Masayu Leylia Khodra

KK IF – Teknik Informatika – STEI ITB

Modul 3: Beyond Classical Search

Classical vs Local Search

Inteligensi Buatan
(*Artificial Intelligence*)

Classical Search

transisi state → kalau ada aksi, selama bisa sampe goal, dia pindah state

Problem

semua state sudah diketahui

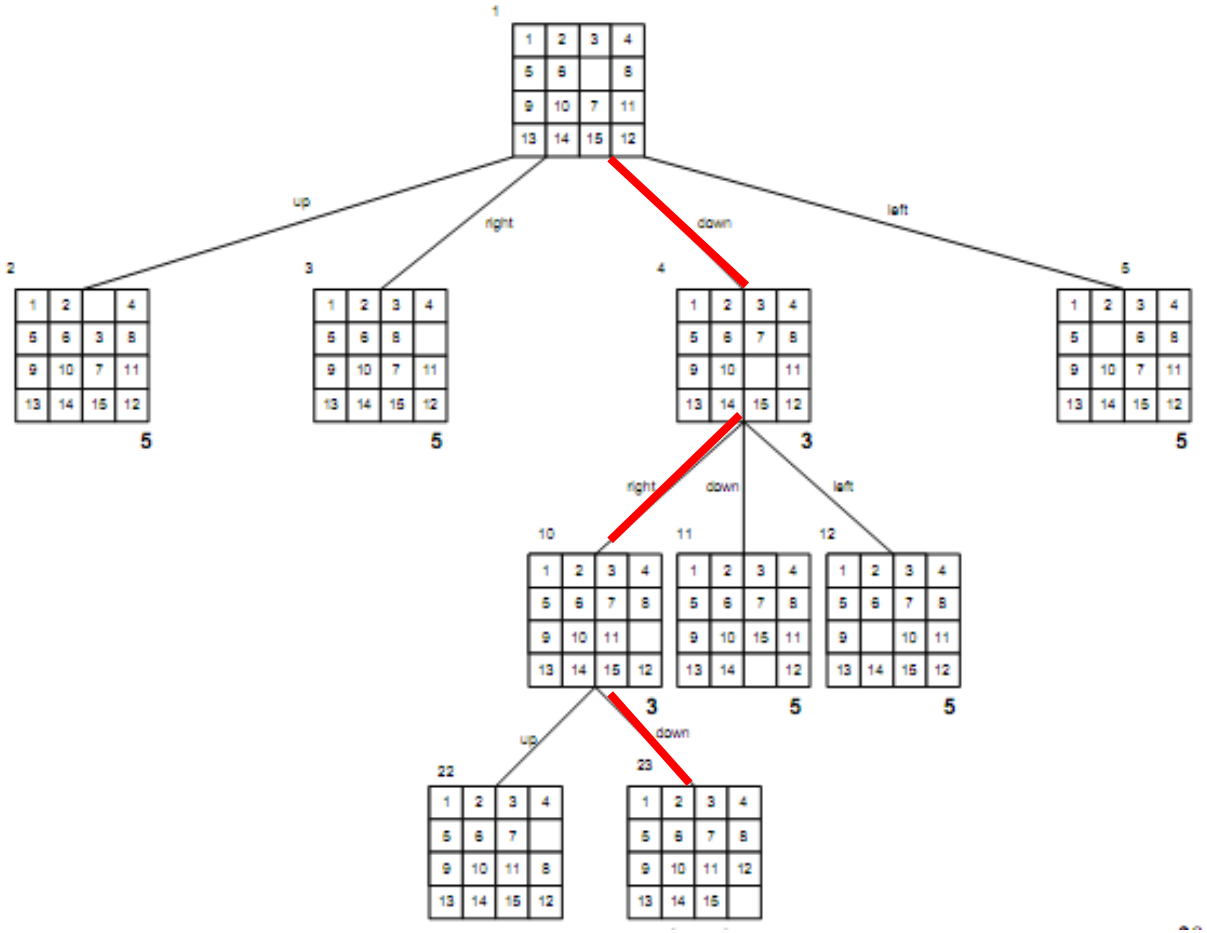
Known environment (observable, deterministic)
States (initial), operators, path cost, goal test

