Foundations of Reinforcement Learning

Introduction

Yuejie Chi

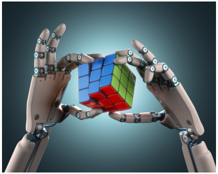
Department of Electrical and Computer Engineering

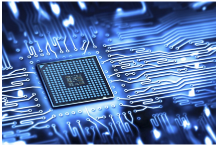
Carnegie Mellon University

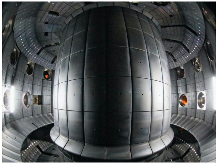
Spring 2023

Outline

Recent successes in reinforcement learning (RL)







RL holds great promise in the next era of artificial intelligence.

Supervised learning

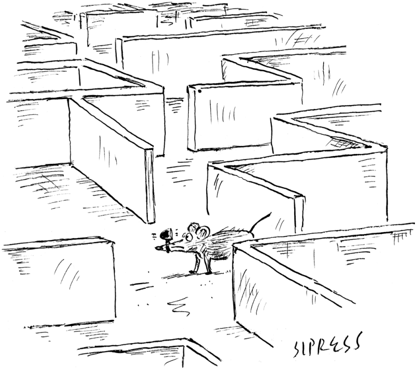
Given training data, make prediction on unseen data:

Primarily deal with **pattern recognition**

Reinforcement learning

In RL, an agent learns by interacting with an environment.

- no training data
- maximize total rewards
- trial-and-error
- sequential and online



"Recalculating ... recalculating ..."

Deal with **decision making**, sometimes with constraints

Sequential decision making

"Those who cannot remember the past are condemned to repeat it."

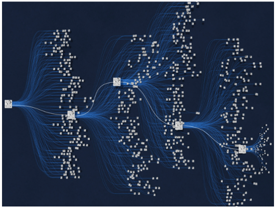
—George Santayana

- Games
- Robotics navigation and control
- Pricing and supply chain management
- Recommendation systems
- Portfolio optimization

Learn from past to predict and optimize future performance

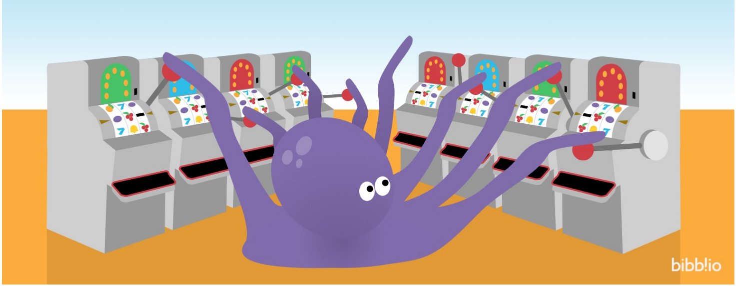
Challenges of RL

- explore or exploit: unknown or changing environments
- credit assignment problem: delayed rewards or feedback
- enormous state and action space
- nonconvex optimization



Multi-arm bandit

Which slot machine will give me the most money?



Learning the best arm

Can we learn which slot machine gives the most money?

$1
$0
$0

$1
$4
$0
$2
$1
$3
$5

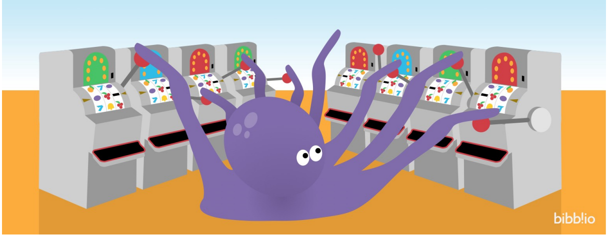
$1
$0
$1
$2

Learning the best arm via trial-and-error

Which arm do I pick next, so that I maximize my reward over time?

