

# Chapter 1. Introduction

강준하

# Reinforcement Learning?

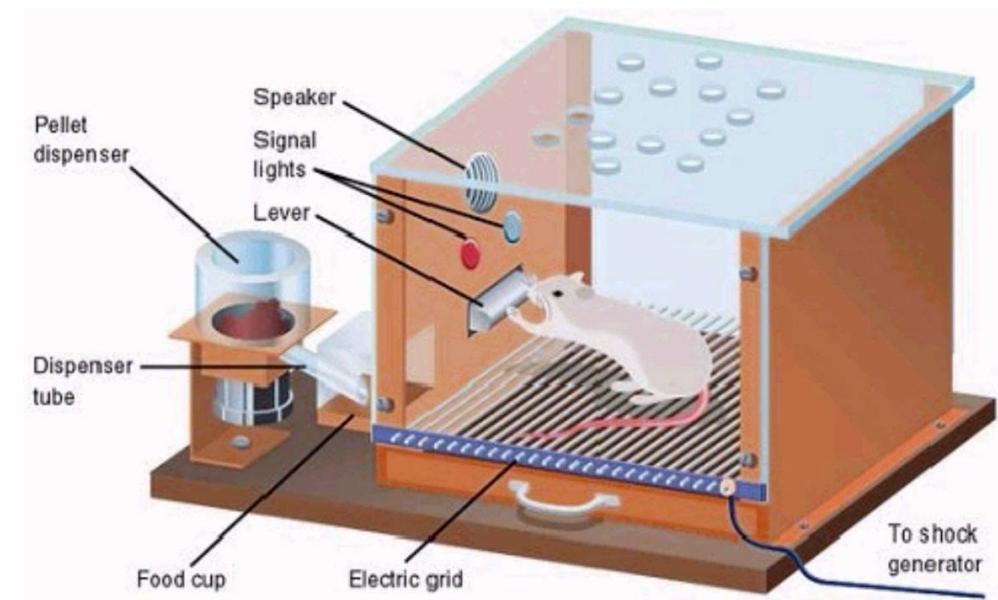
# Reinforcement



# Learning

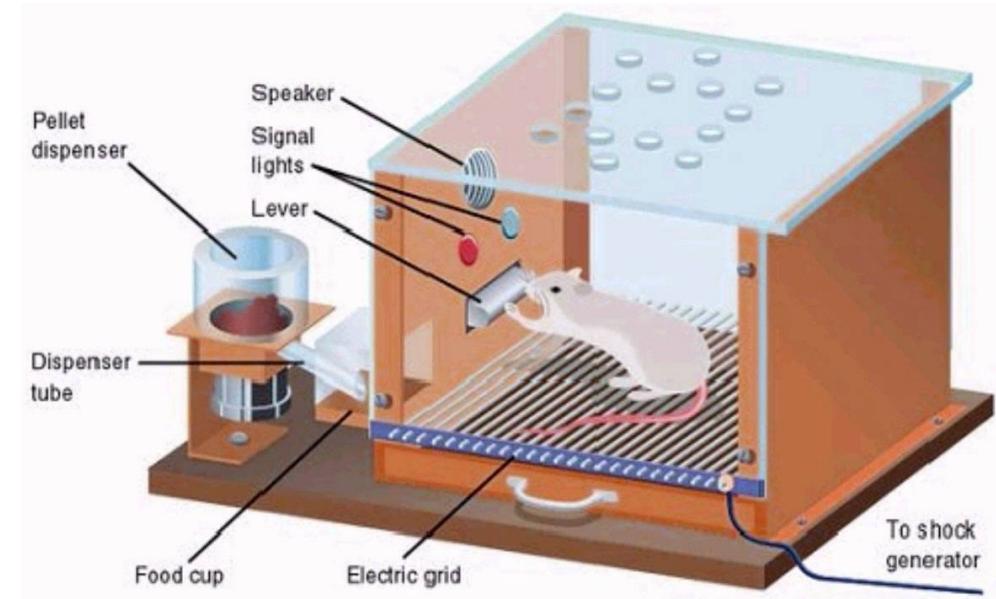
- a problem
- a class of solution methods that work well on the problem
- the field that studies this problem and its solution methods

