Introduction to Artificial Intelligence

DA 221

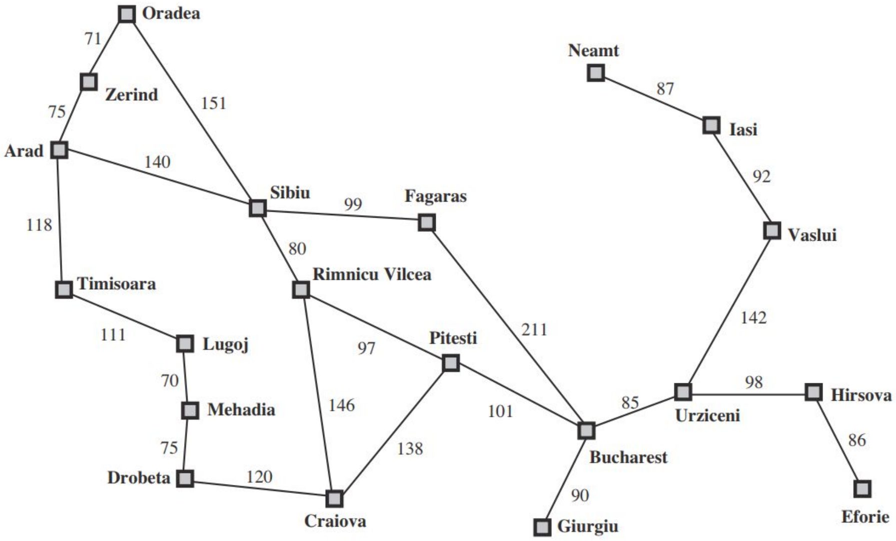
Jan - May 2023

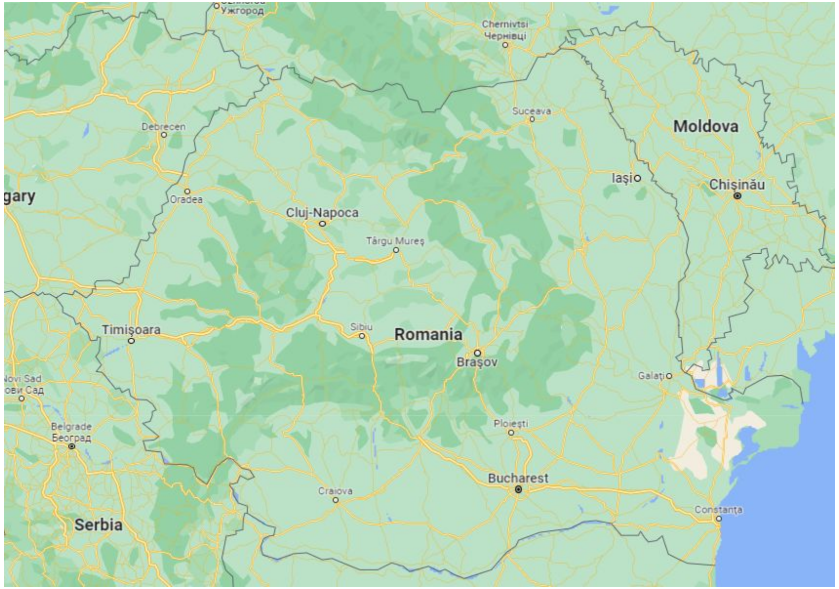
IIT Guwahati

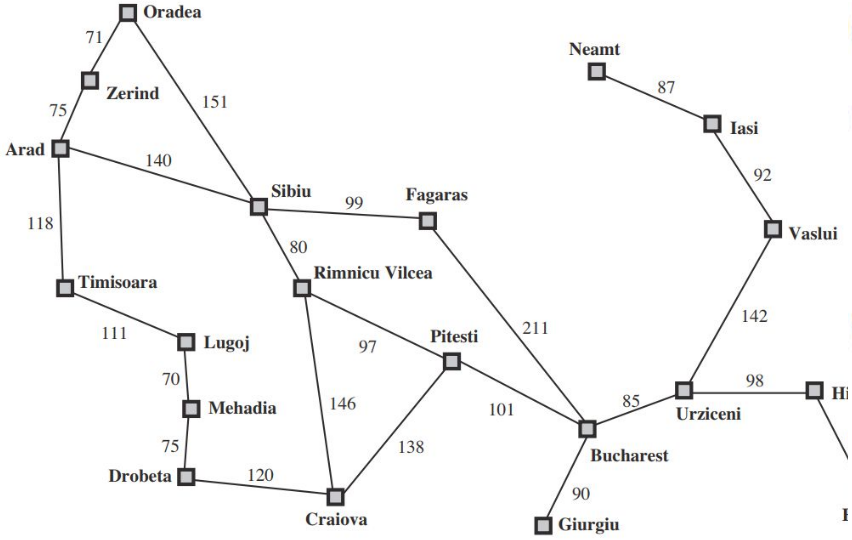
Instructors: Neeraj Sharma (& Arghyadip Roy)

