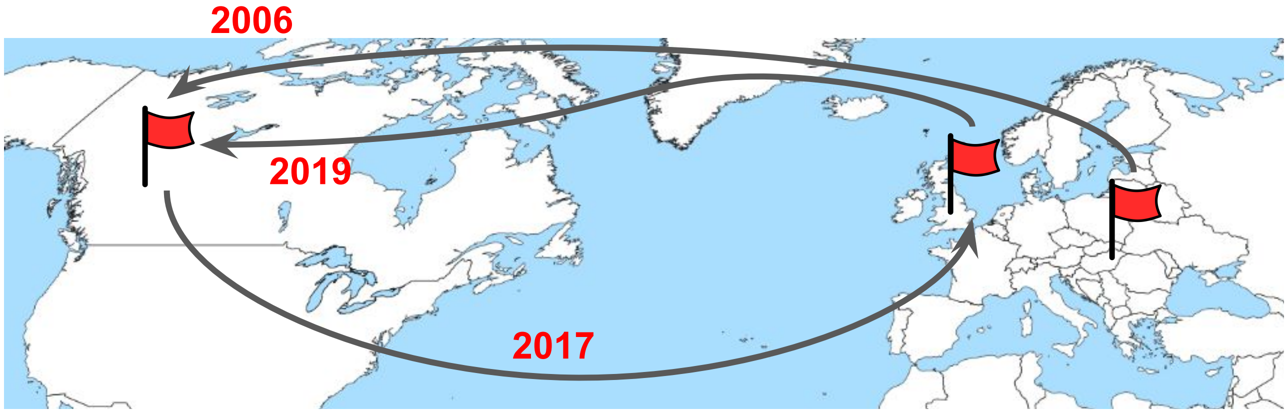
CMPUT 653 W2022

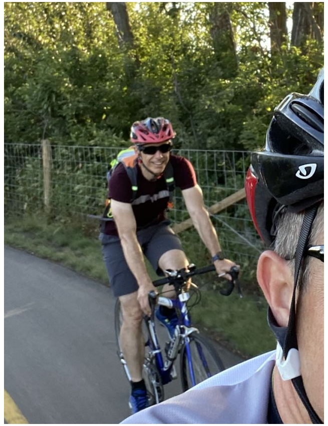
Introductions

Contents

- Administrivia
  - Introductions: Csaba/Course
  - Course structure: delivery, grading
  - Expectation management
- Intro to RL
  - What is RL?
  - The MDP framework
  - Pesky probabilities

Csaba ⇒ Chaba(?)





Work

PhD’99, RL	1999
Mindmaker	1997-2002
MTA SZTAKI	2003-2011
UofA	2006-
DeepMind	2017-

Research

- Control book, RL book
  Bandit book
- MCTS, RL+Generalization,
  Exploration (PM, linear bandits)

Course: Theoretical Foundations of RL

- Website: RL Theory
- Eclass: https://eclass.srv.ualberta.ca/course/view.php?id=76687
  - For submissions and marking only
- Slack: AMII workplace
  - cmput653-discussion-w2022, cmput653-private-discussion-w2022
- Classes
  - MW 2:00pm-3:20pm, GSB 5-53
  - Until Jan 25: Virtual, flipped class
  - After Jan 25: In-person
- Work you will do at home
  - Reading, watching lectures, preparing questions, voting on questions
  - Assignments, midterm, group-project. Deadlines posted on website
  - Working with others: Project: √√ assignment discussions: √ , midterm: -
  - Late submissions, contesting marks: See website

Alex: aayoub@ualberta.ca

Vlad: vtkachuk@ualberta.ca

Why theory and what is “theory”?

- Theory = Math (not theorizing!)
- True/false: Crisp, truth values are constant in time
- Questions:
  - Algorithms, efficiency, effectiveness
  - Do they exist?
  - When?
  - How efficient? How effective?
- Math: A way of learning about reality (reality of algorithms)
- Abstract! Simplified!
- May miss detail
- Art: Choose level of detail. “Modeling”

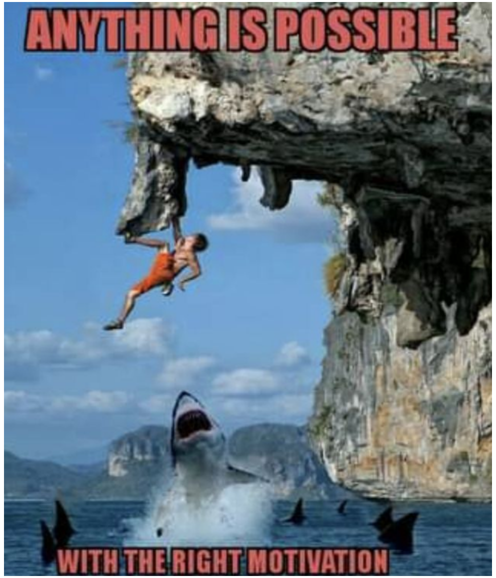
Expectation management

I expect that you ...

- .. are here to learn, want to learn ..
- .. take charge of your learning ..
- .. participate in class, ask questions ..
- .. respect your peers' learning needs

You can expect me to …

- .. respect you
- .. help you to learn and grow
- .. try to understand (and answer) your questions
- .. teach you about the state-of-the-art in RL



Course contents

RL = Reinforcement learning

Why this formulation?
What other formulations could we use?
Why is learning important? For AI!

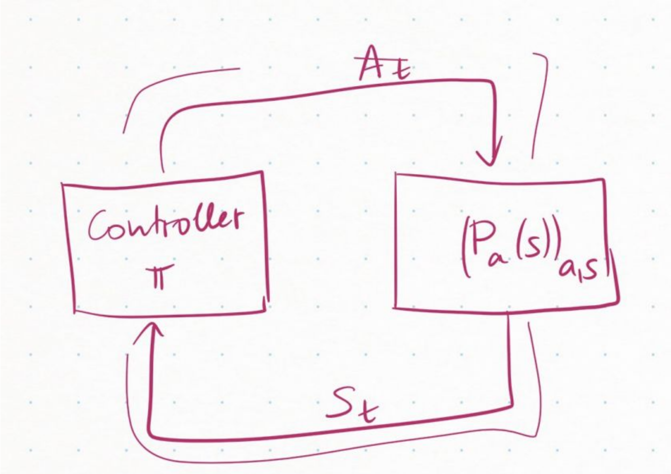
RL is about..

- Problems
- Body of knowledge
- Techniques/methods

The general RL problem formulation

- Take actions in a stochastic environment to maximize total reward while taking observations about the environment’s state
- Learning
  - Algorithm needs to work across multiple environments
  - It is NOT given the environment
  - It needs to use observations to decide what action to take

The MDP framework



Markov Decision Process =
Controlled Markov Process + Markov rewards

Controller = policy = algorithm

- Can use state: Feedback!
- May be restricted use something less

General controller/policy

- Can use all past observations
- “History dependent”
- Do we need these?

From policies to value functions

Trajectories: $(s_0, a_0, s_1, a_1, ..)$

Where are the rewards?

Policy *π* + MDP + initial state *s* ⇒ distribution over trajectories P_s^π

Can take expectation of a function that assigns a number to each trajectory
w.r.t. the distribution P_s^π ⇒ V^π(s)

How many states?

How many actions?

Breakout room practice [~20 mins]

Do in parallel:

1. Introductions [5 mins]
2. Formulate and discuss a question [5 mins]
    a. Why this way?
    b. What else?
    c. Limitations?

Do serially:

3. Rejoin main session, summarize question/discussion nrooms Nrooms * [2 mins]