This lecture is based in part on, or inspired by, slides/blogs/demos from:

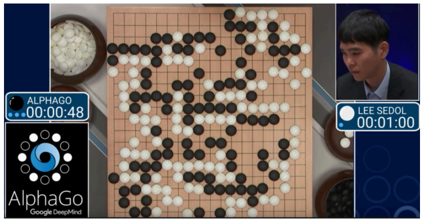
- Rich Sutton, U Alberta [Sutton] http://incompleteideas.net/book/the-book-2nd.html
- Emma Brunskill, Stanford [Brunskill] https://web.stanford.edu/class/cs234/index.html
- David Silver, UCL [Silver] http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html
- Pieter Abbeel, UC Berkeley [Abbeel]
- Sergey Levine, Chelsea Finn, John Schulman, UC Berkeley [UC Berkeley]
- Mark Schmidt, UBC [Schmidt]
- Andrej Karpathy [Karpathy] https://karpathy.github.io/2016/05/31/rl/
- Lillian Weng [Weng] https://lilianweng.github.io/lil-log/2018/04/08/policy-gradient-algorithms.html

Goals for this morning

- what is RL?
  - what can it solve? how does it differ from other ML problems?
- why Deep RL, and some caveats

- RL basics
- RL algorithms
  - Q-learning, DDPG, gradient-free methods, policy-gradient methods
- current perspectives

What is RL?



[DeepMind]

[Mobileye]

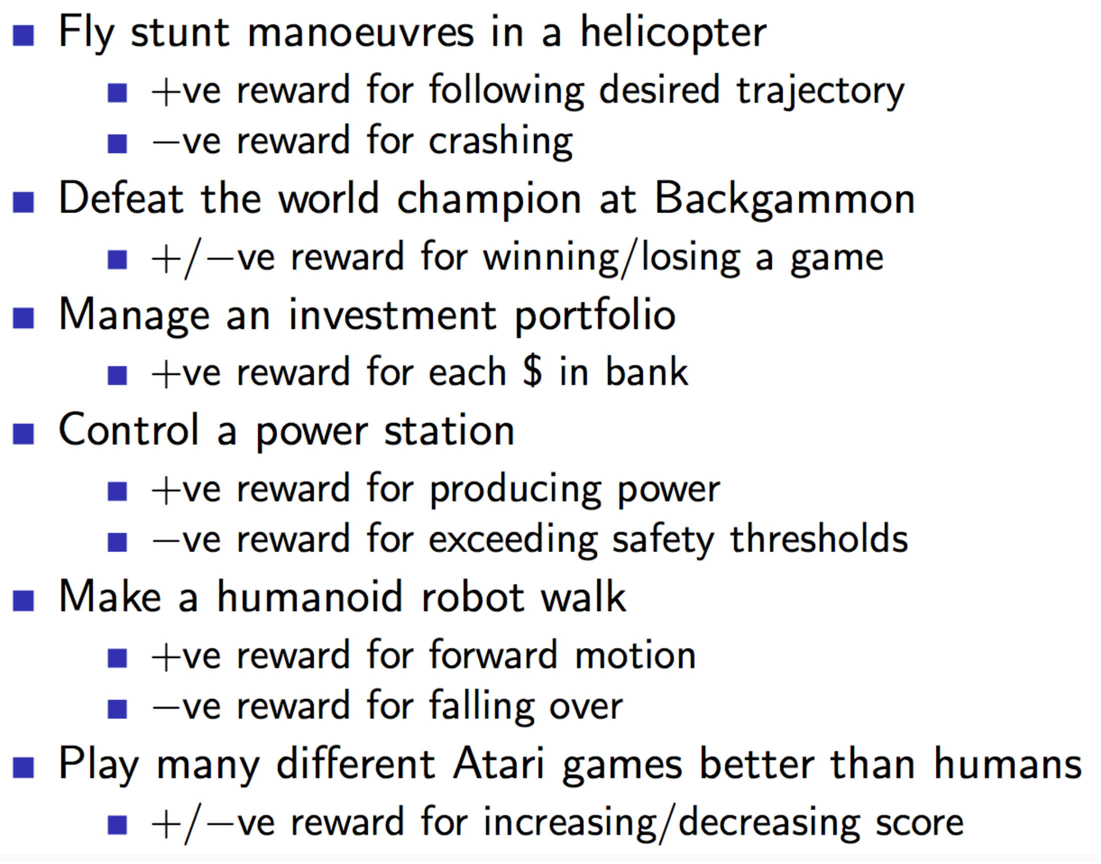
[Anybotics]