Explore search space systematically

Solution (path to goal)

Solution: sequence of actions

N-Puzzle Problem



Solution: down → right → down

Path to Goal as Solution

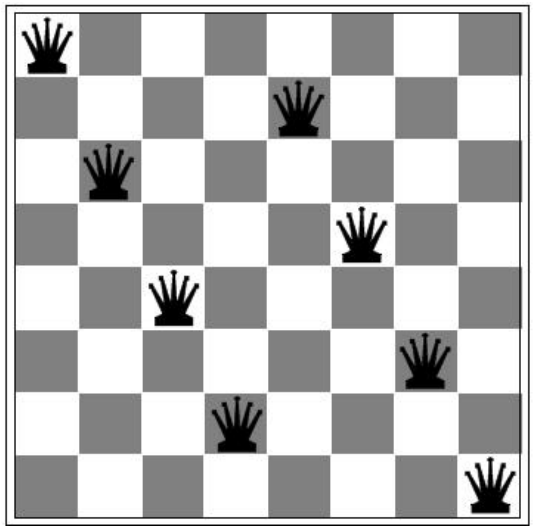
- In many problems, the path to goal is irrelevant.
- Solution: X=(8,6,4,2,7,5,3,1)
- Q1=8 → Q2=6 → … → Q8=1

path relevant : path jadi bagian dari solusi
contoh : rubiks, water jug.

path irrelevant : path dari initial state ke goal state tidak menjadi bagian dari solusi. contoh : N-queens, graph-coloring, knapsack, cryptarithmetic (contoh huruf mana aja dulu doesn't matter)

variable : setiap karakter
↳ nilai yg diassign itu angka

N-Queen Problem



objective : menempatkan queen ke bidak agar tidak bisa saling makan.

