ECE7121 Learning-based control – 2025 Fall

Introduction

INHA UNIVERSITY

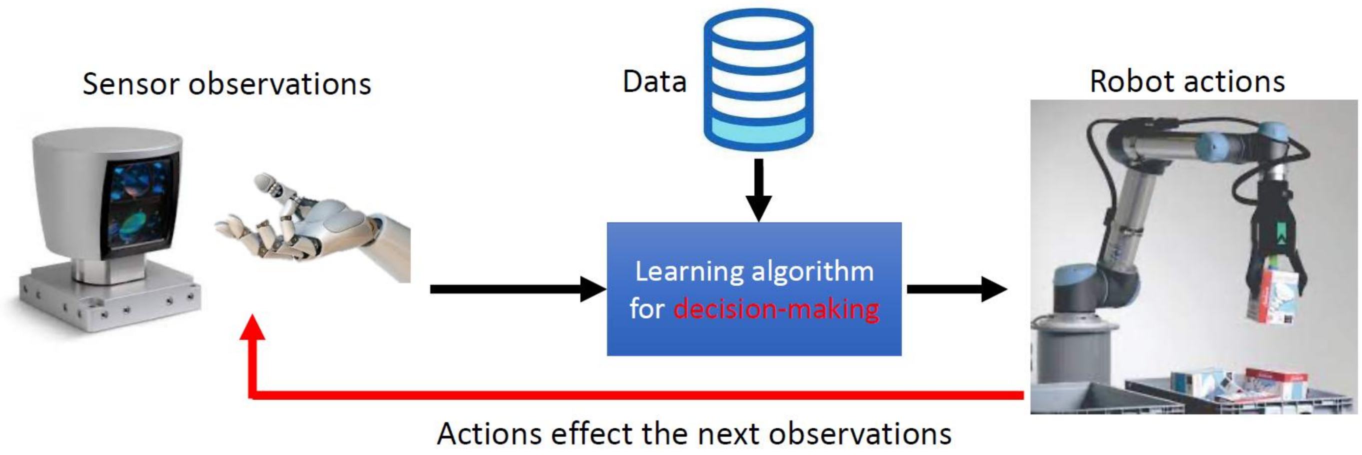
Overview

- Goal of the course / why it is important
- What is Reinforcement Learning (RL)? Why study RL?
- Where are we today? How far are we from the goal?

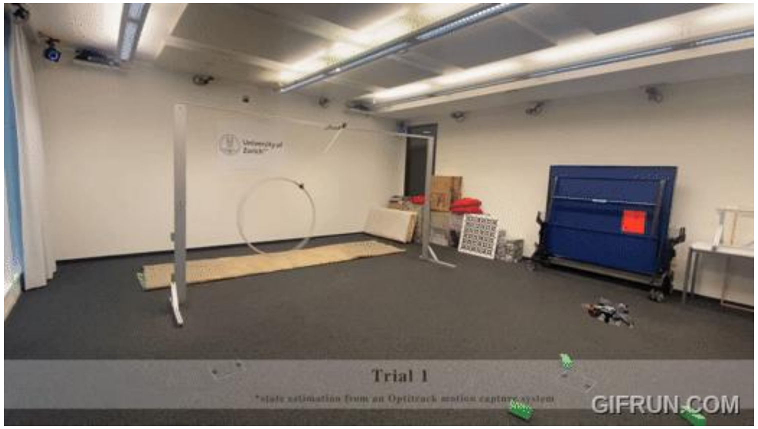
Robot learning

> Learning to make sequential decisions in the physical world
  - A system need to make multiple decisions based on stream of information

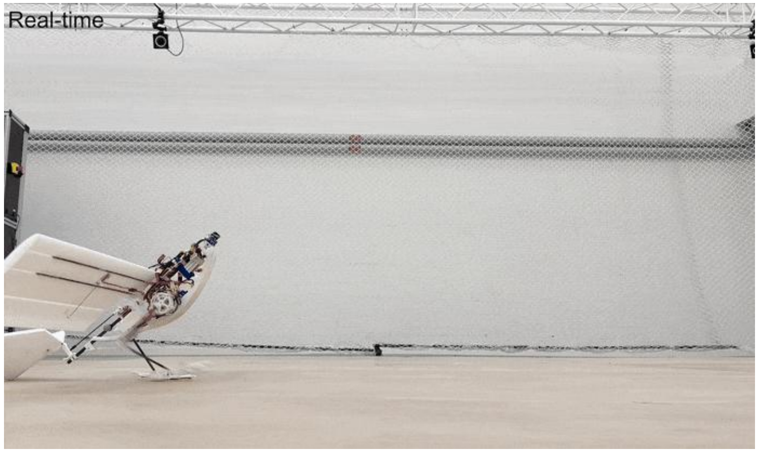
> The solutions to such problems
  - imitation learning - offline & online RL
  - model-free & model-based RL - multi-task & meta RL