Learn to make good sequences of decisions

[Brunskill]

- Drive a car
- Defeat the world champion at Go
- Manage an investment portfolio
- Sequence a series of medical tests and interventions
- Control a power station or a chemical process to maximize revenue
- Make a humanoid robot walk
- Direct attention for a computer vision task
- Understand the role of *dopamine* in the brain

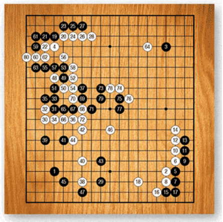
Defining good decisions: rewards



- Fly stunt manoeuvres in a helicopter
  - +ve reward for following desired trajectory
  - −ve reward for crashing
- Defeat the world champion at Backgammon
  - +/−ve reward for winning/losing a game
- Manage an investment portfolio
  - +ve reward for each $ in bank
- Control a power station
  - +ve reward for producing power
  - −ve reward for exceeding safety thresholds
- Make a humanoid robot walk
  - +ve reward for forward motion
  - −ve reward for falling over
- Play many different Atari games better than humans
  - +/−ve reward for increasing/decreasing score

[Abbeel]

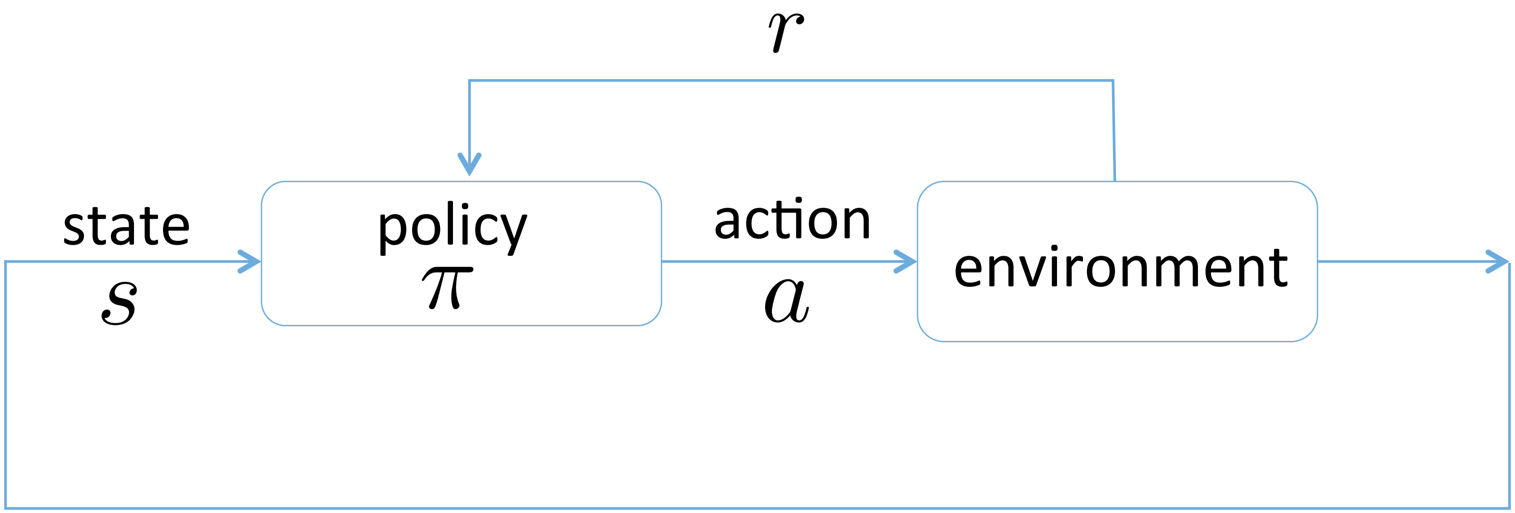
RL involves:



- optimization
  - find an optimal way to make sequential decisions
- delayed consequences
  - decisions now can impact things much later, e.g., climate change
  - challenge: temporal credit assignment is hard (what caused later high or low rewards?)
  - challenge: need to reason about long-term ramifications
- exploration
  - explore vs exploit tradeoff: try a new restaurant or go to one you already like?
  - current policy impacts future data collection: potential instabilities
- generalization
  - impossible to visit all states during learning, e.g. image input to a policy

~[Brunskill]

Reinforcement Learning



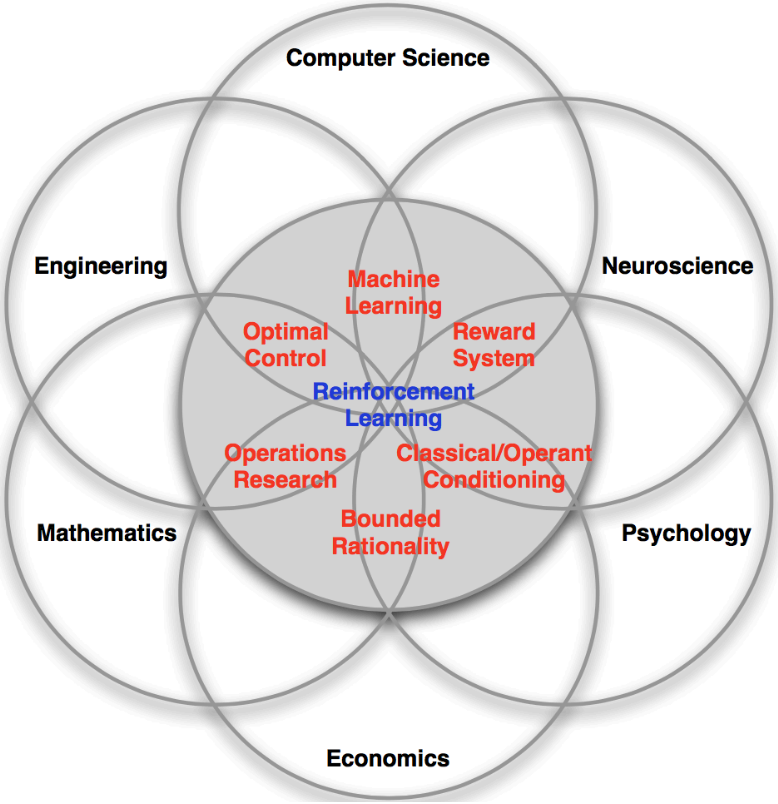
maximize sum of rewards: $G_t = r_{t+1} + \gamma r_{t+2} + \ldots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$

$$0 \leq \gamma \leq 1$$

Discrete vs Continuous

- discrete states, discrete actions
  - e.g., Go, Chess, tic-tac-toe
  - "tabular" policies for small problems
- continuous states, discrete actions
  - e.g., Atari games, DOTA, cart-and-pole
- continuous states, continuous actions
  - robotics, process control, autonomous driving

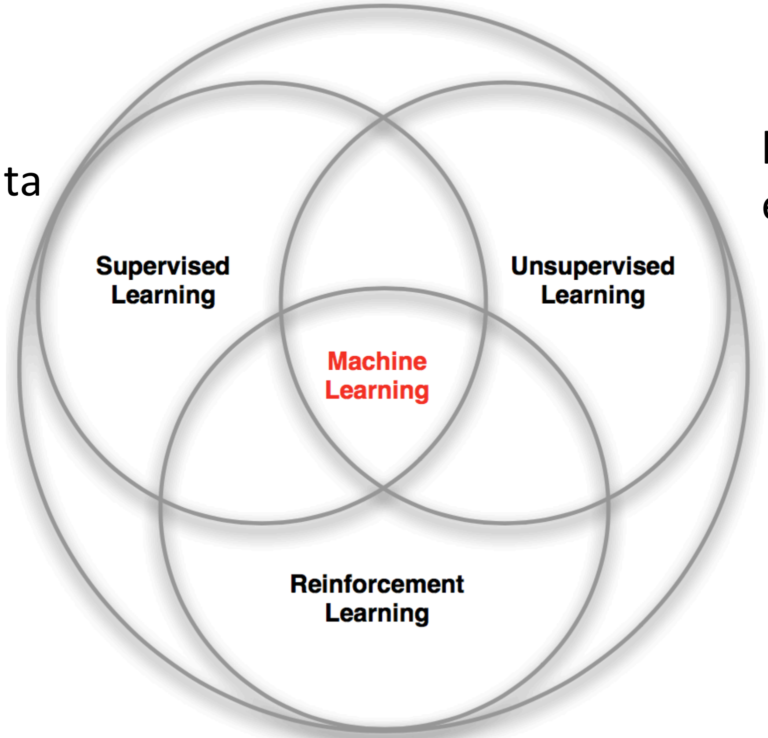
The Many Faces of Reinforcement Learning



