEN

Simulated annealing, acceptance probability $e^{\Delta E/T}$ for a worse move ($\Delta E < 0$).

FIND

The acceptance probability for $\Delta E = -2$ and $\Delta E = -10$ at $T = 10, 2$ and 0.5 . What do the numbers say?

Method: Metropolis acceptance rule, as in Russell & Norvig's simulated annealing

SOLUTION

1. Minimising h , so a worse move has $\Delta E = -(h_{\text{new}} - h_{\text{old}}) < 0$. The acceptance rule:

$$P(\text{accept}) = e^{\Delta E/T}$$

2. $\Delta E = -2$ (slightly worse):

$$T = 10 : e^{-0.2} = 0.82 \quad T = 2 : e^{-1} = 0.37 \quad T = 0.5 : e^{-4} = 0.018$$

3. $\Delta E = -10$ (much worse):

$$T = 10 : e^{-1} = 0.37 \quad T = 2 : e^{-5} = 0.0067 \quad T = 0.5 : e^{-20} \approx 2 \times 10^{-9}$$

4. Read the table both ways: **bigger mistakes** are accepted less often, and **as T falls** every mistake becomes rarer. At $T \rightarrow 0$ only improvements pass: hill climbing.

ANSWER

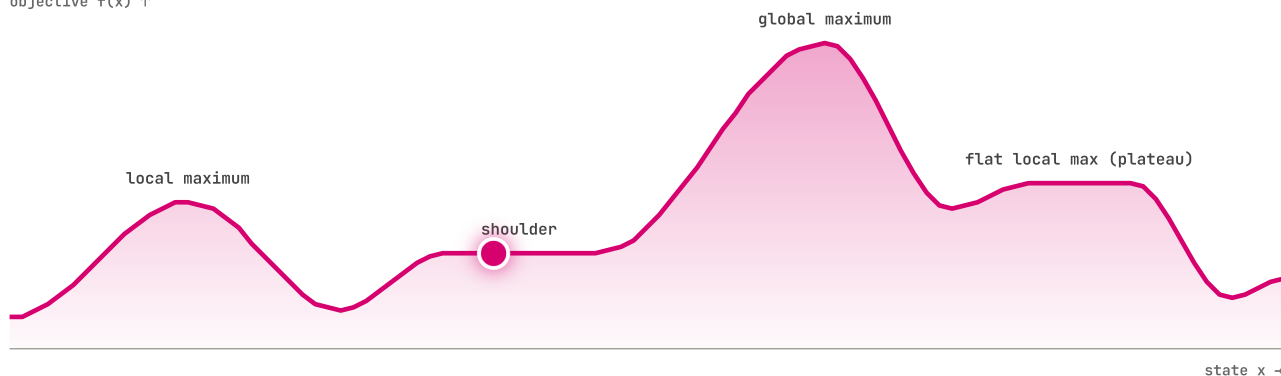
$\Delta E = -2$: 0.82, 0.37, 0.018. $\Delta E = -10$: 0.37, 0.0067, $\approx 2 \times 10^{-9}$. Small mistakes survive longer; cooling kills all of them.

Live lab: annealing on the landscape

State-space landscape Click the ground to drop the climber (hill climbing and annealing), then Run. The climber only sees its two neighbours.

start temperature $T_0 = 30$

objective $f(x) \uparrow$



HEIGHT NOW

30

BEST SEEN (MAX 96)

30

STEPS

0

TEMPERATURE T

30.00

LAST DOWNHILL MOVE

—

Critical thinking •

tuning the temperature



TRAP

A classmate runs SA with $T_0 = 0.1$ and $\alpha = 0.5$ (T halves every step) on the landscape and says: "Annealing is no better than hill climbing, it got stuck on the first hill."

What happened? What does SA become when T is very large, and when T is very small? How would you choose T_0 ?

SOLUTION

1. **T huge:** $e^{\Delta E/T} \approx e^0 = 1$, so every move is accepted: a **random walk**. It explores but never commits; with too slow cooling it wastes time.
2. **T tiny (or cooled too fast):** $e^{\Delta E/T} \approx 0$, so only improvements pass: **hill climbing** with random neighbours. It freezes on the nearest peak.
3. The classmate's schedule freezes in a few steps: annealing with that schedule is just hill climbing, so the "improvement" is no improvement.
4. Rule of thumb: pick T_0 so that a typical bad move is accepted about half the time at the start, and cool slowly (α close to 1).

| e.g. if typical bad moves have $\Delta E \approx -5$, $T_0 \approx 5/\ln 2 \approx 7$ gives $P \approx 0.5$.

ANSWER

Too hot = random walk; too cold or too fast = hill climbing. Set T_0 so bad moves are often accepted at first, then cool slowly.

12 **CONCEPT** 46–50 min

Local beam search: k climbers that talk

- Start with **k random states**.
- Each step: generate **all successors of all k**; if one is a goal, stop.
- Otherwise keep the **best k from the whole pool** and repeat.

Unlike k independent restarts, information flows: if one state finds a great hill, its successors take over the beam and the climbers on poor hills are dropped.