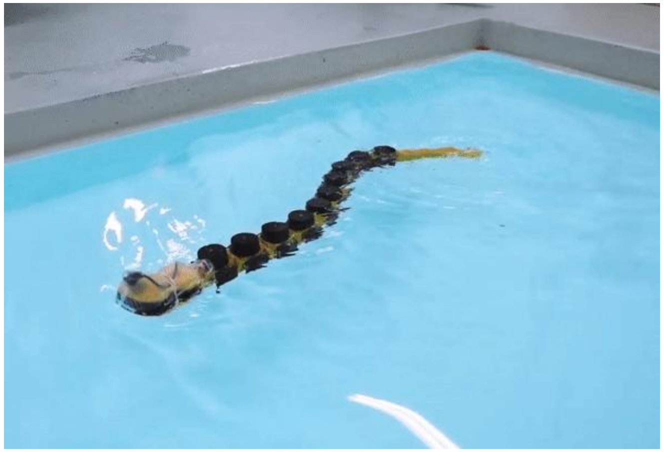


Robot control



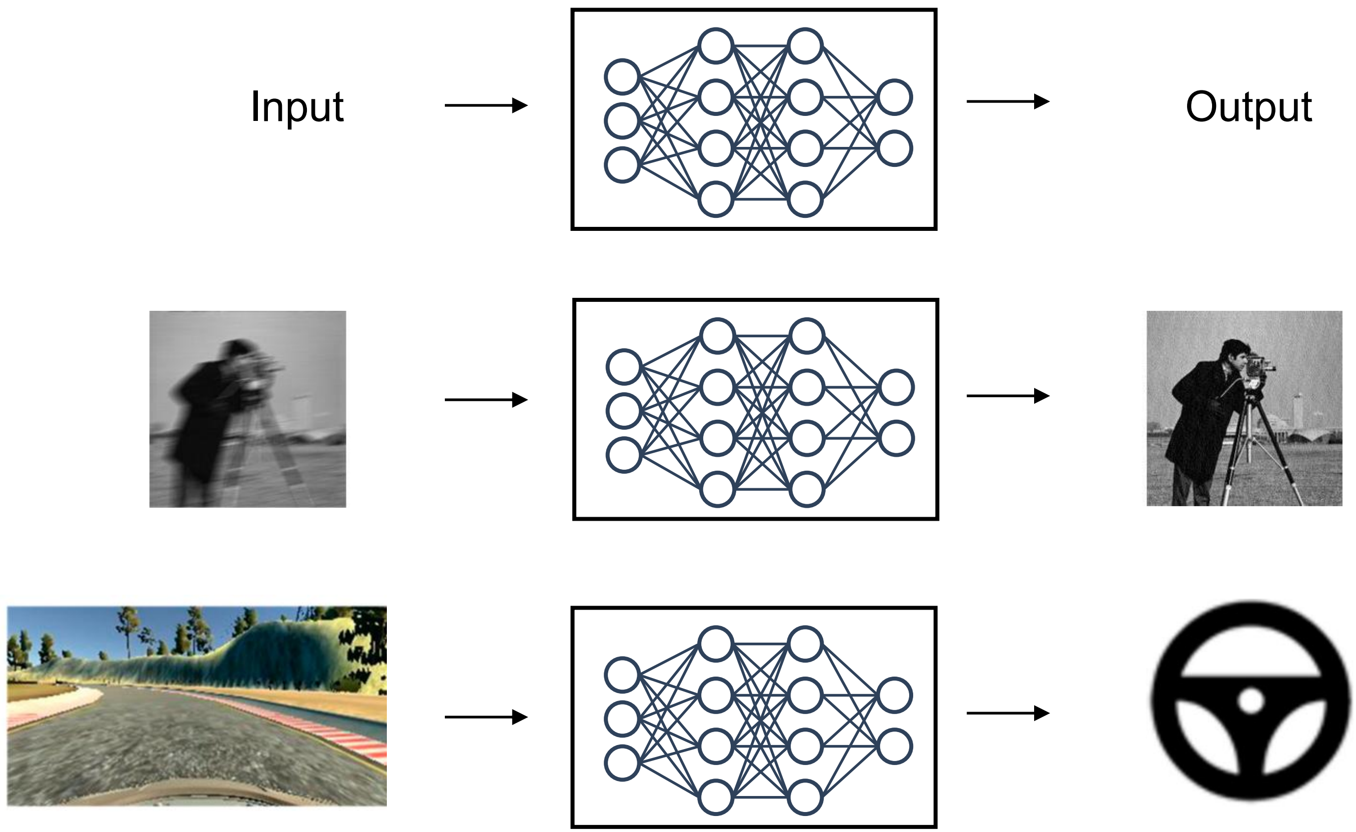
Robot control





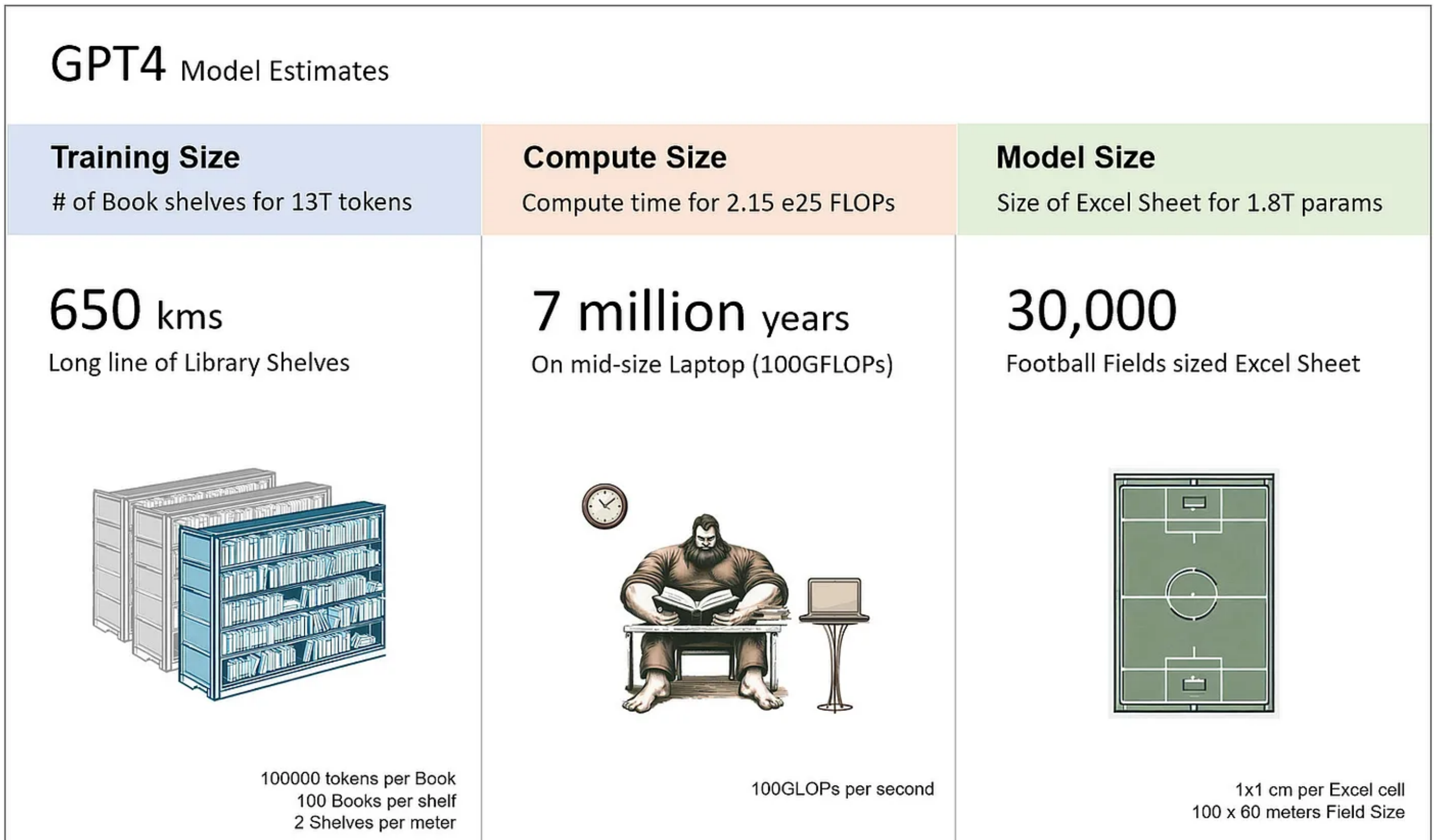
Learning nature

- Neural networks? = universal approximator of any function



Learning nature

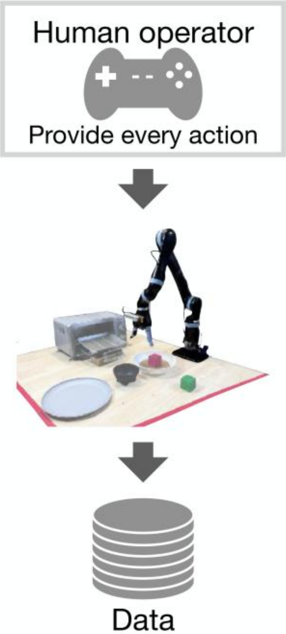