[Silver]

Branches of Machine Learning

labeled data



latent structure ,
e.g., autoencoders

scalar rewards over time

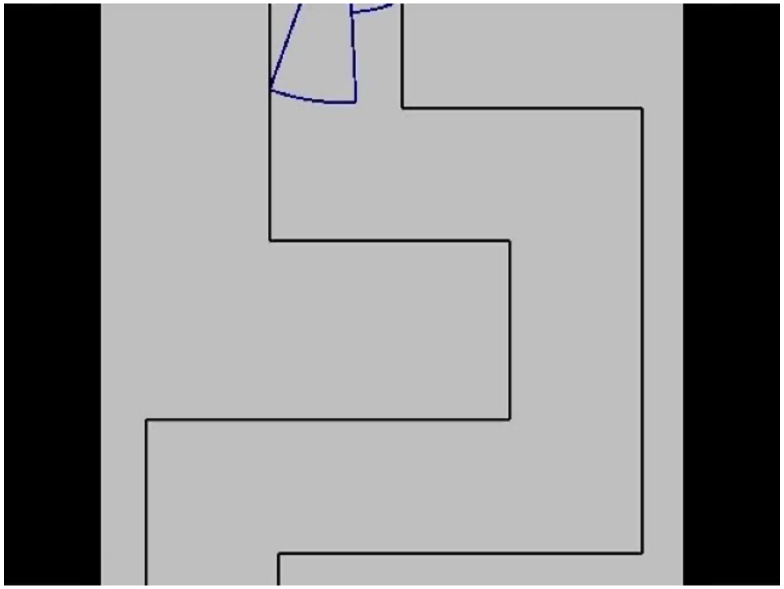
[Silver]