Lecture 08: Neeraj Sharma

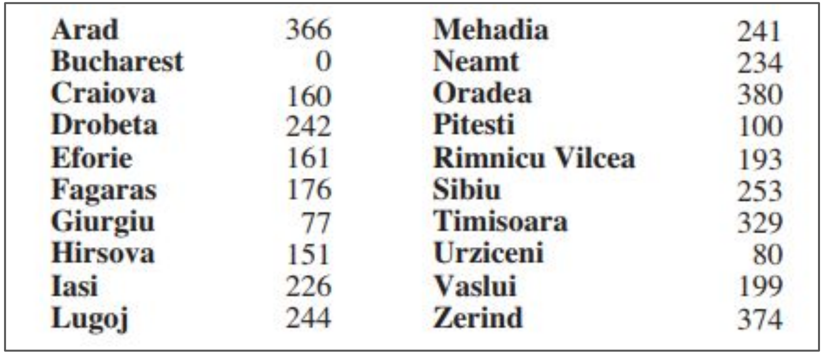
Informed (Heuristic) Search Strategies



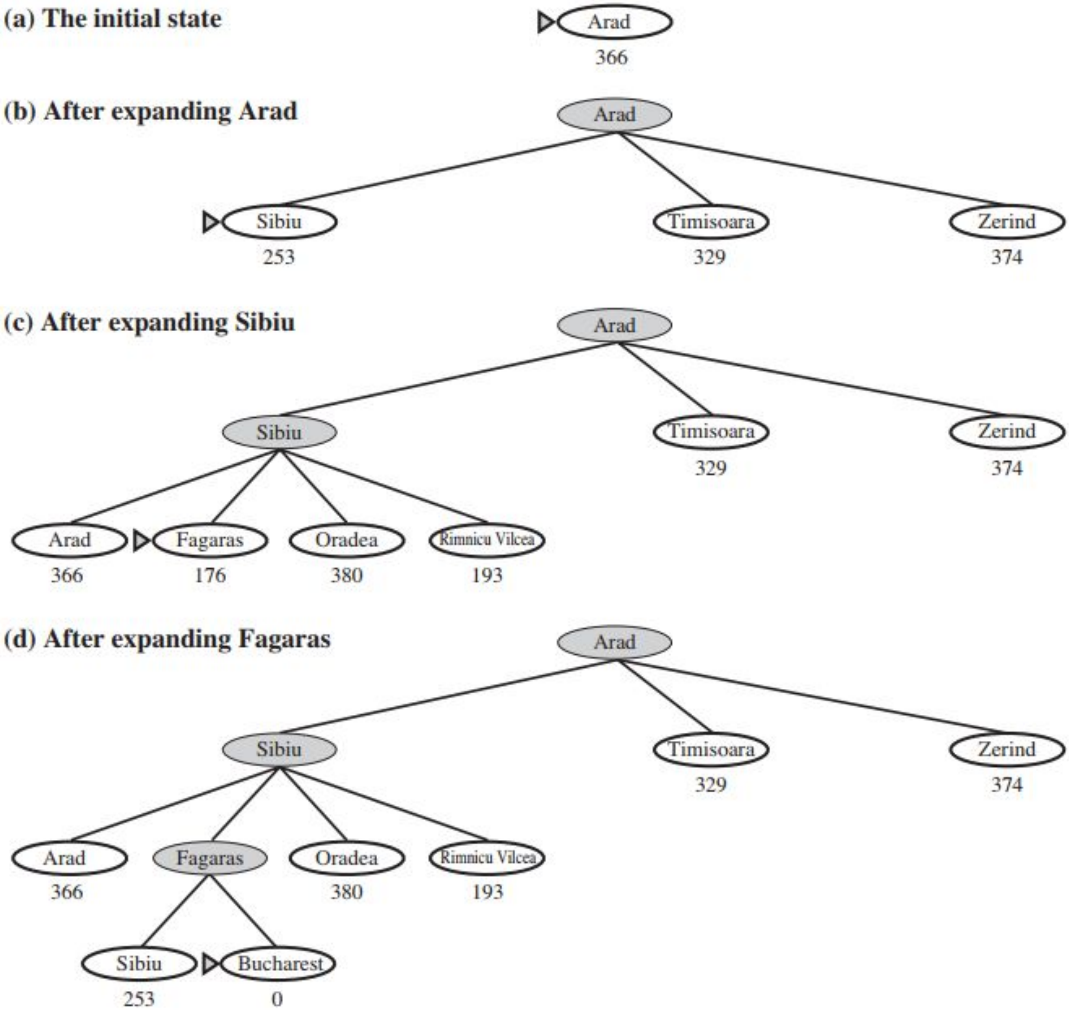




Heuristic



Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Drobeta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374



(a) The initial state

(b) After expanding Arad

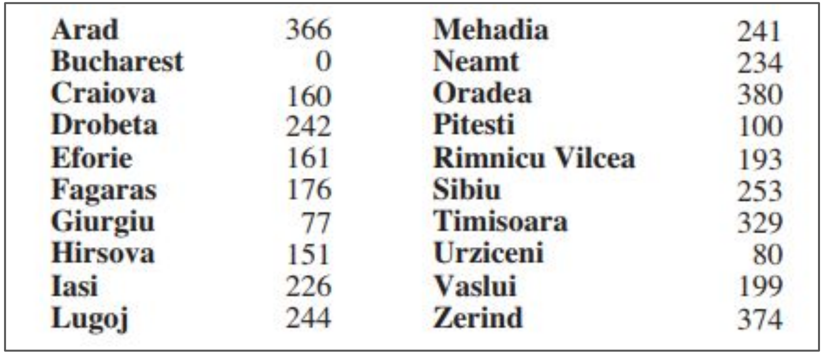
(c) After expanding Sibiu

(d) After expanding Fagaras

Informed (Heuristic) Search Strategies

- Greedy best-first search
- A* search

Heuristic



Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Drobeta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374

Local Search Algorithms

- Problem: observable, deterministic, known environments
- Solution: sequence of actions

Local Search Algorithms

- Problem: observable, deterministic, known environments
- Solution: sequence of actions

What happens when these requirements/assumptions are relaxed?

Local Search Algorithms

- Problem: observable, deterministic, known environments
- Solution: sequence of actions

What happens when these requirements/assumptions are relaxed?

Local Search: evaluating (and modifying) current state only; no systematic exploration of paths

Local Search Algorithms

Local Search: evaluating (and modifying) current state only; no systematic exploration of paths

Solution: a state; and not the path to reach the state

- Path to goal is irrelevant
- 8 queens problem, integrated circuit design, job-shop scheduling, and many more.

Local Search Algorithms

Local Search: evaluating (and modifying) current state only; no systematic exploration of paths

Solution: a state; and not the path to reach the state

- Path to goal is irrelevant
- Operate using a single current node (rather than multiple paths)
- Move only to a “neighbor” node

Local Search Algorithms

Local Search: evaluating (and modifying) current state only; no systematic exploration of paths

Solution: a state; and not the path to reach the state

- Path to goal is irrelevant
- Operate using a single current node (rather than multiple paths)
- Move only to a “neighbor” node