> Foundation model – Large language model (LLM). $63 million cost



Source: https://the-decoder.com/gpt-4-architecture-datasets-costs-and-more-leaked

Controller training

> Expensive expert dataset



Human Teleoperation

> Can't surpass the expert

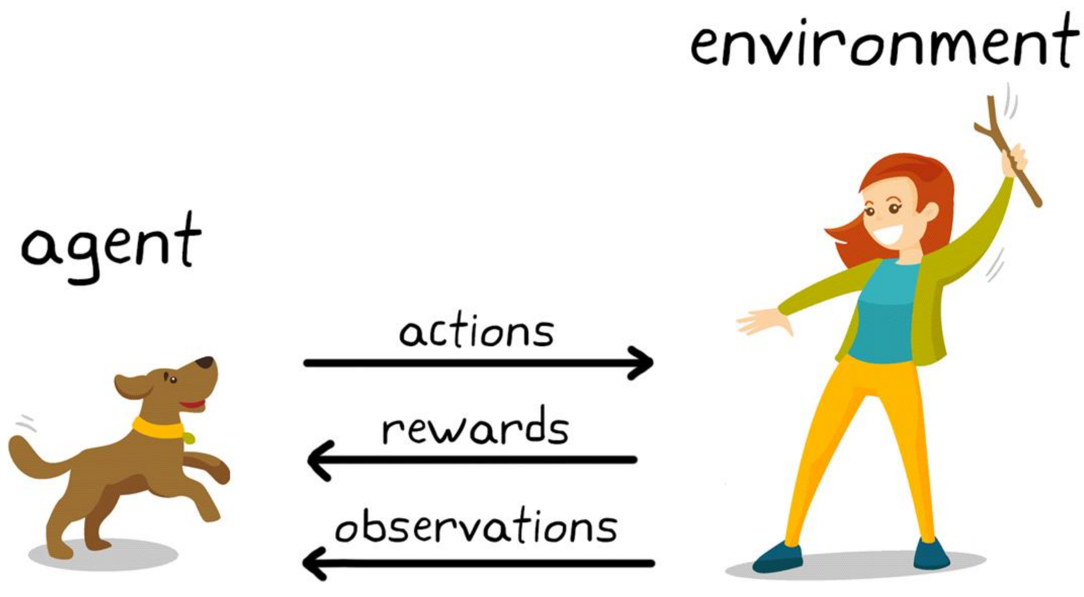
How RL differ from other ML topics?