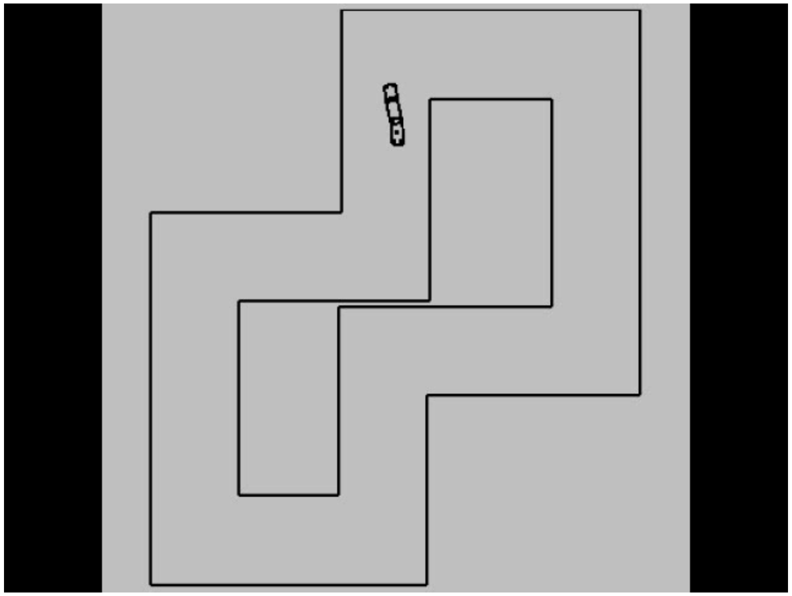
Simple Example: Backing up a Truck

$\pi : \mathbb{R}^5 \to \mathbb{R}$

input: 4 distances + cab-angle
output: steering angle

backing up a double trailer

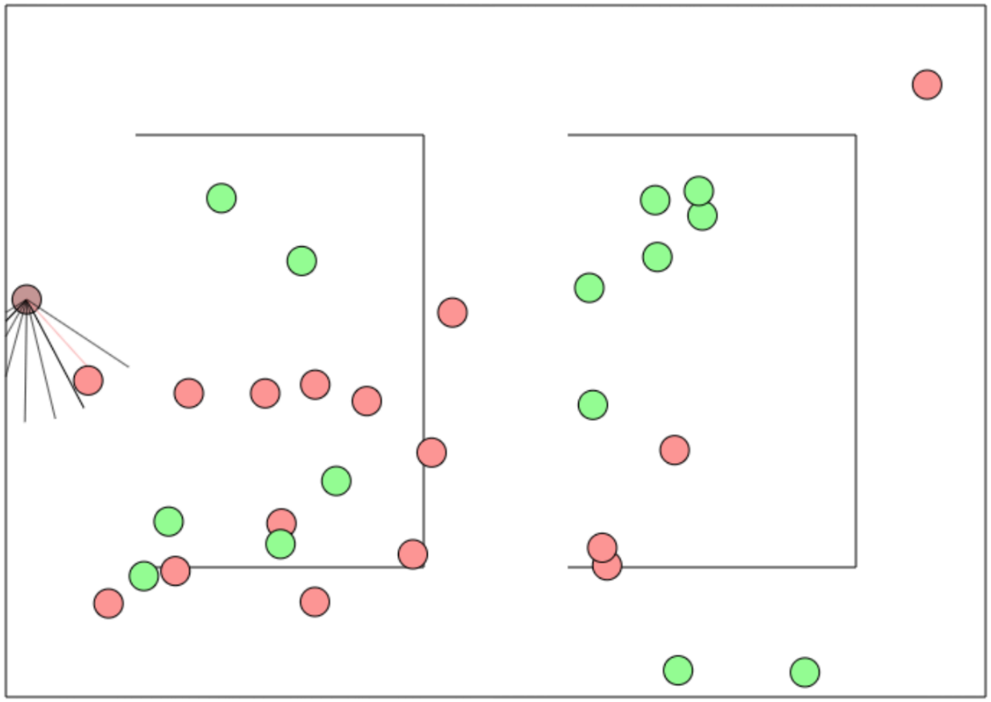




[“Learning to Steer on Winding Tracks Using Semi-Parametric Control Policies”, ICRA 2005]

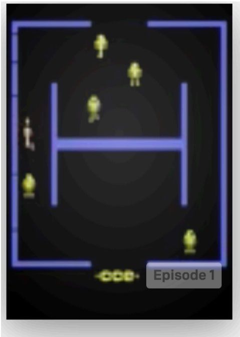
Simple Example

state: 9 sensors x 3 values x 2 timesteps
actions: move in 5 directions

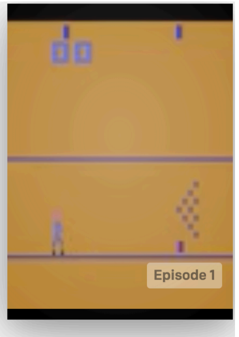


[https://cs.stanford.edu/people/karpathy/convnetjs/demo/rldemo.html]

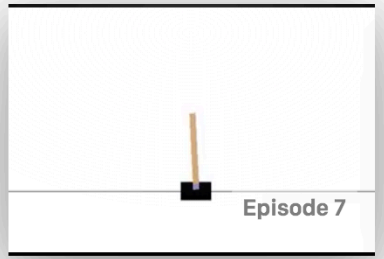
OpenAI Gym [gym.openai.com]