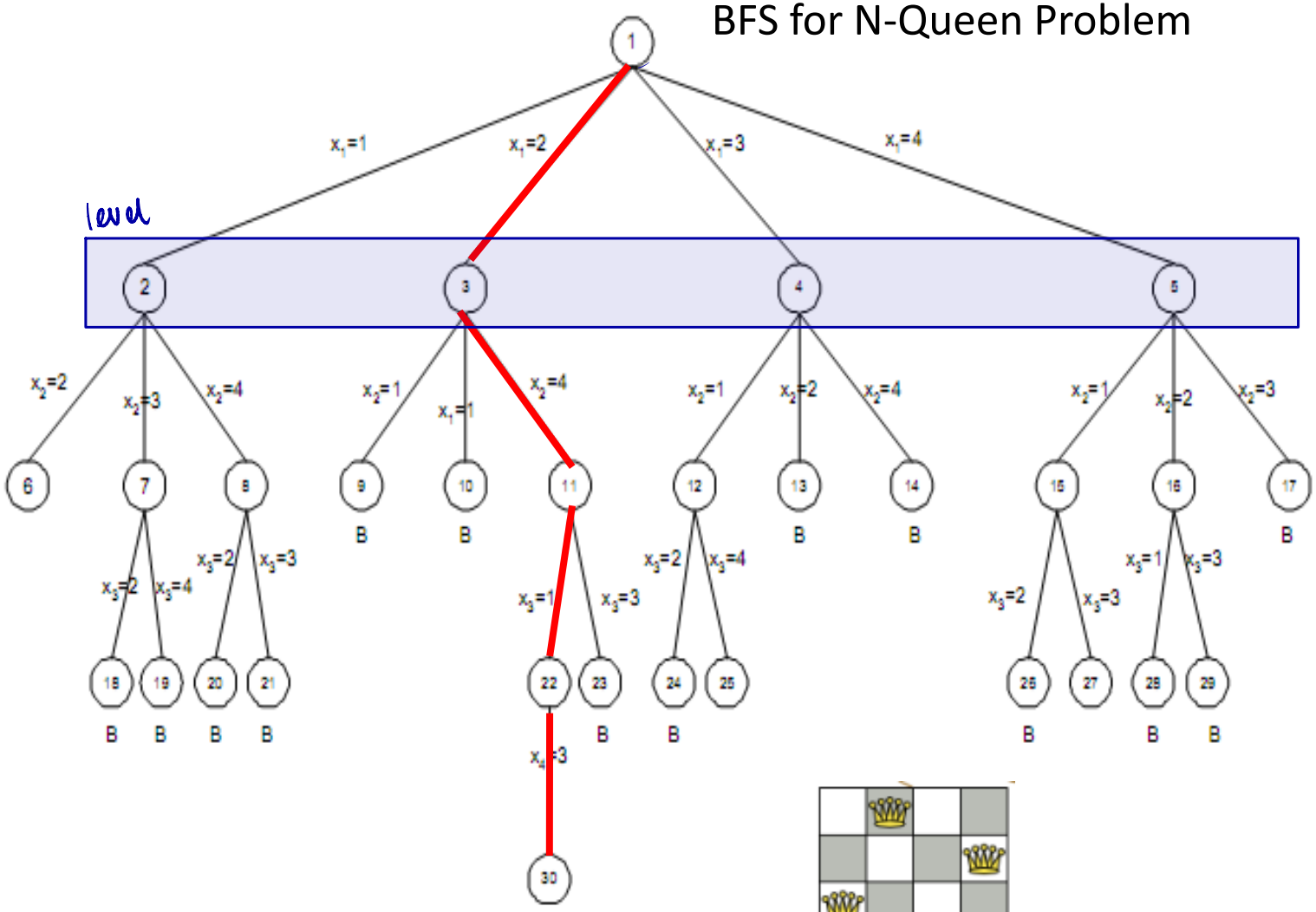
variabel : queen1, queen2, ... queen-N.
↳ nilai yg di assign ke variabel itu posisinya.

n-queens are on the board, none attacked. one queen per column

Path to Goal as Solution (2)

- For example, in n-queen
  - BFS solution: x1=2 → x2=4 → x3=1 → x4=3
- What matters is the final configuration of queens, not the order in which they are added.

BFS for N-Queen Problem



First solution 4-queen: X=(2,4,1,3)

Local Search

Jika path ke goal nya TIDAK PENTING (path irrelevant)

pada initial state, complete configuration → tiap variable sudah diisi nilai secara random, pasti ada nilai dlm state. tiap statenya jd perbaiki nilai var nya smpe dh ga ada yg melanggar constraints

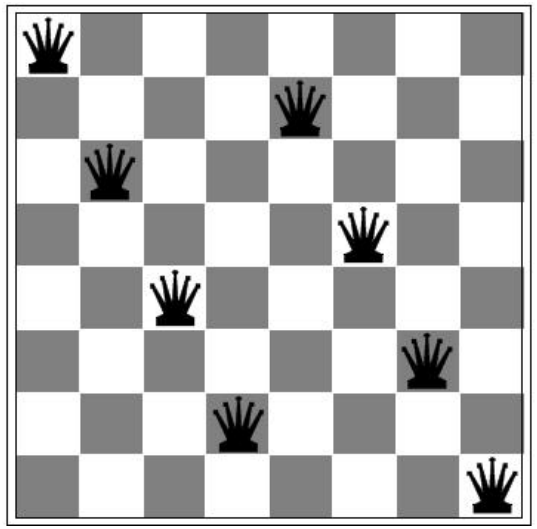
- If path to goal does not matter, we might consider different class of algorithms.
- Local search: complete state formulation
  - State: "complete" configurations
  - Keep single current state, not paths.
  - Action: move only to neighbors of current state.
  - No path cost → state value: value according to objective function or heuristic cost function
  - No goal test → maximum state value (berhenti pas nilai statenya sudah maksimum)
  - Solution: final state.

