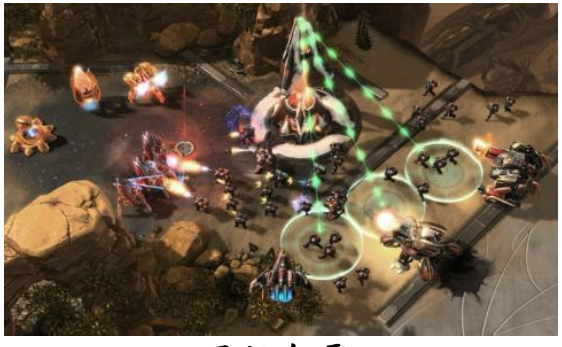
Algorithmic and Theoretical Foundations of RL

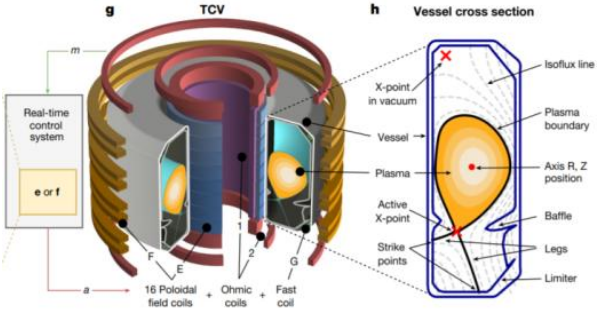
Introduction

Success of RL

围棋
(Nature, 2016)



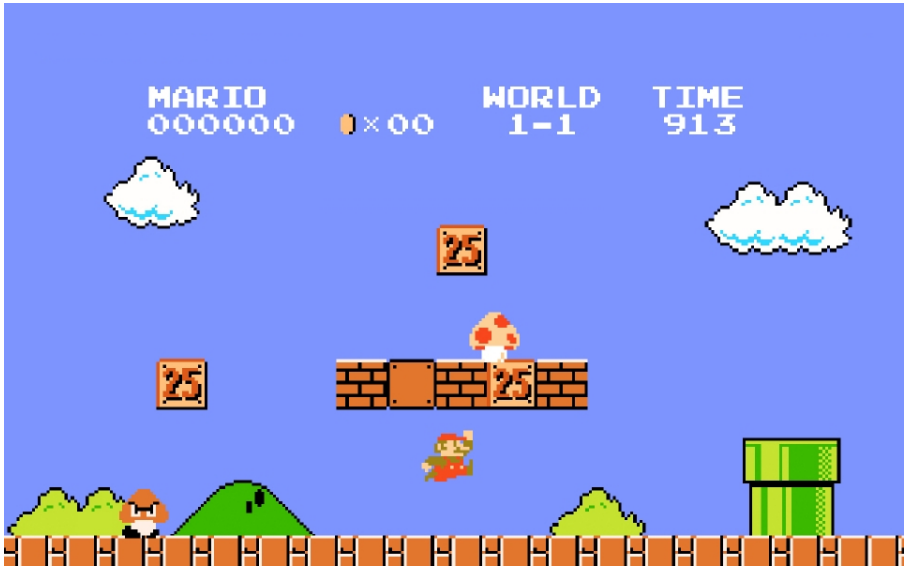
星际争霸
(Nature, 2019)



核聚变
(Nature, 2022)

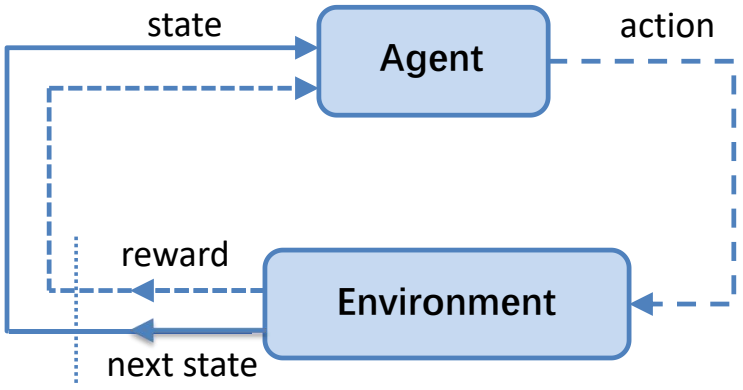
- RL has also been used to solve science problems like traveling salesman problem and plays an important role in “AI for Science”.

Example: Super Mario



Super Mario makes a decision, then receives a reward and transfers to the next state; Goal: high cumulative reward by making right decisions.

