

지역 탐색

# 지역 탐색

- 딥러닝 모델 학습 과정에 활용
- 체스 게임을 탐색 문제로 정의하면 목적은 상대방의 왕을 잡는 것(path cost보다는 goal에 도달하는 것이 더 중요)
- 현재 어떤 state에 있다고 할 때 내가 어떤 액션을 취해서 바로 갈 수 있는 인접한 state들 중에 지금 내 state보다 낫다고 판단되는 데가 있으면 그것을 그냥 취한다.
- 그냥 매 state에서 '지금 보다 더 나은 state로만 가보자.'

# N-queens

- Queen이  $n$ 개가 있다고 가정,  $n \times n$  보드 존재
- Queen과 Queen이 서로 안 싸우게 되는 경우를 만든다.

# Hill-Climbing Search

- 에베레스트 산에 올라가는 중 (매우 깊은 안개 속)
- 발 앞뒤로 1m만 볼 수 있는 상태
- 기억상실증이 걸림(전에 왔던 state를 기억 못함)
- Initial point를 어디로 잡는지에 따라 결과가 매우 달라짐
- Steepest ascent : 내 neighbor들 중에서 가장 Optimization한 값들만 찾겠다.
- Greedy local search(Stochastic Hill Climbing) : goal에 어떻게 갈지, 이후에 무슨 일이 일어날지 고려하지 않고 단순히 좋은 이웃으로만 가겠다.
- 가장 높은 지점(global maximum)인 목표보다 낮은 어떤 언덕의 꼭대기(local maximum)에 도달하게 될 수 있다.

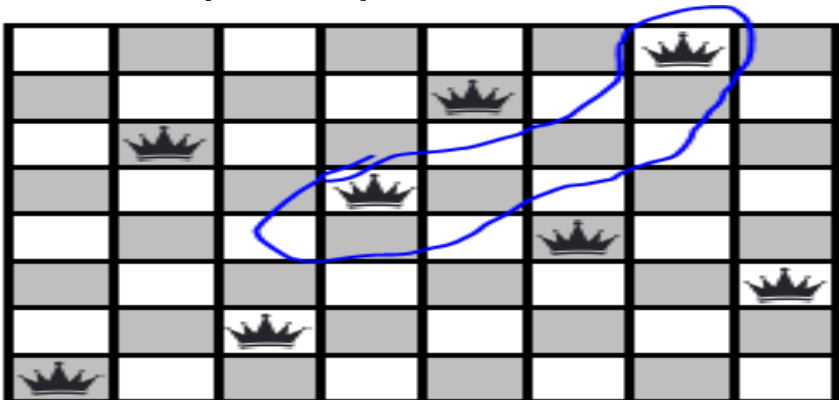
# 8-queens Problem

- 8x8 판에 8개의 queen이 존재. Queen들이 서로 공격하지 않는 배열 찾아야함
- $H$  = 서로 공격하는 여왕의 쌍
- 즉,  $H=0$ 이 되도록 만들어야함.
- $H=17$

18	12	14	13	13	12	14	14
14	16	13	15	12	14	12	16
14	12	18	13	15	12	14	14
15	14	14	♔	13	16	13	16
♔	14	17	15	♔	14	16	16
17	♔	16	18	15	♔	15	♔
18	14	♔	15	15	14	♔	16
14	14	13	17	12	14	12	18

# Hill-Climbing Search

- $H=17$ 에서 조금이라도 줄어드는 방향이 있다면 그 노드로 이동한다.
- 장점 : 단순하고 계산의 부하가 적다.
- 단점 : 최적의 솔루션에 도달하지 못하는 경우도 있다.
- 결과 : Global minimum을 찾아야 하나, local minimum에 빠져 버림( $H=1$ )



# Simulated Annealing Search

- Hill-Climbing Algorithms는 결코 downhill을 하지 않는다.
- Local maximum에서 빠져나갈 수 없다.
- Hill-Climbing + random walk(가끔 best move를 선택하는 대신에 random move를 선택하겠다.)
- 만약 그 move가 현재 상황보다 낫다면 그 state를 받아들이고, 안 좋다고 하면 randomness를 통해 받아들이든지, 안 받아들이든지 결정한다.
- 봉우리를 만났을 때 다시 갈림길까지 내려가서 다른 봉우리로 이동한다. '온도'와 더 안 좋아지는 정도를 가지고 전이할 지 말지 결정한다.
- 온도가 충분히 느리게 낮아진다면 Optimal하다. 하지만 실제로 이용하기에는 너무나 속도가 느리다.
- 최적해를 반드시 구할 때 보다는 어느 정도 수준이 보장되는 해답을 구해야 할 때 주로 사용된다.