- Supervised learning
  - Given labeled data: $\{(x_i, y_i)\}$ learn $f(x) \approx y$
  - directly told what to output
  - usually assume i.i.d. data

- Reinforcement learning
  - ground truth is not known, only know if we succeeded or failed
  - from experience, indirect feedback
  - data not i.i.d.: actions affect the future observations

Reinforcement learning

- Behaviors is primarily shaped by reinforcement rather than free-will
  - B.F. Skinner (1904-1990) Harvard psychology
  - behaviors that result in praise/pleasure tend to repeat
  - behaviors that result in punishment/paint tend to become extinct



Reinforcement learning

- Fundamental aspect of intelligence
  - enables the ability to get better with practice
- How des robot learn to represent what is good or bad for the task?
  - reward learning / inverse RL
- How can an agent generalize its behavior to many different scenarios?
  - leverage large, diverse datasets -> offline RL
  - transfer from other tasks, goals -> multitask RL, meta-RL
- Can we use the prior knowledge to enhance the performance?
  - model-based RL
- Can use RL to learn long-horizon tasks, like cooking a meal?
  - hierarchical RL
- Can we guarantee the avoidance of collision (severe accident)?
  - safe RL

Types of algorithm

- Objective
  - maximize expected sum of rewards $\max_{\theta} \mathbb{E}[\sum_t^T r(s_t, a_t)]$

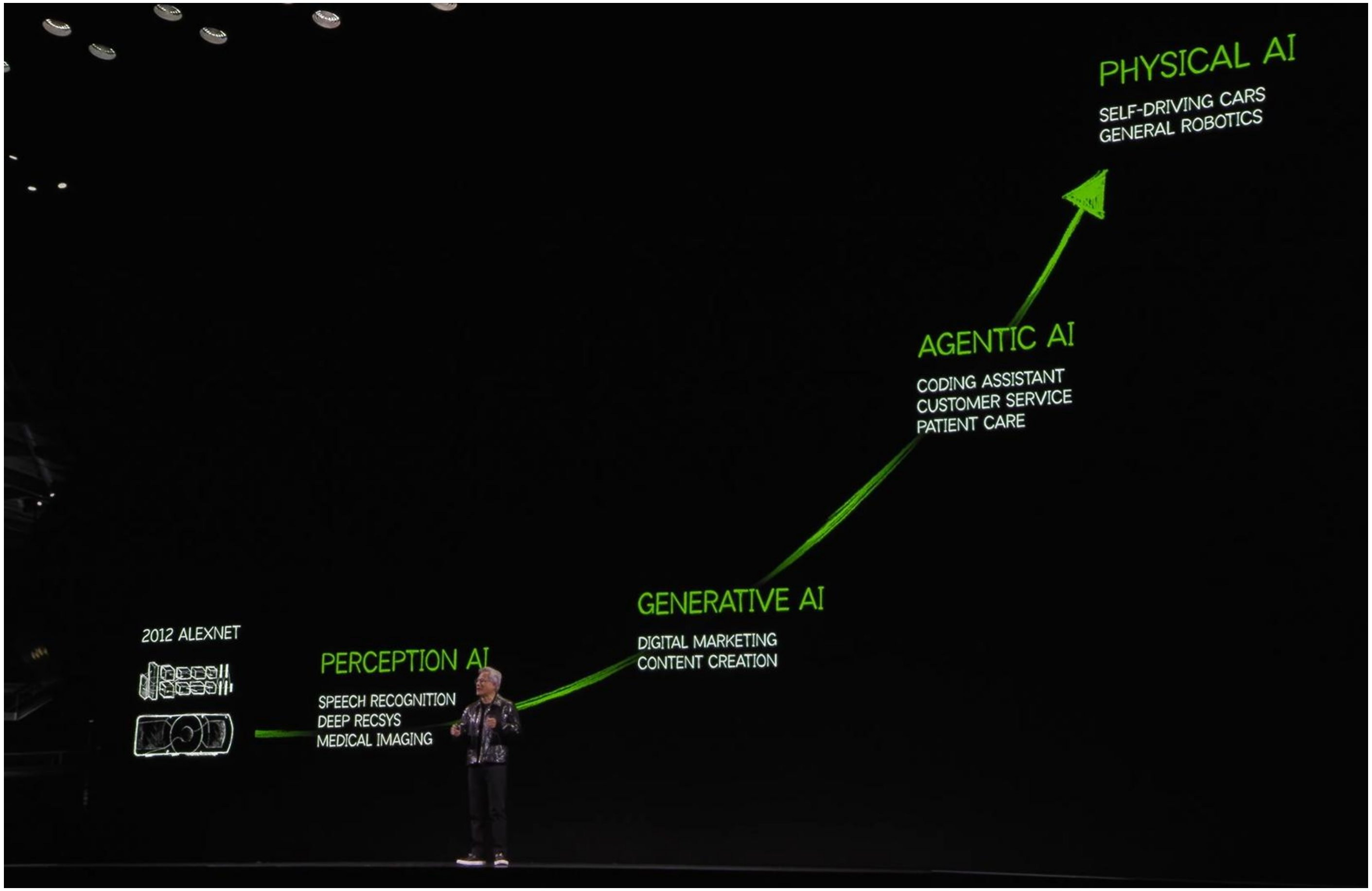
- Imitation learning: mimic a policy that achieves high reward
- Policy gradients: directly differentiate the above objective
- Actor-critic: estimate value of the current policy and use it to make the policy better
- Value-based: estimate value of the optimal policy
- Model-based: learn to model the dynamics, and use it for planning or policy improvement

Why so many algorithms?