GOTCHA

The same feature is the weakness: all k can crowd onto one hill and lose diversity. **Stochastic beam search** picks the k successors at random, weighted by value. Add "parents combine" and you get **genetic algorithms** (Week 5).

TRY IT

In the landscape lab, choose **Local beam (k = 4)** and press *New random start* a few times. Four climbers start; within a few steps the beam usually sits on one hill.

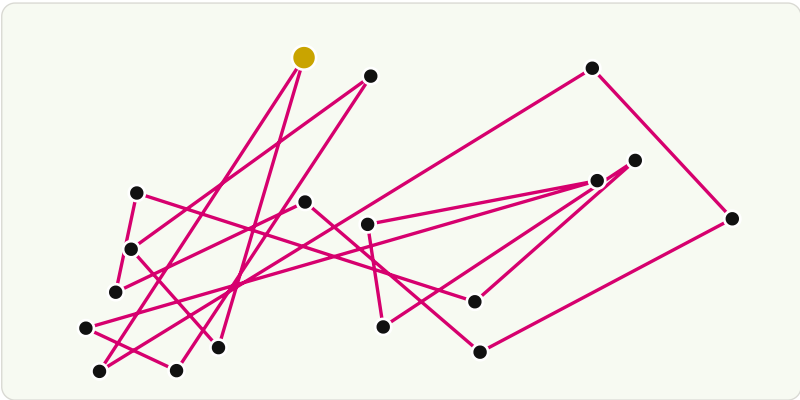
	MEMORY	SHARES INFORMATION?
Hill climbing	1 state	—
k random restarts	1 at a time	No
Local beam (k)	k states	Yes

13 REAL WORLD 50–55 min

Real world: routes, timetables and chips

Annealing a delivery-drone route 18 drop-off points. A move reverses one stretch of the route (2-opt). Hill climbing keeps only improvements; annealing sometimes accepts a longer route.

$T_0 = 8$ km



ROUTE LENGTH

189.3 km

BEST THIS SESSION

—

MOVES TRIED

0

TEMPERATURE

8.00

ROUTE LENGTH OVER TIME

14 POP QUIZ 55–58 min

Pop quiz

1. Local search keeps in memory...

- A. The whole search tree
- B. The frontier and the explored set
- C. Only the current state (or k states) ✓
- D. Every state ever visited

That is why it can work on huge spaces (8-queens, timetables, TSP): memory is constant. The price: no path, no guarantee of the best answer.

2. Steepest-ascent hill climbing on a flat plateau...

- A. Speeds up
- B. Stops, because no neighbour is strictly better ✓**
- C. Always finds the global maximum
- D. Turns into BFS

With a strict "better than current" rule it halts. Allowing a limited number of sideways moves lets it cross shoulders (8-queens: 14 % → 94 % solved).

3. In simulated annealing, a move with $\Delta E = -3$ at $T = 3$ is accepted with probability...

- A. 0
- B. $e^{-1} \approx 0.37$ ✓**
- C. e^{-9}
- D. 1

$$P = e^{\Delta E/T} = e^{-3/3} = e^{-1}.$$

4. As the temperature T approaches 0, simulated annealing behaves like...

- A. A random walk
- B. Hill climbing (only improvements accepted) ✓**
- C. BFS
- D. Local beam search

$e^{\Delta E/T} \rightarrow 0$ for every $\Delta E < 0$. At very high T it is the opposite: almost every move is accepted, a random walk.

5. How is local beam search with k states different from k independent random restarts?

- A. It uses less memory
- B. The k states share information: the best successors of all of them are kept ✓**
- C. It never gets stuck
- D. There is no difference

Each step pools all successors and keeps the best k , so effort moves to where progress is being made. The risk: the k states can crowd onto one hill (stochastic beam search helps).

15 **RECAP** 58-60 min

Three things to remember

01

Local search keeps one complete state and moves to neighbours: $O(1)$ memory, no path, no guarantee. Problems: local maxima, plateaus, ridges.

02

Hill climbing takes the best neighbour until none is better. Fixes: sideways moves, random restarts (8-queens: 14 % → 94 % → ≈ 100 %).

03

Simulated annealing accepts bad moves with $e^{\Delta E/T}$, cooling T slowly. **Local beam** keeps the best k of all successors.



PRO TIP

Before next class: **Assignment 1 is due Week 4, Class 1.** Next week: informed search. Heuristics, greedy best-first and A*.

16 **RECAP** 60-80 min

Class Test 1

Class Test 1 • 20 minutes

COVERS

Problem formulation · BFS and DFS · UCS · DLS and IDS (traces and the comparison table) · local search basics (hill climbing, annealing acceptance, beam search).

RULES

Closed book, written, answer all questions on the paper. Calculators allowed, phones away. Write your tie-breaking and goal-test convention at the top of every trace.

Questions are handed out on paper. The practice set is on the course page under Assessments.

S. M. Sakeef Sani

Adjunct Lecturer, CSE · Northern University Bangladesh

smsakeefsani.hbitcorp.com