- Use little memory - a constant amount
- Reasonable solution in finite time

Local Search Algorithms

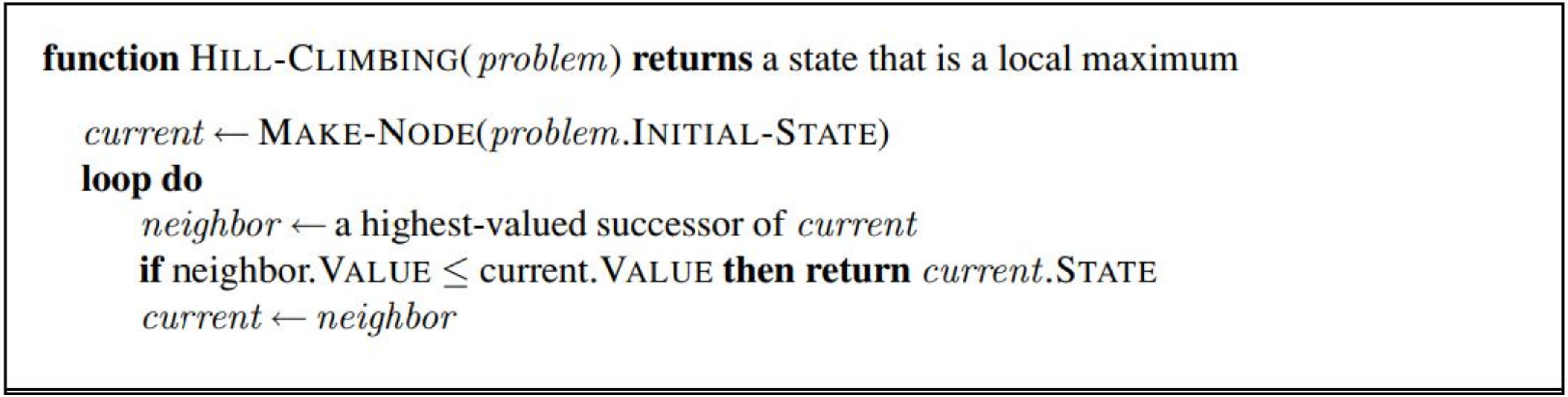
Local Search: evaluating (and modifying) current state only; no systematic exploration of paths

Solution: a state; and not the path to reach the state

- Path to goal is irrelevant
- Operate using a single current node (rather than multiple paths)
- Move only to a “neighbor” node

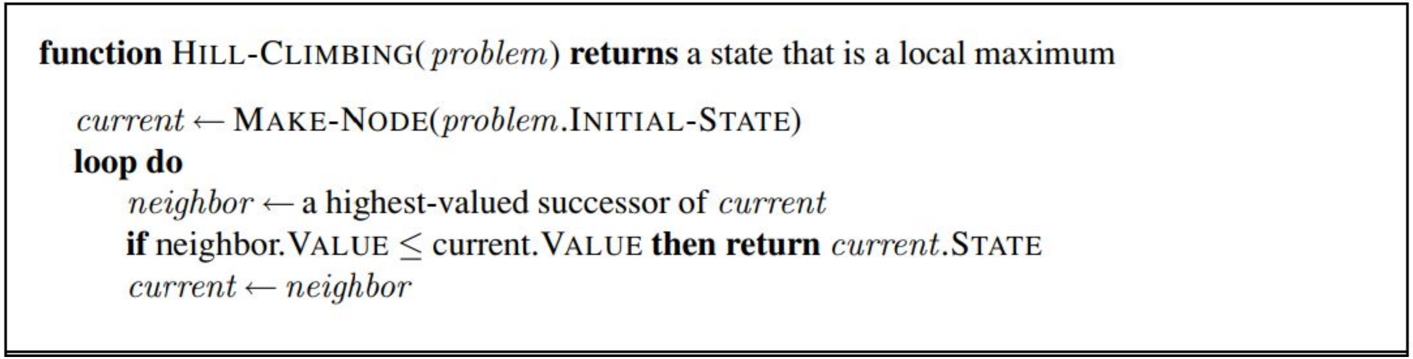
- Use little memory - a constant amount
- Reasonable solution in finite time

Hill-climbing Search

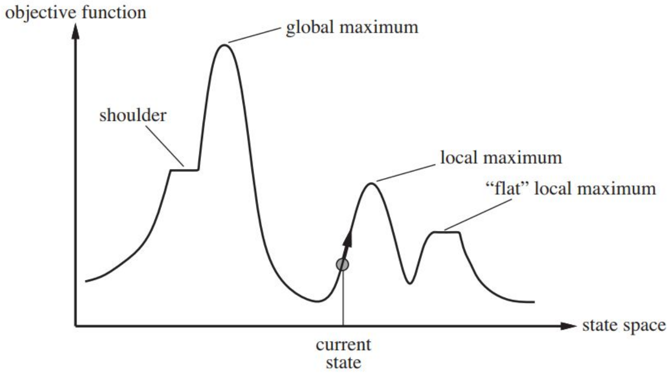


```
function HILL-CLIMBING(problem) returns a state that is a local maximum

  current ← MAKE-NODE(problem.INITIAL-STATE)
  loop do
      neighbor ← a highest-valued successor of current
      if neighbor.VALUE $\leq$ current.VALUE then return current.STATE
      current ← neighbor
```