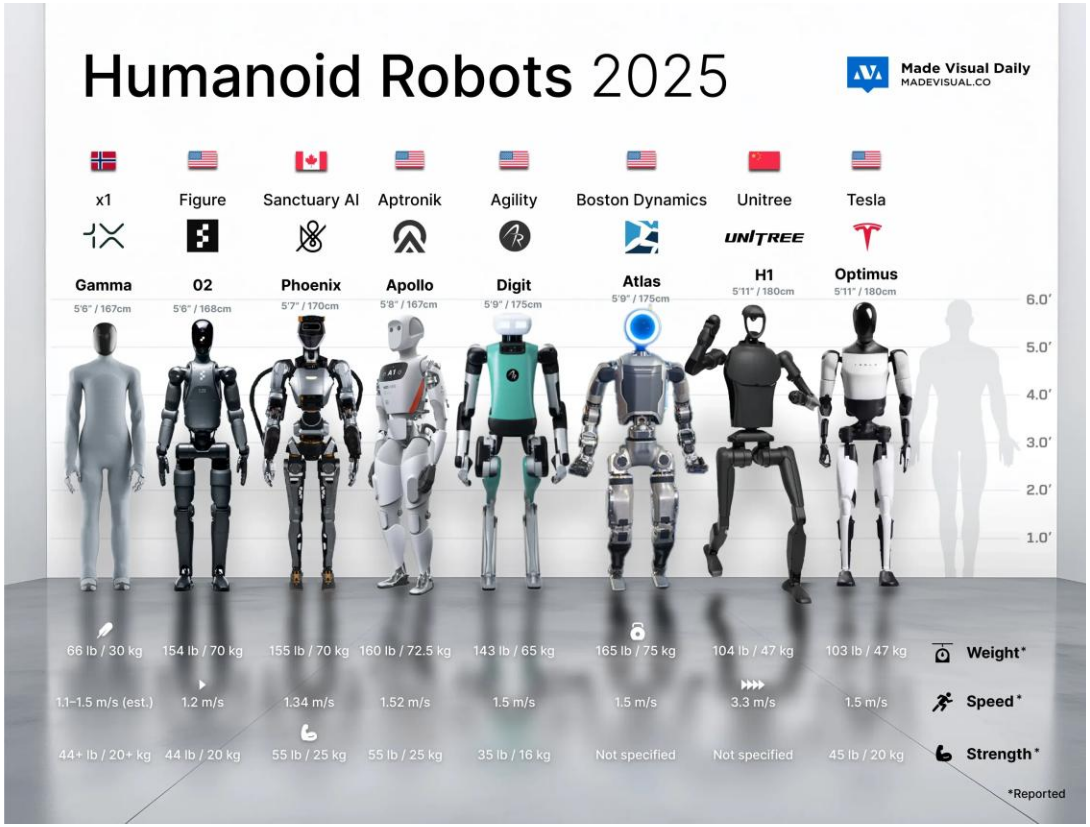
- Algorithms make different trade-offs.
  - How easy / cheap is it to collect data with policy?
  - How easy / cheap are different forms of supervision?
  - How important is stability and ease-of-use?
  - Action space dimensionality, continuous vs discrete
  - Is it easy to learn the dynamics model?

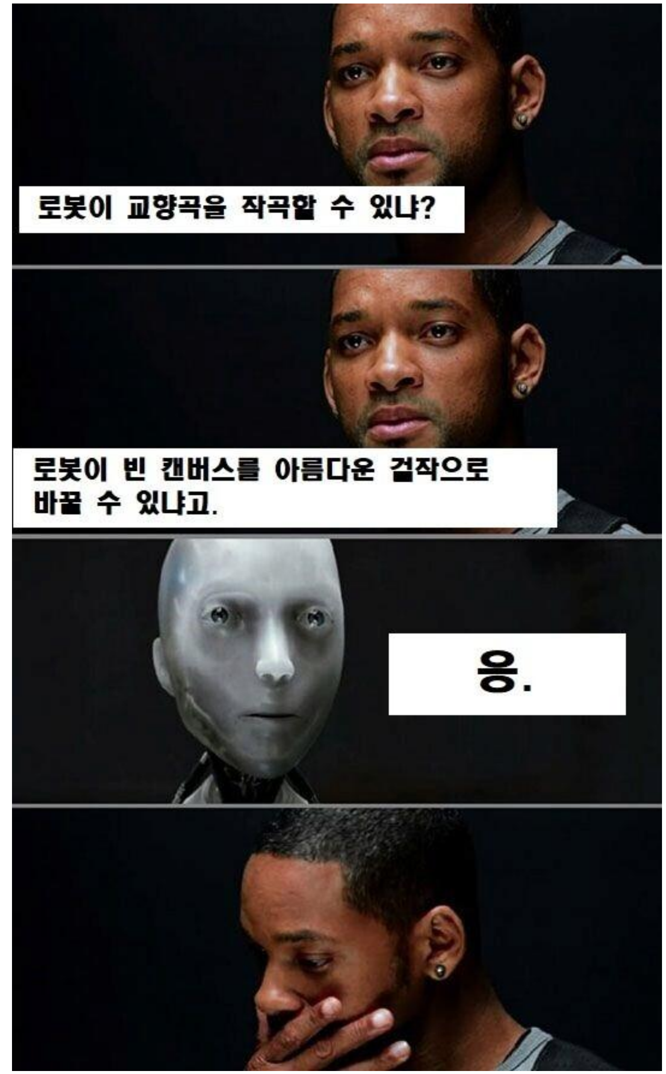
Ultimate goal

> Build general-purpose embodied intelligence by learning to make sequential decisions in the physical world.