# Reinforcement Learning



# The Skinner Box

1. 배고픈 상태의 흰 쥐를 스ки너 상자에 넣는다(이렇게 배고픈 상태로 만드는 것을 박탈이라고 한다).
2. 쥐는 스ки너 상자 안에서 돌아다니다가 우연히 지렛대를 누르게 된다.
3. 지렛대를 누르니 먹이가 나온다.
4. 지렛대와 먹이 간의 상관관계를 알지 못하는 쥐는 다시 상자 안을 돌아다닌다.
5. 다시 우연히 지렛대를 누른 흰 쥐는 또 먹이가 나오는 것을 보고 지렛대를 누르는 행동을 자주하게 된다.
6. 이 과정이 반복되면서 흰 쥐는 지렛대를 누르면 먹이가 나오다는 사실을 학습하게 된다.

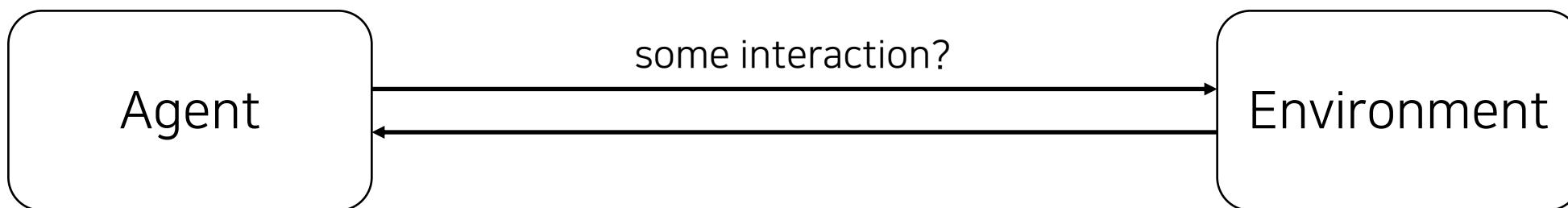


# Pigeon Ping Pong

<https://www.youtube.com/watch?v=vGazyH6fQQ4>

# Reinforcement Learning

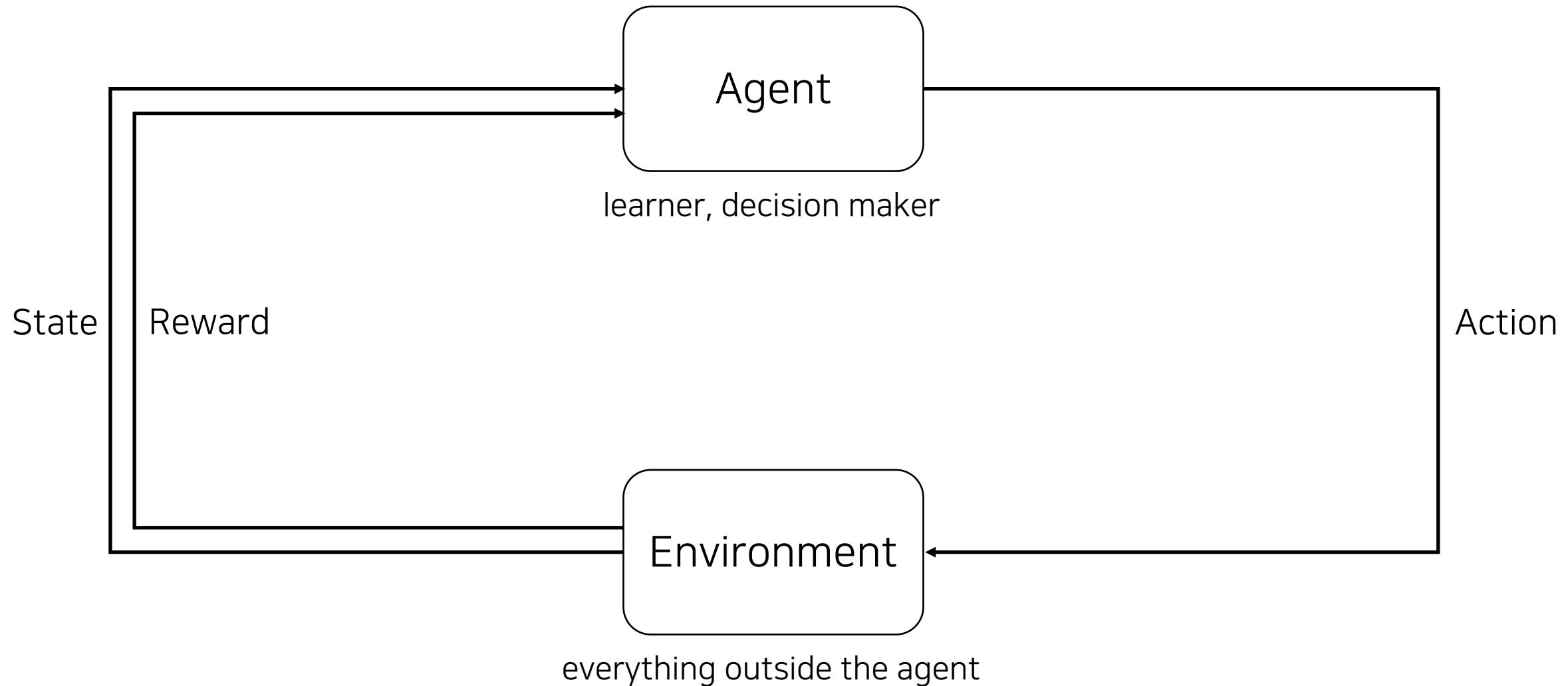
- 우리의 삶은 환경과의 상호 작용으로 이루어짐
- 이러한 상호 작용으로부터 배우는 행위 → RL
- 이에 대한 computational approach



# Reinforcement Learning

- Learning what to do
- 어떻게 상황과 행동을 mapping할 것인가
- 문제 - 어떤 행동이 좋은 행동인지 모른다!
- 직접 행동해보고 보상을 받는 과정에서 배워야 함
- trial-and-error search / delayed reward

