04/02/2022

Local Search

Path V/S State optimization.

- Searching in state space
- searching in solution space.

Satisfaction V/S Optimization.

- Satisfaction → Goal satisfaction
  - Reach the goal node.
  - Constraint satisfaction.
- Optimization. → Optimize. (function cost)
  - Constraint optimization.

Goal state:

Constraints:

Convert

Minimize objective func^n : # constraints not satisfied.

Constraints:- $C_1\ C_2 \dots C_n$

$\downarrow$

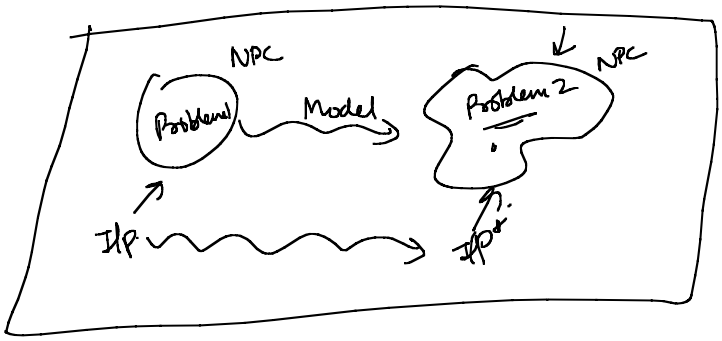
$f(n)$ reaches 0

Max $f(n)$ st C_1 C_2 C_n.

$\leftarrow$ Unt: $f(n) \geq C^*$, $C_1, C_2 \dots C_n$.

8-Queens problem

- 8 * 8 chessboard
- Place 8 queens
- such that no two queens can attack each other.



$f = 17$.

Constraint:- Exactly one queen in each column.

Objective func^n:- (Minimize) Number of pairs of queens that are attacking each other. $\Rightarrow f$.

Local Search

- keep track of single current state.
- move only to neighboring state
- Ignore paths.

Adv:- = use very little memory.

= can often find reasonable soln in even large or infinite state space.

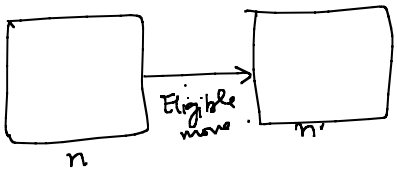
Search Space

State:- All 8 queens on the board in some configuration.

$$\#\text{of possible states} = 8^8$$

Min. $f(n)$: # pairs of queens in attacking position.

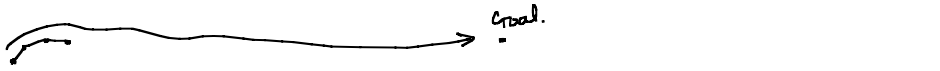
Neighborhood / Succesor func^n :- Move a single queen to any other position in the same column.

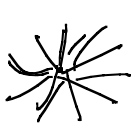


neighbors = 56 (7×8)

Move all queens one position in same column.

#neighbors 8





balance ← A good neighbor func^n.

Hill Climbing