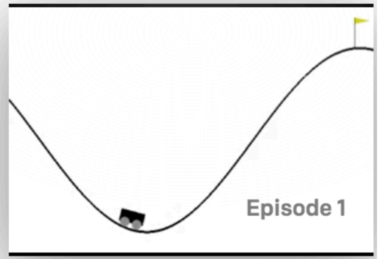
Berzerk-v0



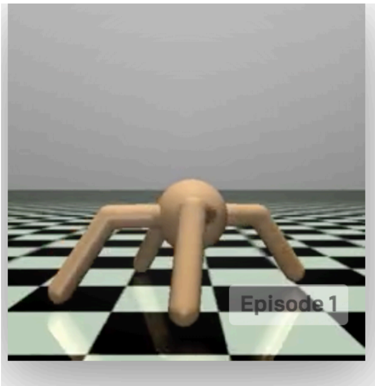
Bowling-ram-v0



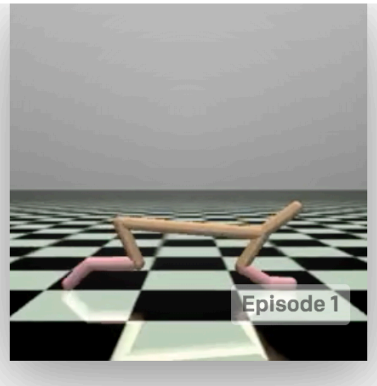
CartPole-v1



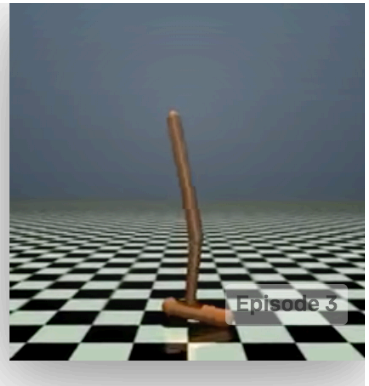
MountainCar-v0



Ant-v2



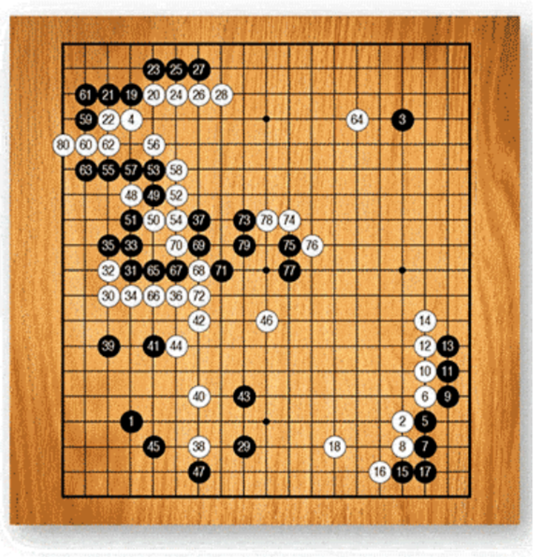
HalfCheetah-v2



Hopper-v2

Why Deep RL?

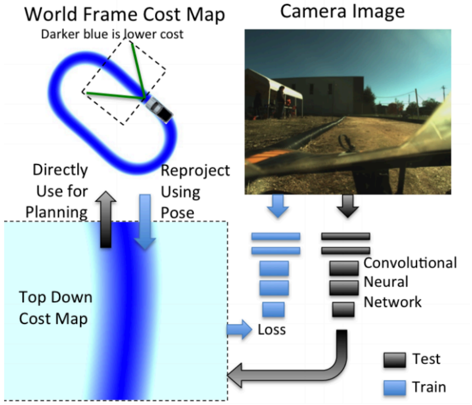
AlphaZero (DeepMind)