Problem
States (initial), neighbors
Path to goal is irrelevant

Find the best state by moving to neighbors of current state

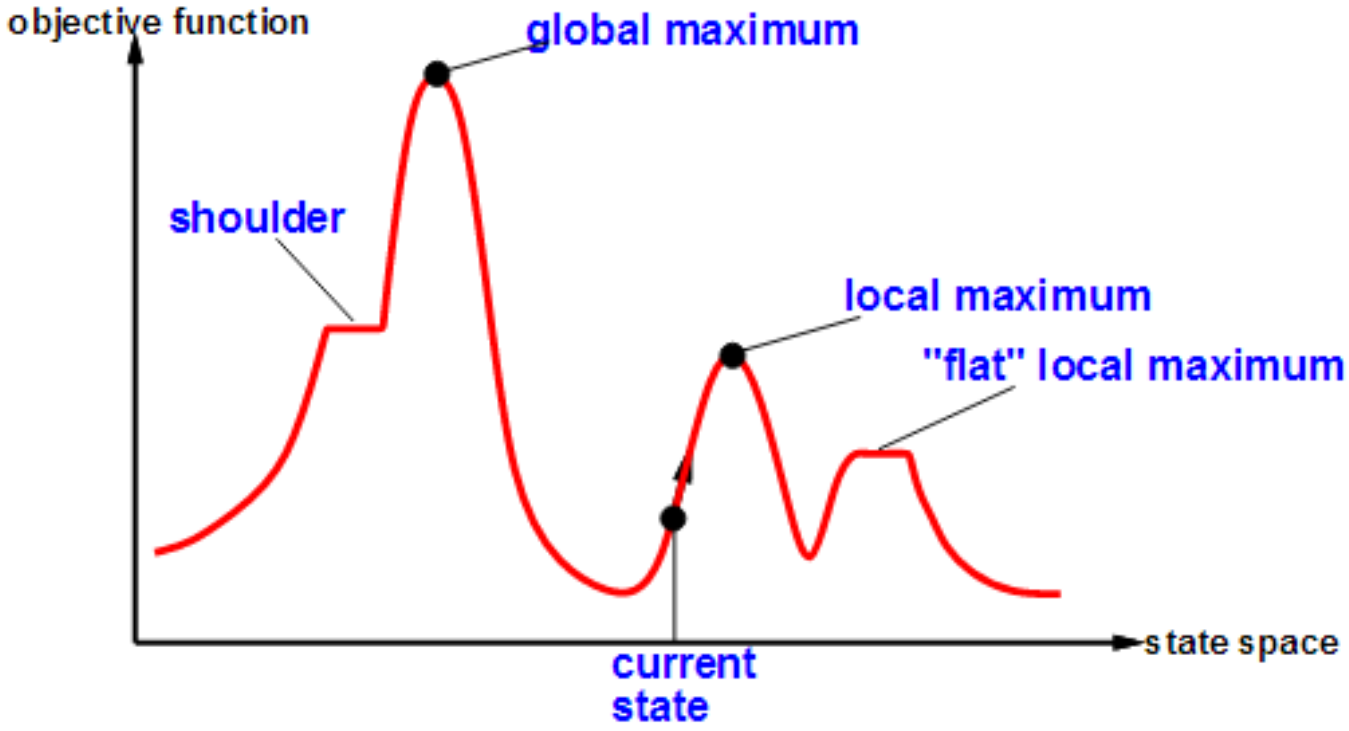
Solution
Final state/configuration

N-Queen: Classical vs Local Search



Classical search	Local search
State: any arrangement of **0 to n** queens on the board (incremental)	State: any arrangement of **n** queens on the board (complete)
Initial state: no queens on the board. Goal test: n-queens are on the board, none attacked	Initial state: a [random] state. Goal test: state value = global maximum
Action: add a queen to any empty square	Action: move a single queen to another square (neighbor)
Solution: path to goal	Solution: final state