$1	$1	$1
$0	$4	$0
$0	$0	$1
	$2	$2
	$1	$12
	$3	$11
	$5	

Exploration-exploitation trade-off



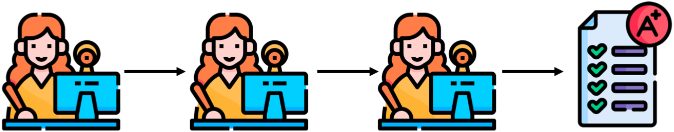
Which arm should I play?

- Best arm observed so far? (exploitation)
- Or should I look around to try and find a better arm? (exploration)

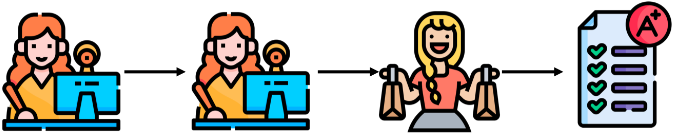
We need both in order to maximize the total reward.

Credit assignment problem

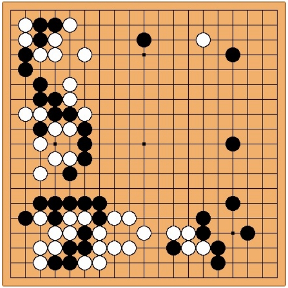
What is the action that leads to the desired outcome?



What if....



Enormous problem size and function approximation



$$S \approx 2 \cdot 10^{170}$$

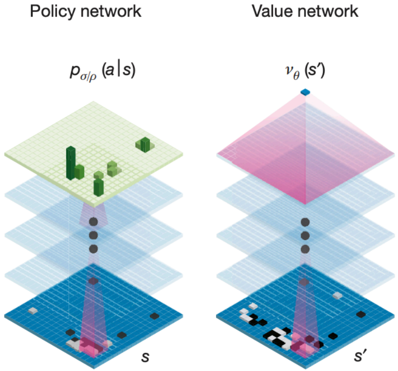
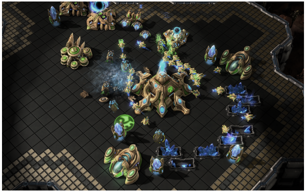


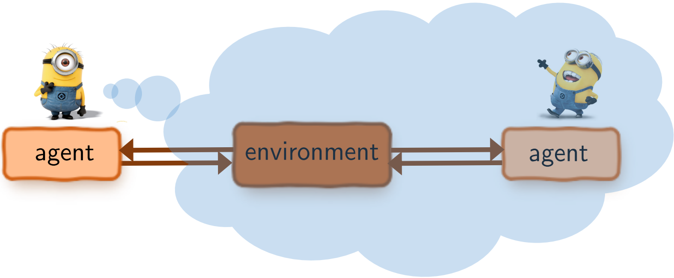
Figure credit: Alphago

Multi-agent RL



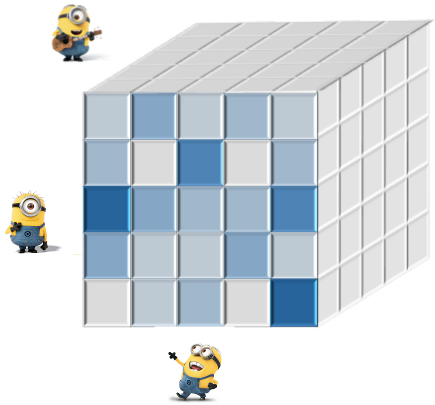
To collaborate or to compete, that is the question.

Challenges in MARL: nonstationarity



From a single-agent perspective:
the environment is **time-varying** and **nonstationary**!

Challenges in MARL: curse of multiple agents

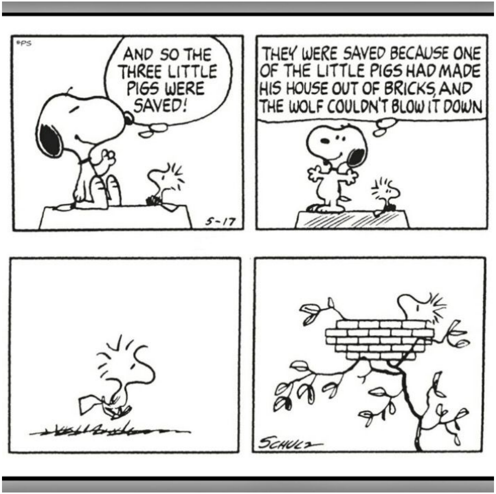


The explosion of choices:
The joint action space grows **exponentially** with the agents!