RL Model

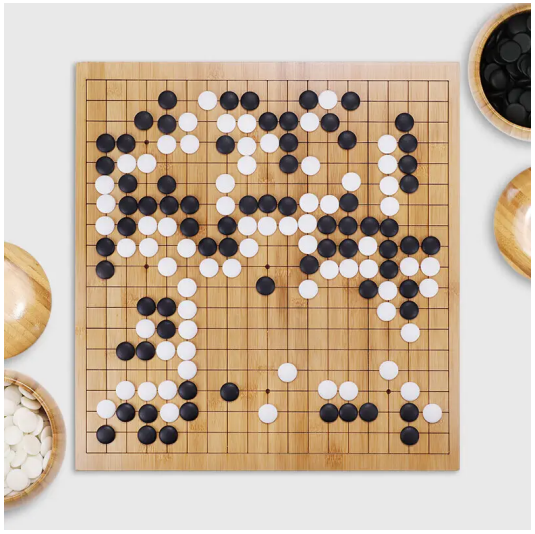


$$\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, P, r, \gamma \rangle$$

Reinforcement learning is learning what to do –how to map situations to actions –so as to maximize a numerical reward signal in an unknown uncertain environment.

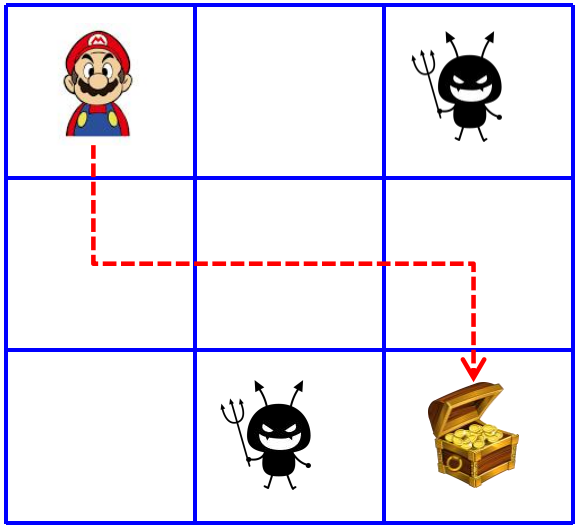
–Sutton and Barto (1998).

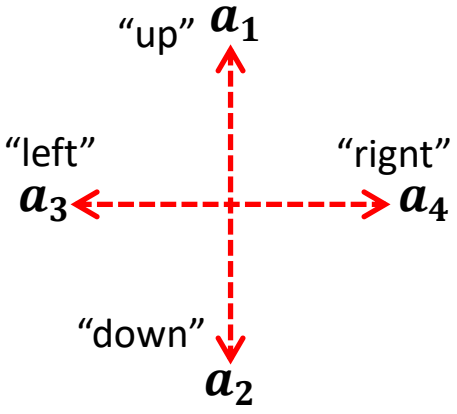
Challenges in RL



- Ultra-high dimensional problem
- Non-convexity
- Uncertainty
- Changing data distribution up to policy
- Trial-and-error
- . . . . . .

Simple Environment: GridWorld

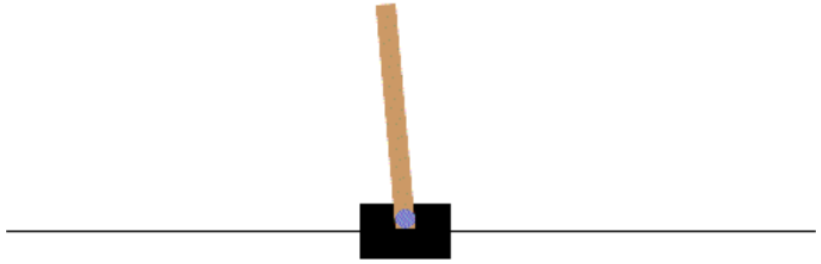




s_1	s_2	s_3 -5
s_4	s_5	s_6
s_7	s_8 -5	s_9 10

- State space: $\mathcal{S} = \{s_i\}_{i=1}^{9}$
- Action space: $\mathcal{A} = \{a_i\}_{i=1}^{4}$
- Reward: $r = -5$ if hitting "obstacle" grid; $r = 10$ if arriving at "goal" grid
- Goal: Arriving "goal" grid while avoiding "obstacle" grid