# Reinforcement Learning



# Reinforcement Learning & Machine Learning?

# Machine Learning

## Supervised Learning

- Find correct thing from external supervised knowledge

## Unsupervised Learning

- Find hidden structure from arbitrary data set
- Does not rely on correctness

## Reinforcement Learning

- Find state-action map from reward signal



# Reinforcement Learning is Difficult!

- trade-off between exploration and exploitation
- 지속된 exploration : 성능을 높일 수 없음
- 지속된 exploitation : 보다 효과적인 행동을 모르고 지나갈 수도 있음

# How to solve RL problem?

- 확실한 목표와 순차적으로 결정을 내려야 하는 문제
- Markov Decision Process(MDP)로 환원
  - sensation
  - action
  - goal

# MDP

- 상태(state) : 정적인 요소 + 동적인 요소(ex. 속도, 가속도 등)
- 행동(action) : 어떠한 상태에서 취할 수 있는 행동(ex. 상, 하, 좌, 우)
- 보상(reward) : 에이전트가 학습할 수 있는 유일한 정보
- 정책(policy) : 순차적 행동 결정 문제에서 구해야 할 정답, 모든 상태에 대해 에이전트가 어떤 행동을 해야 하는지 정해놓은 것

순차적 행동 결정 문제를 풀었다? -> 'Optimal Policy'를 찾았다!