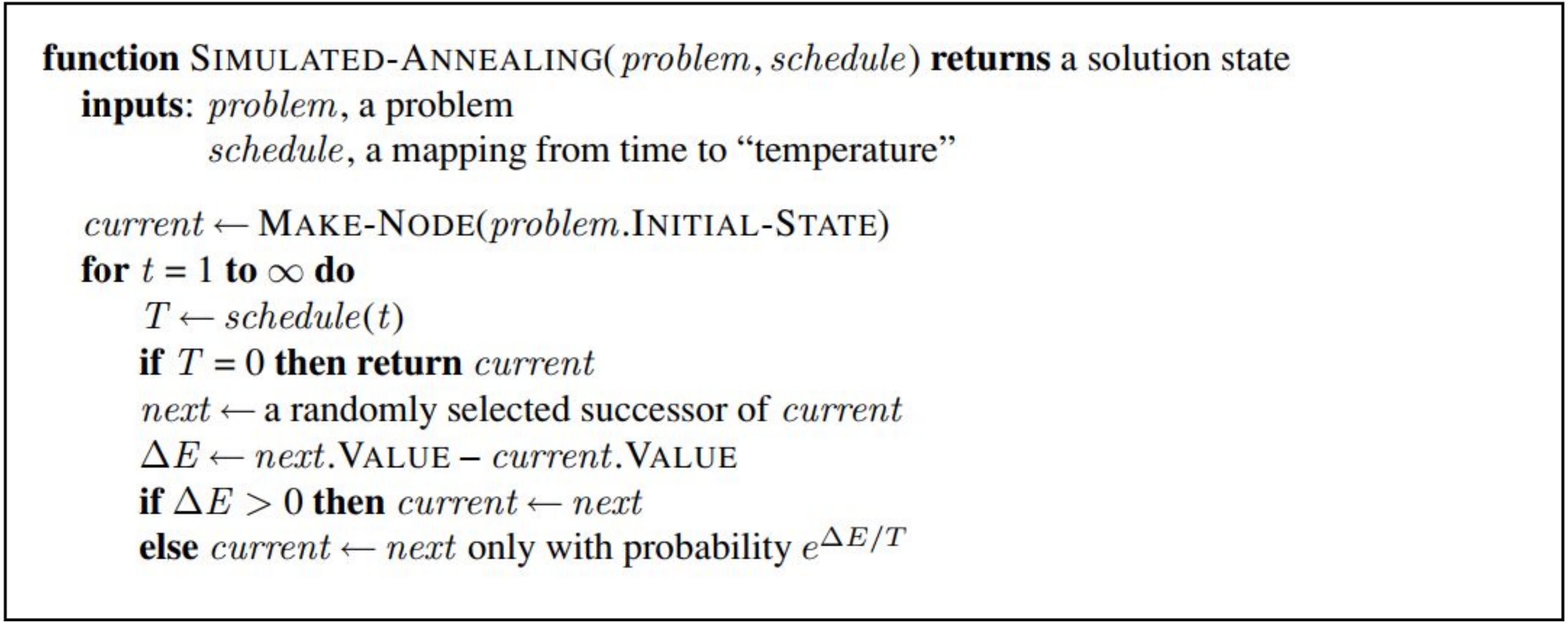
Hill-climbing Search



```
function HILL-CLIMBING(problem) returns a state that is a local maximum

  current ← MAKE-NODE(problem.INITIAL-STATE)
  loop do
    neighbor ← a highest-valued successor of current
    if neighbor.VALUE ≤ current.VALUE then return current.STATE
    current ← neighbor
```



Simulated Annealing



```
function SIMULATED-ANNEALING(problem, schedule) returns a solution state
   inputs: problem, a problem
           schedule, a mapping from time to “temperature”

   current ← MAKE-NODE(problem.INITIAL-STATE)
   for $t$ = 1 to $\infty$ do
       $T \leftarrow schedule(t)$
       if $T = 0$ then return current
       next ← a randomly selected successor of current
       $\Delta E \leftarrow next$.VALUE – current.VALUE
       if $\Delta E > 0$ then current ← next
       else current ← next only with probability $e^{\Delta E/T}$
```

Local Beam Search

Genetic Algorithms