Ultimate goal - Humanoids





Where are we today: non-learning method

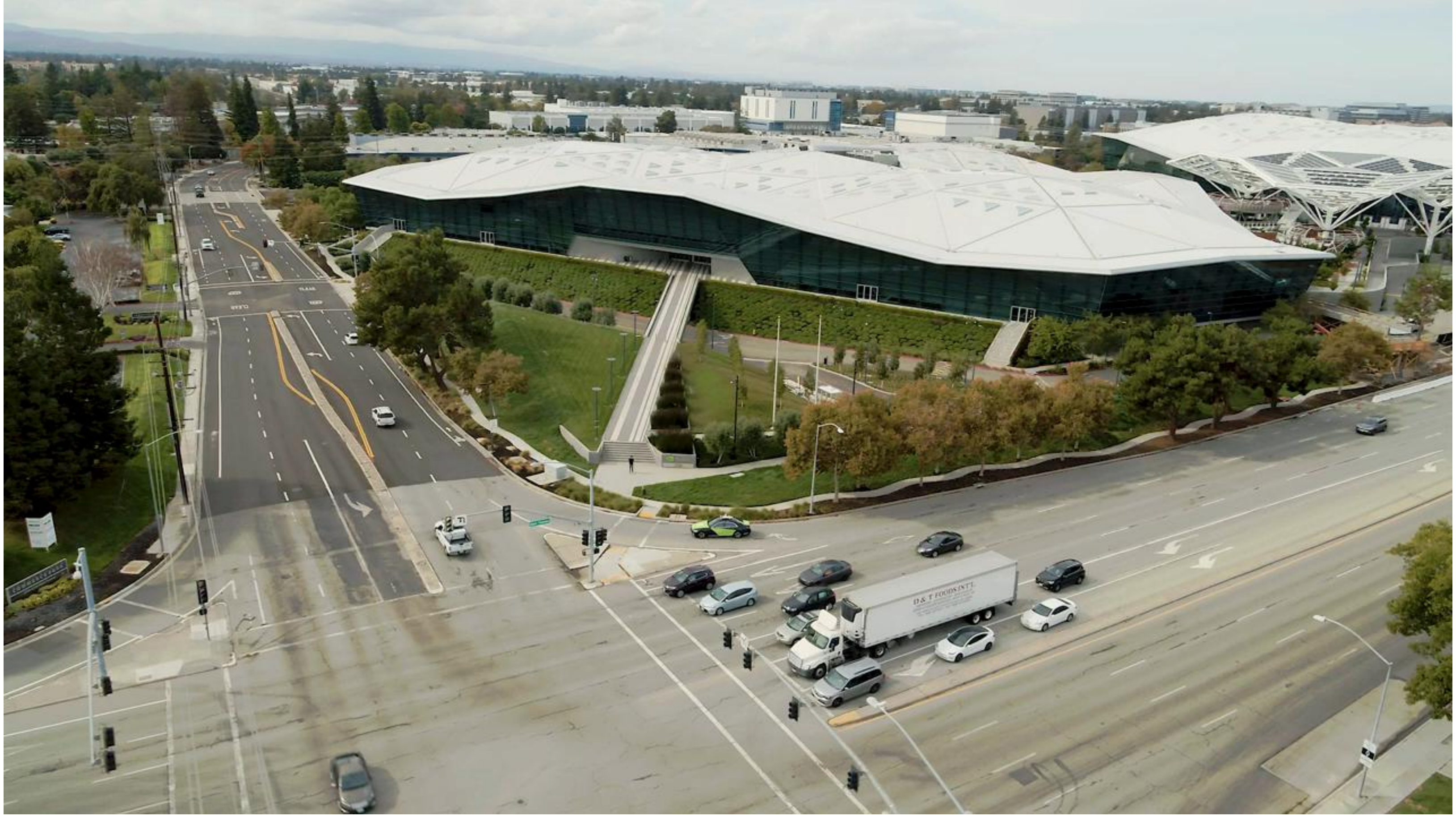
> trajectory optimization and control: optimal control + robust control

Where are we today: non-learning method

> trajectory optimization + MPC

Where are we today: learning method

> Sim2Real - NVIDIA



Where are we today: learning method

> Collect real-world data efficiently – Mobile ALOHA

Where are we today: learning method

> Control foundation model: A general navigation model (GNM)

Robust MPC

Lemma 5 (*Point Estimate*). *If* $\sup_{k\in\mathbb{N}} \|x_k\| < \infty$, $\sup_{k\in\mathbb{N}} \|u_k\| < \infty$, *then the parameter estimate* $\hat{\theta}_k$ *is bounded, in accordance with the prior parameter set, i.e.* $\hat{\theta}_k \in \Theta$, *and*

$$\sup_{m\in\mathbb{N}, w_k\in\mathbb{W}, \hat{\theta}_0\in\Theta} \frac{\sum_{k=0}^{m} \|\tilde{x}_{1|k}\|^2}{\frac{1}{\mu}\|\hat{\theta}_0 - \theta^*\|^2 + \sum_{k=0}^{m} \|w_k\|^2} \leq 1.$$