# MDP solving

- 우선 어떤 state에서 action을 취했을 때 reward를 얼마나 받을 수 있을지 예측하고자 함
- 이를 value (function) 이라고 부름
- value function을 보다 정확하게 추정하는 방법을 공부할 예정

# RL could do more things!

- Discrete time step이 아닌, continuous time step에도 적용될 수 있다.
- Large state set 또는 infinite state set에도 사용 가능하다. Gerry Tesauro는 약 1020가지의 state에 달하는 backgammon이라는 게임을 강화학습으로 학습시키기 위해 artificial neural network를 사용했고, 인간 제일의 선수에게서 승리하는데 성공했다.
- efficient learning을 위해 사전지식을 활용할 수도 있다.
- 미리 가능한 수를 내다보고, 가늠하기 위해 model을 이용할 수 있다. Model이 있는 방식을 model-based methods라 하고, model이 없는 방식을 model-free methods라고 한다.

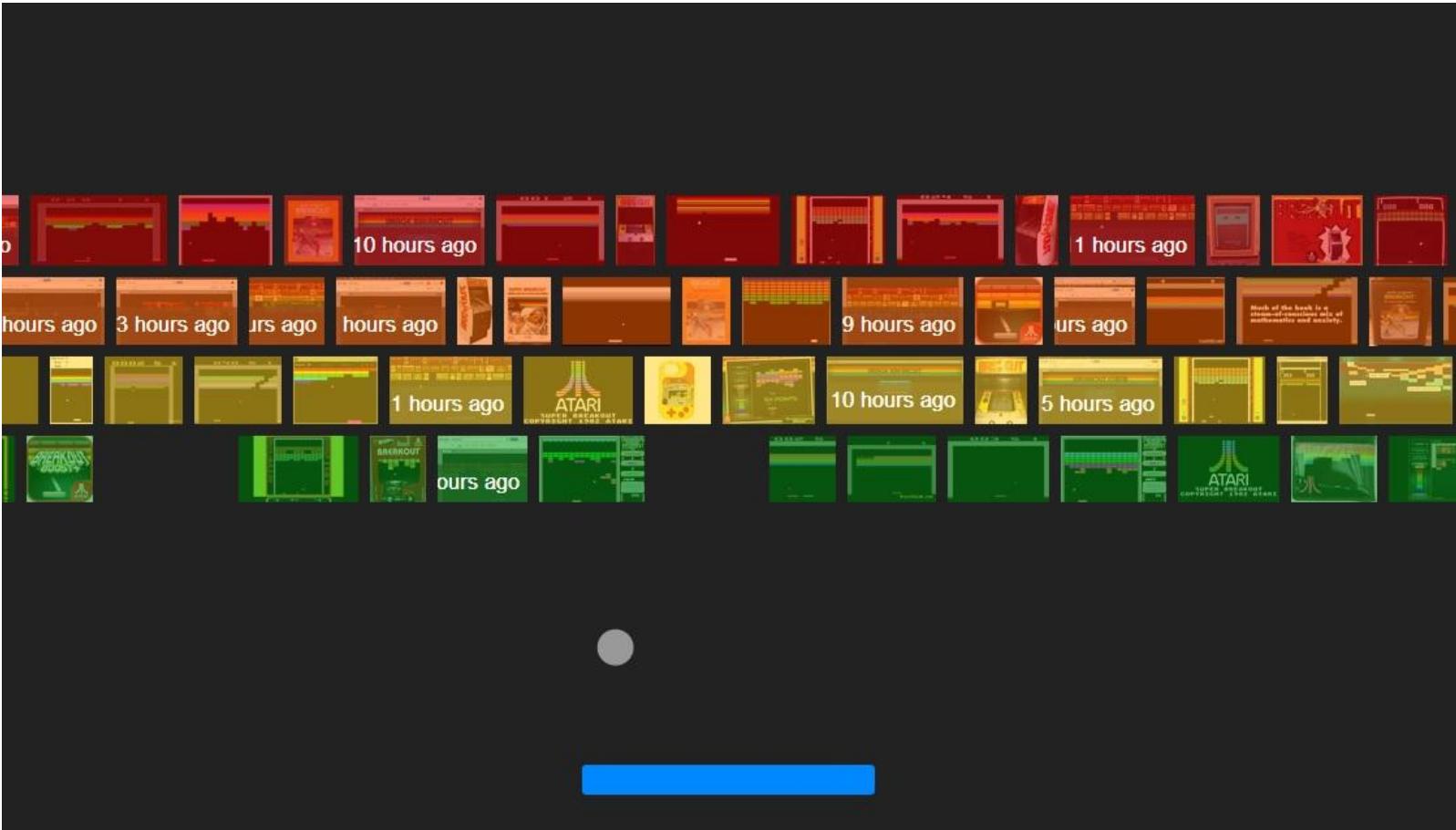
# Limitations and Scope 1

- State는 policy와 value function에 대한 입력으로 사용되고, model에 대해서는 입력과 출력으로 사용됨