Local Search Explore State-space Landscape



A one-dimensional state-space landscape in which elevation corresponds to the objective function. The aim is to find the global maximum. (Russel & Norvig, 2010)

- A landscape has "location" (defined by state) and "elevation" (value of objective or heuristic cost value).
- Local search aims to find global optimum.
- Problem: depending on initial state, can get stuck in local optimum.

Summary: Local Search

Keep single current state
State: "complete" configurations
Complete formulation

Find the best state (global optimum)
Action: move to neighbors

Path to goal is irrelevant
Solution: final state

Next:

- State
- Successor
- Neighbors

Masayu Leylia Khodra

KK IF – Teknik Informatika – STEI ITB

Modul 3: Beyond Classical Search

State: Value, Successor, and Neighbor

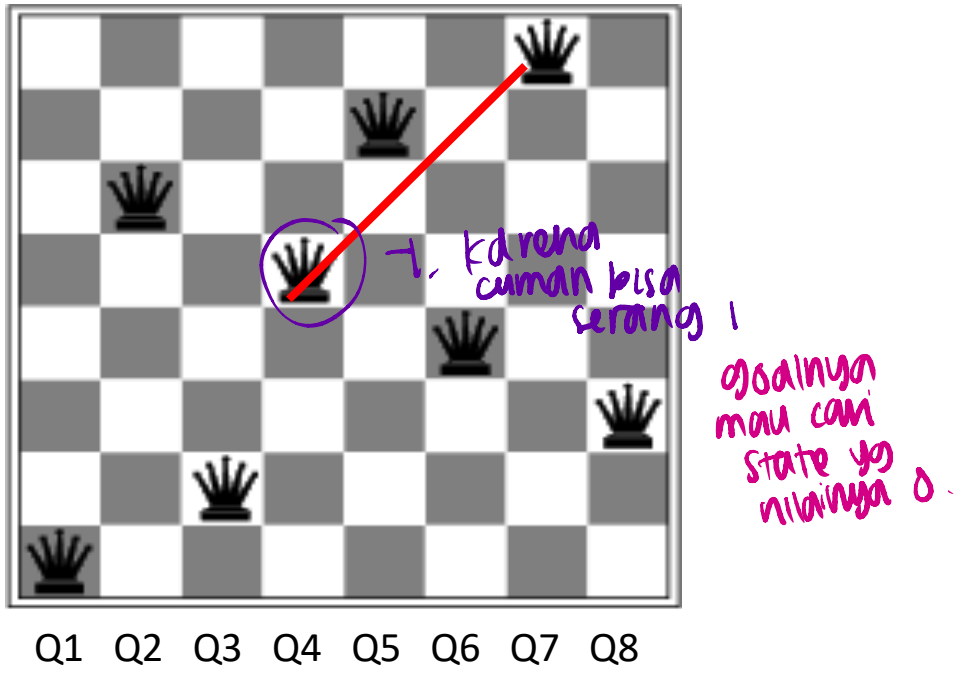
Inteligensi Buatan
(*Artificial Intelligence*)

Local Search: State

Keep single current state
State: "complete" configurations (one queen per column)
Solution: final state