Proof. Boundedness of $\hat{\theta}_k$ and $\hat{\theta}_k \in \Theta$ follow trivially from the set update (6), (7) and projection. To prove the bound on the prediction error consider

$$\begin{aligned}
&\frac{1}{\mu}\|\hat{\theta}_{k+1} - \theta^*\|^2 - \frac{1}{\mu}\|\hat{\theta}_k - \theta^*\|^2 \\
&\leq \frac{1}{\mu}\|\tilde{\theta}_{k+1} - \theta^*\|^2 - \frac{1}{\mu}\|\hat{\theta}_k - \theta^*\|^2 \\
&= \frac{1}{\mu}\|\tilde{\theta}_{k+1} - \hat{\theta}_k\|^2 + \frac{2}{\mu}(\tilde{\theta}_{k+1} - \hat{\theta}_k)^\top(\hat{\theta}_k - \theta^*) \qquad (13) \\
&= \frac{1}{\mu}\|\mu D_k^\top(\tilde{x}_{1|k} + w_k)\|^2 + 2(\tilde{x}_{1|k} + w_k)^\top D_k(\hat{\theta}_k - \theta^*) \\
&\leq (\mu\|D_k\|^2 - 1)\|\tilde{x}_{1|k} + w_k\|^2 - \|\tilde{x}_{1|k}\|^2 + \|w_k\|^2 \\
&\leq -\|\tilde{x}_{1|k}\|^2 + \|w_k\|^2
\end{aligned}$$

Proposition 9 (*Prediction Tube*). *Let* $\{\mathbb{X}_{l|k}\}_{l\in\mathbb{N}_0^N}$ *be parametrized as in* (14) *with decision variables* $\mathbf{z}_{N|k}$, $\boldsymbol{\alpha}_{N|k}$, *and* $\mathbf{v}_{N|k}$.

Eqs. (5a)–(5c) *are satisfied if and only if for all* $j \in \mathbb{N}_1^v$, $l \in \mathbb{N}_0^{N-1}$ *there exists* $\Lambda_{l|k}^j \in \mathbb{R}_{\geq 0}^{u\times q_k}$ *such that*

$$\begin{aligned}
(F + GK)z_{l|k} + Gv_{l|k} + \alpha_{l|k}\bar{f} &\leq \mathbf{1} && (15a)\\
-H_x z_{0|k} - \alpha_{0|k}\mathbf{1} &\leq -H_x x_k && (15b)\\
\Lambda_{l|k}^j h_{\theta_k} + H_x d_{l|k}^j - \alpha_{l+1|k}\mathbf{1} &\leq -\bar{w} && (15c)\\
H_x D_{l|k}^j &= \Lambda_{l|k}^j H_{\theta_k}. && (15d)
\end{aligned}$$

Proof. Inequality (5c) is equivalent to

$$(F + GK)z_{l|k} + Gv_{l|k} + \alpha_{l|k}(F + GK)x \leq \mathbf{1} \quad \forall x \in \mathbb{X}_0,$$

which is equivalent to (15a) when maximized over $x \in \mathbb{X}_0$.

Inequality (5a) is equivalent to (15b), and (5b) is equivalent to (15c), (15d) as shown by the following reformulation.

$$\begin{aligned}
&\mathbb{X}_{l+1|k} \supseteq A_{cl}(\theta)\mathbb{X}_{l|k} \oplus B(\theta)v_{l|k} \oplus \mathbb{W} \qquad \forall\theta \in \Theta_k \\
\Leftrightarrow\ & H_x(A_{cl}(\theta)x + B(\theta)v_{l|k} + w - z_{l+1|k}) \leq \alpha_{l+1|k}\mathbf{1} \\
&\qquad \forall x \in \mathbb{X}_{l|k}, \theta \in \Theta_k, w \in \mathbb{W} \\
\Leftrightarrow\ & H_x(A_{cl}(\theta)(z_{l|k} + \alpha_{l|k}x^j) + B(\theta)v_{l|k} - z_{l+1|k}) \\
&\qquad - \alpha_{l+1|k}\mathbf{1} \leq -\bar{w} \qquad \forall j \in \mathbb{N}_1^v, \theta \in \Theta_k \\
\Leftrightarrow\ & \max_{\theta\in\Theta_k} \left\{H_x(A_{cl}(\theta)(z_{l|k} + \alpha_{l|k}x^j) + B(\theta)v_{l|k})\right\} \\
&\qquad - H_x z_{l+1|k} - \alpha_{l+1|k}\mathbf{1} \leq -\bar{w} \qquad \forall j \in \mathbb{N}_1^v \\
\Leftrightarrow\ & \max_{\theta\in\Theta_k} \left\{H_x D_{l|k}^j \theta\right\} + H_x d_{l|k}^j - \alpha_{l+1|k}\mathbf{1} \leq -\bar{w} \qquad \forall j \in \mathbb{N}_1^v \\
\Leftrightarrow\ & \left\{\begin{array}{l}
\Lambda_{l|k}^j h_{\theta_k} + H_x d_{l|k}^j - \alpha_{l+1|k}\mathbf{1} \leq -\bar{w} \\
H_x D_{l|k}^j = \Lambda_{l|k}^j H_{\theta_k} \\
\Lambda_{l|k}^j \in \mathbb{R}_{\geq 0}^{u\times q_k}
\end{array}\right\} \qquad \forall j \in \mathbb{N}_1^v
\end{aligned}$$

Deterministic policy gradient

Performance measure: $J(\mu_\theta) \equiv \int_S \; p_0(s)\nabla_\theta v^{\mu_\theta}(s)ds$ $\quad\rho^\mu$: discounted state distribution $\quad$ Objective: find $\nabla_\theta J(\mu_\theta)$

$$\nabla_\theta v^{\mu_\theta}(s) = \nabla_\theta q^{\mu_\theta}(s, \mu_\theta(s))$$

$$= \nabla_\theta \left(r(s, \mu_\theta(s)) + \int_S \gamma\, p(s'|s, \mu_\theta(s)) v^{\mu_\theta}(s')ds' \right)$$