Partial observability in RL

Goal of this course

- **Not** a deep RL course
- Aim to build the "foundations"
- 800-level course: research-oriented
- models, algorithms and their analyses



Sample efficiency

Collecting data samples might be expensive or time-consuming

clinical trials

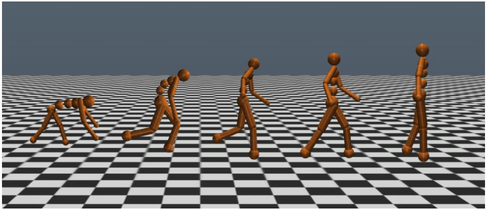
autonomous driving

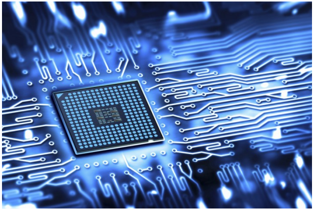
online ads

Calls for design of sample-efficient RL algorithms!

Computational efficiency

Running RL algorithms might take a long time and space





many CPUs / GPUs / TPUs + computing hours

Calls for computationally efficient RL algorithms!

From asymptotic to non-asymptotic analyses

Non-asymptotic analyses are key to understand sample and computational efficiency in modern RL.

Logistics

Basic information

- Tue/Thu: 3:30 – 4:50 pm
- Instructor's office hours: Wed 1 – 2pm, PH B25
- TA's office hours: Jiin Woo, Thu 1 – 2pm, CIC 4117 Bellefield
- Course website:
  `https://users.ece.cmu.edu/~yuejiec/ece18813B.html`
- Piazza and gradescope.

Why you should consider taking this course

- There will be quite a few THEOREMS and PROOFS ...
  - Promote deeper understanding of scientific/engineering results
- Nonrigorous / heuristic from time to time
  - "Nonrigorous" but grounded in rigorous theory
  - Help develop intuition
- No exams!

Tentative topics

- Multi-arm bandit
- Markov decision processes
- RL with a generative model
- Online RL
- Offline RL
- Policy optimization
- Actor critic
- Function approximation and representation learning
- Multi-agent RL
- Partially-observed MDP

Useful references

We recommend these books, but will not follow them closely ...

- Reinforcement Learning: Theory and Algorithms (draft), by Alekh Agarwal, Nan Jiang, Sham M. Kakade, Wen Sun
- Reinforcement learning: An introduction, by Richard S. Sutton, Andrew G. Barto
- Reinforcement learning and optimal control, by Dimitri P. Bertsekas
- Bandit Algorithms, by Tor Lattimore, Csaba Szepesvari

More references will be provided at each lecture.

Prerequisites

- linear algebra
- probability
- a programming language (e.g. Matlab, Python, ...)
- basic optimization

- *Concentration inequalities* are a plus, but not necessary

Grading

- Homeworks (20%): $\sim$2 problem sets
  - Use gradescope for submission and grading.

- Midterm Paper Presentations (25%)
  - An in-class presentation on a selected paper from a given pool is arranged in lieu of the midterm.
  - About 15-20 min each, highlight at least one key result

- Final project (55%)

Final project

Two forms

- literature review on a research **topic** (individual)
- original research (can be individual or a group of two)
  - *You are strongly encouraged to combine it with your own research*

Three milestones

- Proposal (March 23): up to 2 pages (NeurIPS format). Plan early! Use midterm paper as a planner.
- In-class presentation (last week of class)
- Report (May 14): up to 5 pages with unlimited appendix

Enjoy Yourself