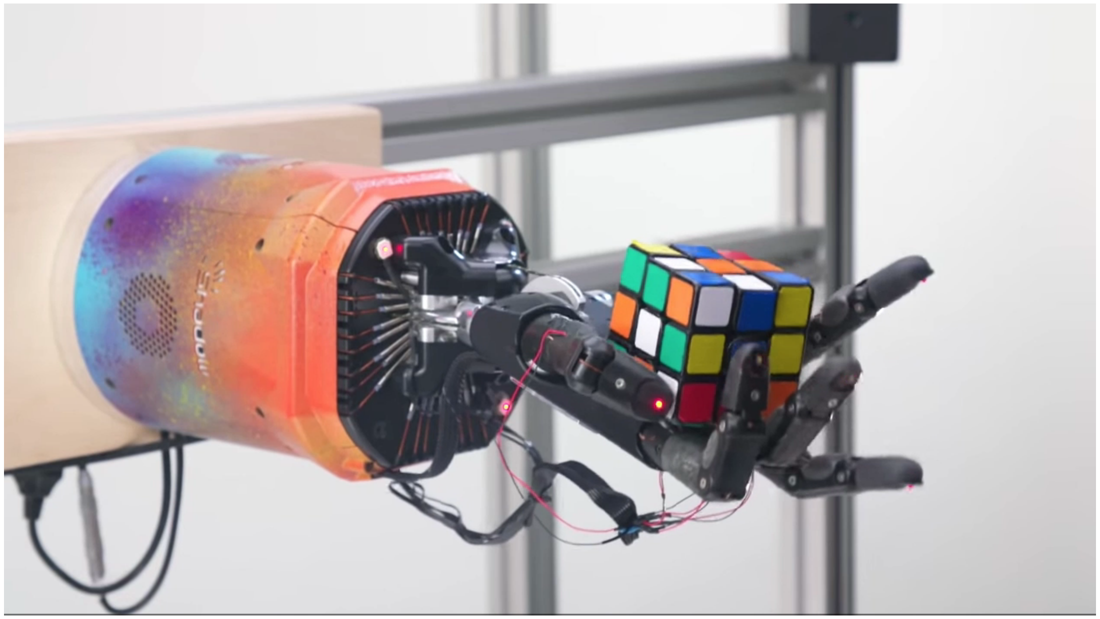
[DeepMind: Nature 2017]

Aggressive autonomous driving of a scale vehicle (Georgia Tech)

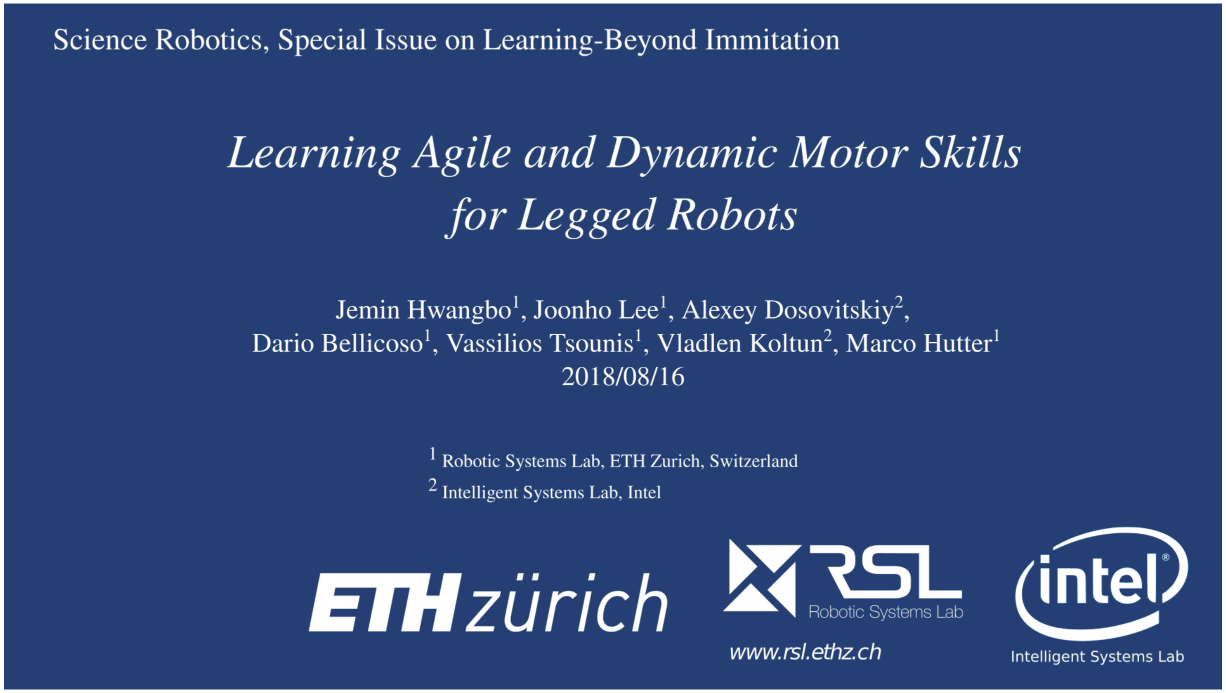
[Georgia Tech: CoRL 2017]