Find the best state (global optimum)
Action: move to neighbors

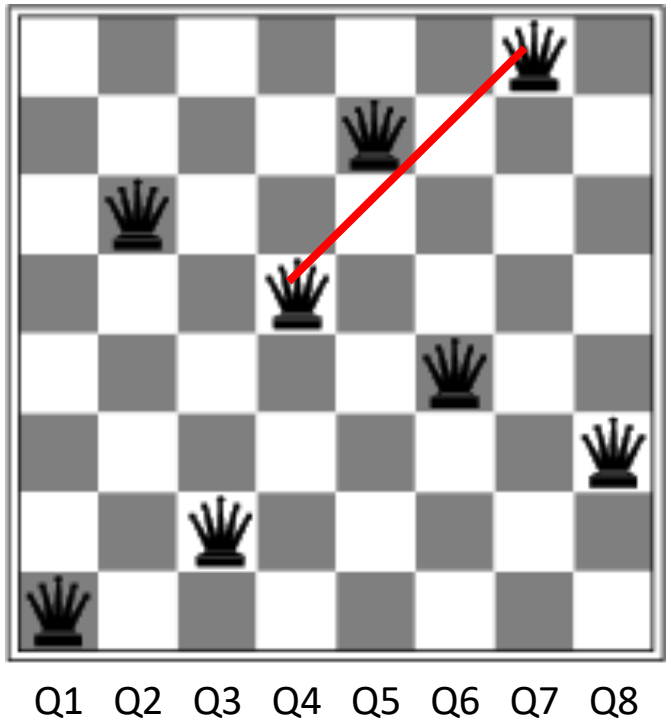
State for 8-queens problem



Each state has state value based on heuristic cost function.

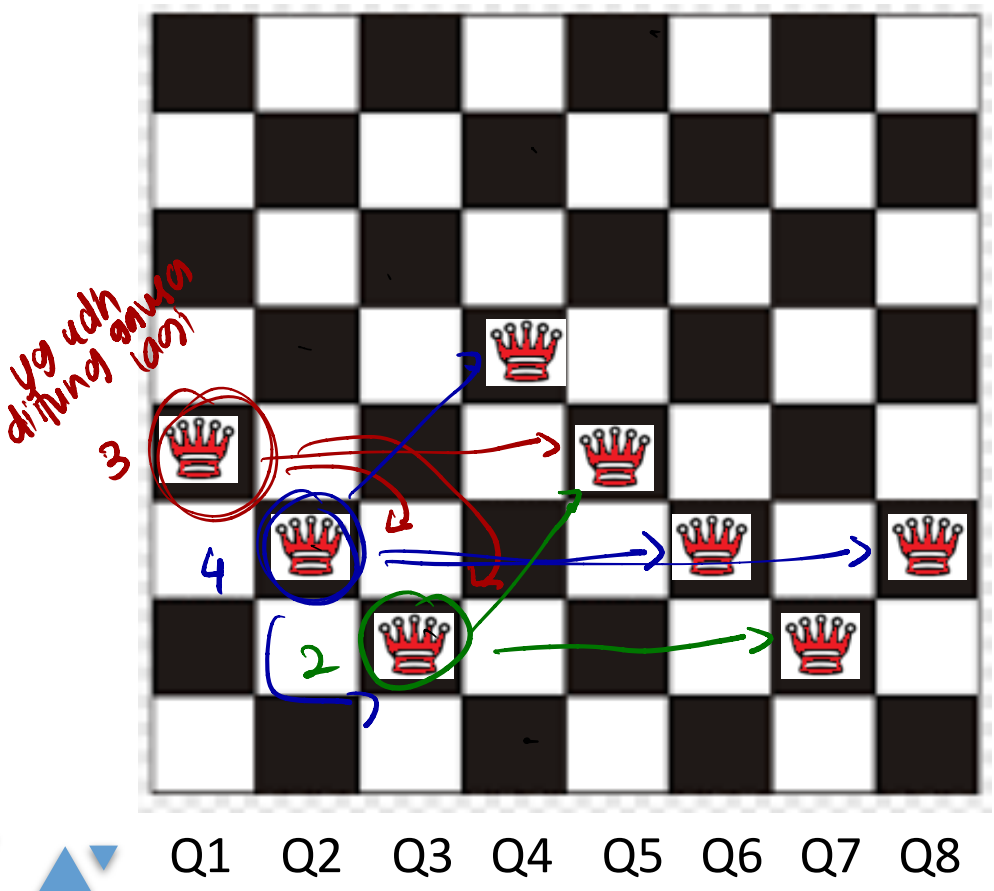
State Value (h) for 8-queens problem

- *h* = - number of pairs of queens that are attacking each other, either directly or indirectly.
- h=-1: only 1 pair of queens (Q4 attacks Q7)
- Optimum solution has global maximum, h=0.