Why is this not zero like in the stochastic case

$$= \nabla_\theta \mu_\theta(s) \nabla_a r(s,a)|_{a=\mu_\theta(s)} + \nabla_\theta \int_S \gamma\, p(s'|s, \mu_\theta(s)) v^{\mu_\theta}(s')ds'$$

$$= \nabla_\theta \mu_\theta(s) \nabla_a r(s,a)|_{a=\mu_\theta(s)} + \int_S \gamma \left(p(s'|s, \mu_\theta(s)) \nabla_\theta v^{\mu_\theta}(s') + \nabla_\theta \mu_\theta(s) \nabla_a p(s'|s, a\,)|_{a=\mu_\theta(s)} v^{\mu_\theta}(s') \right) ds'$$

$$= \nabla_\theta \mu_\theta(s) \nabla_a \left(r(s,a) + \int_S \gamma\, p(s'|s, a\,) v^{\mu_\theta}(s')ds' \right)\Bigg|_{a=\mu_\theta(s)} + \int_S \gamma\, p(s'|s, \mu_\theta(s)) \nabla_\theta v^{\mu_\theta}(s')ds'$$

$$= \nabla_\theta \mu_\theta(s) \nabla_a\, q^{\mu_\theta}(s,a)\big|_{a=\mu_\theta(s)} + \int_S \gamma\, \underline{p(s \to s', 1, \mu_\theta)} \nabla_\theta v^{\mu_\theta}(s')ds'$$

The prob' of state transition in 1 step following the policy

$$\nabla_\theta v^{\mu_\theta}(s) = \nabla_\theta \mu_\theta(s) \nabla_a\, q^{\mu_\theta}(s,a)\big|_{a=\mu_\theta(s)} + \int_S \gamma\, p(s \to s', 1, \mu_\theta) \underbrace{\nabla_\theta v^{\mu_\theta}(s')}_{\text{recursion}} ds'$$

$$= \mu_\theta(s) \nabla_a\, q^{\mu_\theta}(s,a)\big|_{a=\mu_\theta(s)}$$

$$+ \int_S \gamma\, p(s \to s', 1, \mu_\theta) \left(\nabla_\theta \mu_\theta(s') \nabla_a\, q^{\mu_\theta}(s',a)\big|_{a=\mu_\theta(s)} + \int_S \gamma\, p(s' \to s'', 1, \mu_\theta) \nabla_\theta v^{\mu_\theta}(s'')ds'' \right) ds'$$

$$= \mu_\theta(s) \nabla_a\, q^{\mu_\theta}(s,a)\big|_{a=\mu_\theta(s)} + \int_S \gamma\, p(s \to s', 1, \mu_\theta) \nabla_\theta \mu_\theta(s') \nabla_a\, q^{\mu_\theta}(s',a)\big|_{a=\mu_\theta(s)} ds' + \int_S \gamma^2\, p(s \to s'', 2, \mu_\theta) \underbrace{\nabla_\theta v^{\mu_\theta}(s'')}_{\text{recursion}} ds''$$

$$= \int_S \sum_{t=0}^{\infty} \gamma^t p(s \to s', t, \mu_\theta)\, \nabla_\theta \mu_\theta(s') \nabla_a\, q^{\mu_\theta}(s',a)\big|_{a=\mu_\theta(s)} ds'$$

What will you take away?

- Algorithms can be math-heavy.
  - Understanding is important, but not for the beginners

- Rather than knowing the all backgrounds, focusing on
  - core concepts behind deep RL methods
  - implementation of algorithms
  - examples in robotics, control
  - topics that we think are most exciting

- Core class goal: able to understand and implement existing and emerging methods

Pre-requisites

- Some familiarity with machine learning, deep learning and RL
- Basic optimization such as gradient descent
- Some calculus and probability theory

Coursework

- Assignments: (35%)
  - Implement different methods in PyTorch, run experiments in physics simulators and compete with other students.
  - deep RL methods take time to learn behavior!
- Project: (50%)
  - teams of 2-3 students, encouraged to use your research if applicable
  - propose your own topic
  - proposal presentation – midterm period
  - final presentation – final period
- Paper reviews (15%)
  - review SOTA research papers
  - https://docs.google.com/spreadsheets/d/1m5j8pU7EXMzTuexpjlxNeKyY_rIWr1l49XoMoBbMPSg/edit?usp=sharing

- No exams

Syllabus

Week 1	Introduction	Week 9	Model-based RL
Week 2	Imitation learning	Week 10	Exploration
Week 3	MDP basics and simulation	Week 11	Offline RL
Week 4	RL basics	Week 12	Safe RL and Sim2Real
Week 5	Policy gradient (model-free RL)	Week 13	Inverse RL, curriculum learning
Week 6	Actor-critic method (model-free RL 2)	Week 14	Paper review
Week 7	optimal control and planning	Week 15	Final project
Week 8	Project proposal		

Reference

> Lectures

- Sergey Levine, UC Berkeley: https://rail.eecs.berkeley.edu/deeprlcourse/
- Katerina Fragkiadak, CMU: https://16-831-s24.github.io/lectures
- Guanya Shi, CMU: https://cmudeeprl.github.io/403website_s23/lectures/
- Chelsea Finn, Stanford: https://cs224r.stanford.edu/
- Joschka Boedecker and Moritz Diehl, U of Freiburg https://www.syscop.de/teaching/ss2021/model-predictive-control-and-reinforcement-learning