- RL은 state에 매우 의존적임
- 본 책에서는 state signal은 그냥 환경에서 주어진다고 가정하고, state signal이 어떻게 생성되고 변하는지에 대해서는 다루지 않음  
(decision-making에 모든 초점을 맞추기 위해서)

# Limitations and Scope 2

- 대부분의 RL 방법론은 value function을 추정하는 방법에 주안점을 두고 있음
- evolutionary methods의 경우, value function을 추정하지 않더라도 문제 해결 가능하기는 함
  - policy 공간이 충분히 작고 구조화가 잘 되어있을 경우에는 잘 동작함
  - 하지만 RL problem의 유용한 구조들을 잘 활용하지 못함
  - state와 action 간의 관계를 이용하지 못함

# Case study 1 : Atari Breakout



<https://www.youtube.com/watch?v=V1eYniJ0Rnk>

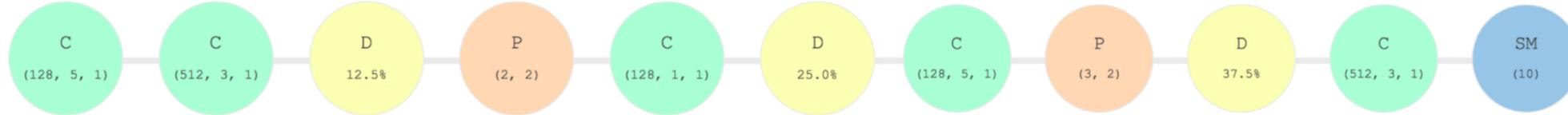
# Case study 2 : AlphaGo



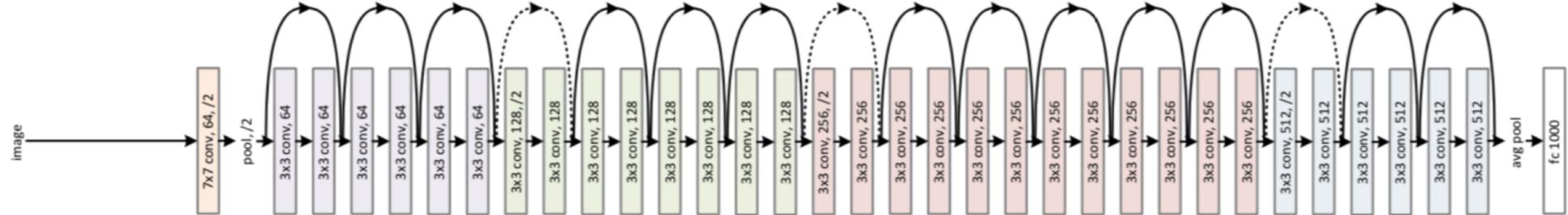
# Case study 3 : Dota 2



# Case study 4 : MetaQNN



VS.



# Case study 5 : and more...