h = *-1* for the above state

Exercise: Determine h



List pairs of queens that are attacking each other, either directly or indirectly.

h = - number of pairs of queens that are attacking each other, either directly or indirectly.

ada 56 tetangga yg mungkin
(1 ratu bs punya 7 kondisi lain)

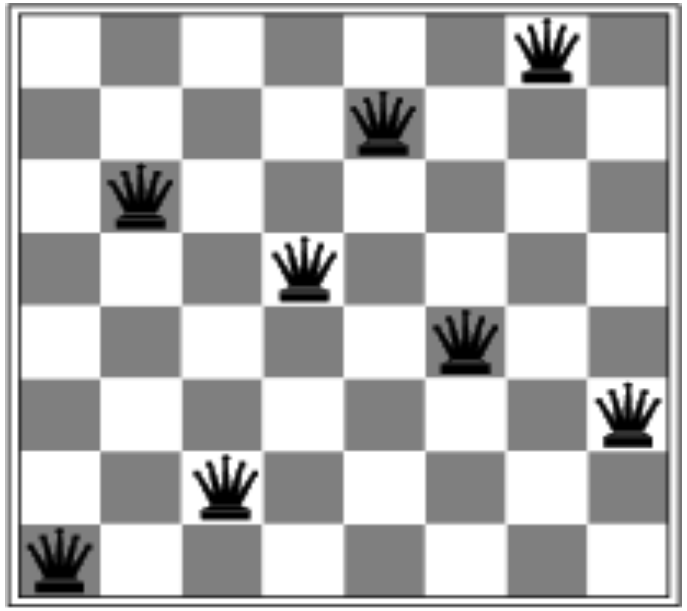
Local Search: Action

Keep single current state
State: "complete" configurations (one queen per column)
Solution: final state

Find the best state (global optimum)
Action: move to neighbor

What is neighbor of a state ?