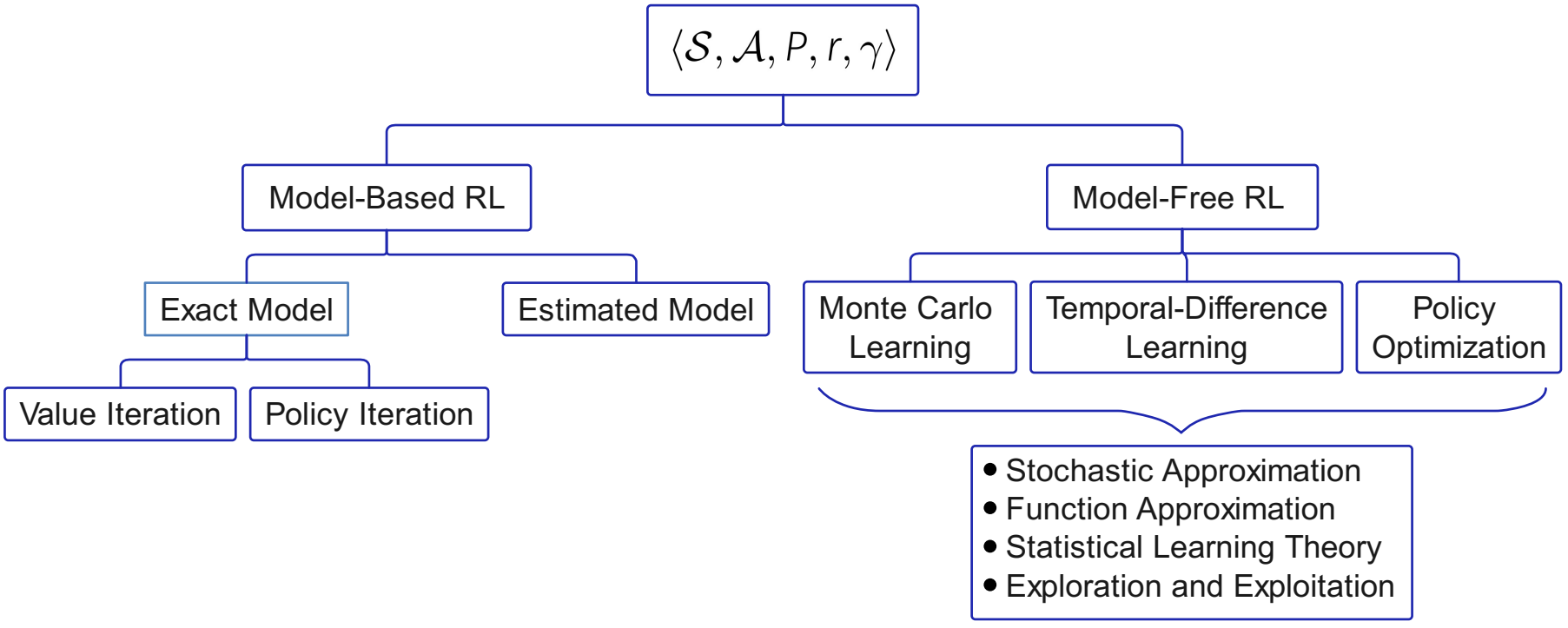
Simple Environment: CartPole



- State: $s = [x, y, \theta, \omega] \in \mathbb{R}^4$
  - $x \in [-4.8, 4.8]$: cart position
  - $y \in \mathbb{R}$: cart velocity
  - $\theta \in [-24°, 24°]$: pole angle
  - $\omega \in \mathbb{R}$: pole velocity at tip
- Action space: $\mathcal{A} = \{\text{left, right}\}$
- Reward: $r = 1$ if the pole remains upright, $r = 0$ otherwise
- Goal: prevent pole from falling over

More typical experiment settings, such as Mountain Car and Cliff Walking, can be found in OpenAI Gym (https://github.com/openai/gym).

Outline



- Emphasize on basic methods and theory, but advanced methods will also be briefly discussed.
- Related topics such as LQR, partially observable setting, Batch RL will be selectively discussed if time permitted.

Logistics

- **Prerequisites:** Probability and statistics, numerical optimization
- **Grading policy:** 60% Homework + 40% Final
- **Homework:**
  - Homework will be assigned via eLearning, and time for homework is typically one week;
  - Coding language for this course is Python.
- **Course policies:**
  - Final exam is closed book.
  - Regrading request must be submitted within five days of the return day.
  - Cheating in assignments and exams is not tolerated! Any sort of suspected cheating will result in zero grade of the corresponding assignments or exams, followed by penalty subject to university rules.

Questions?