Neighbor and Successors for 8-queens



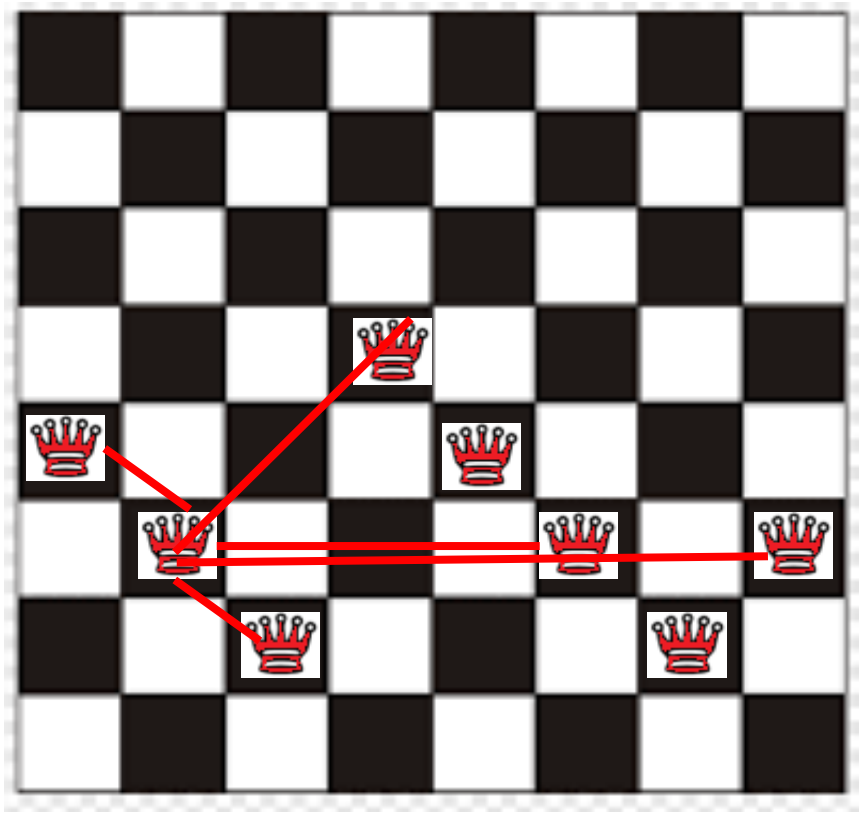
Neighbor: highest-valued successor

- Successor function returns all possible states generated by moving a single queen to another square in the same column.
- Each state has 8*7 = 56 successors
- Choose randomly among the set of best successors if there is more than one.

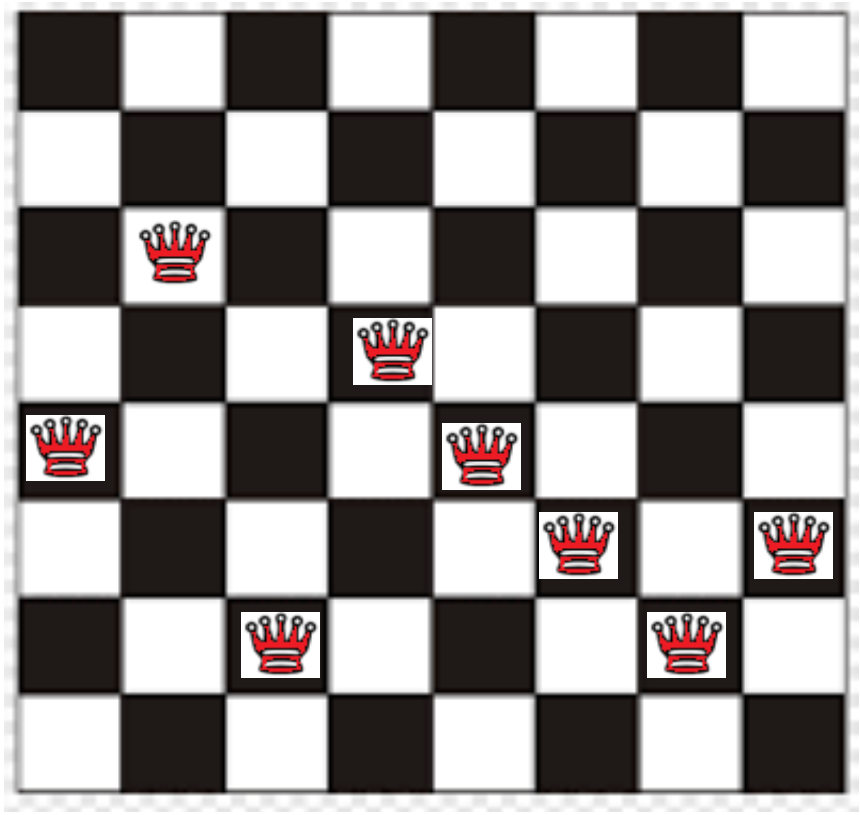
Neighbor: random successor

- Successor function returns a random state generated by moving a random single queen to another square in the same column.

Neighbor: Random Successor



Column: Q2
Row: 3➔6



Neighbor: Highest-valued Successor

Each number indicates h if we move a queen in its corresponding column

State value dari setiap successor belum diberi tanda negative (cost)

Summary: State, Neighbor

Keep single current state
State: "complete" configurations (one queen per column)
Solution: final state

Find the best state (global optimum)
Action: move to neighbors

Next:

- Hill climbing Search
- Simulated annealing
- Genetic algorithm