Rubik's cube manipulation (OpenAI)



Anymal: Quadrupedal locomotion + getup (ETH Zurich)



Science Robotics, Special Issue on Learning-Beyond Immitation

Learning Agile and Dynamic Motor Skills
for Legged Robots

Jemin Hwangbo[1], Joonho Lee[1], Alexey Dosovitskiy[2],
Dario Bellicoso[1], Vassilios Tsounis[1], Vladlen Koltun[2], Marco Hutter[1]
2018/08/16

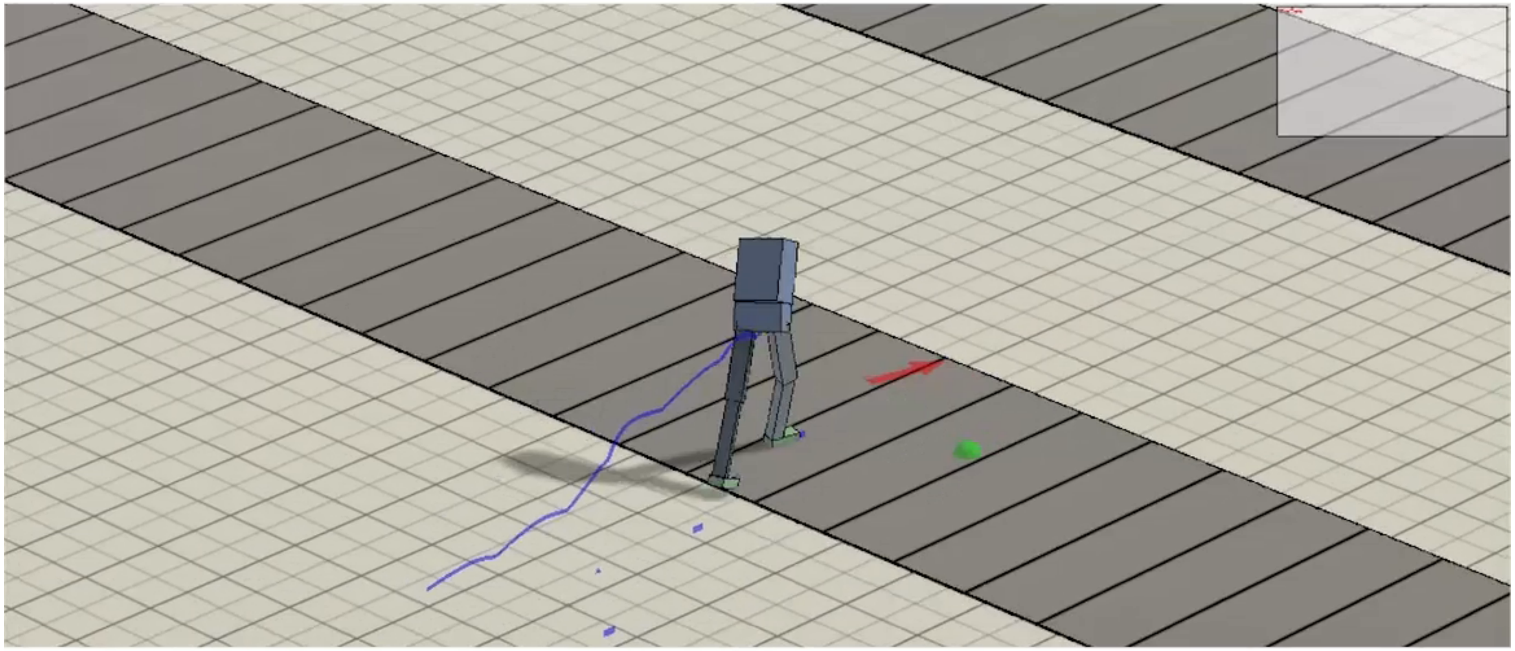
[1] Robotic Systems Lab, ETH Zurich, Switzerland
[2] Intelligent Systems Lab, Intel

ETH zürich
RSL
Robotic Systems Lab
www.rsl.ethz.ch
intel
Intelligent Systems Lab

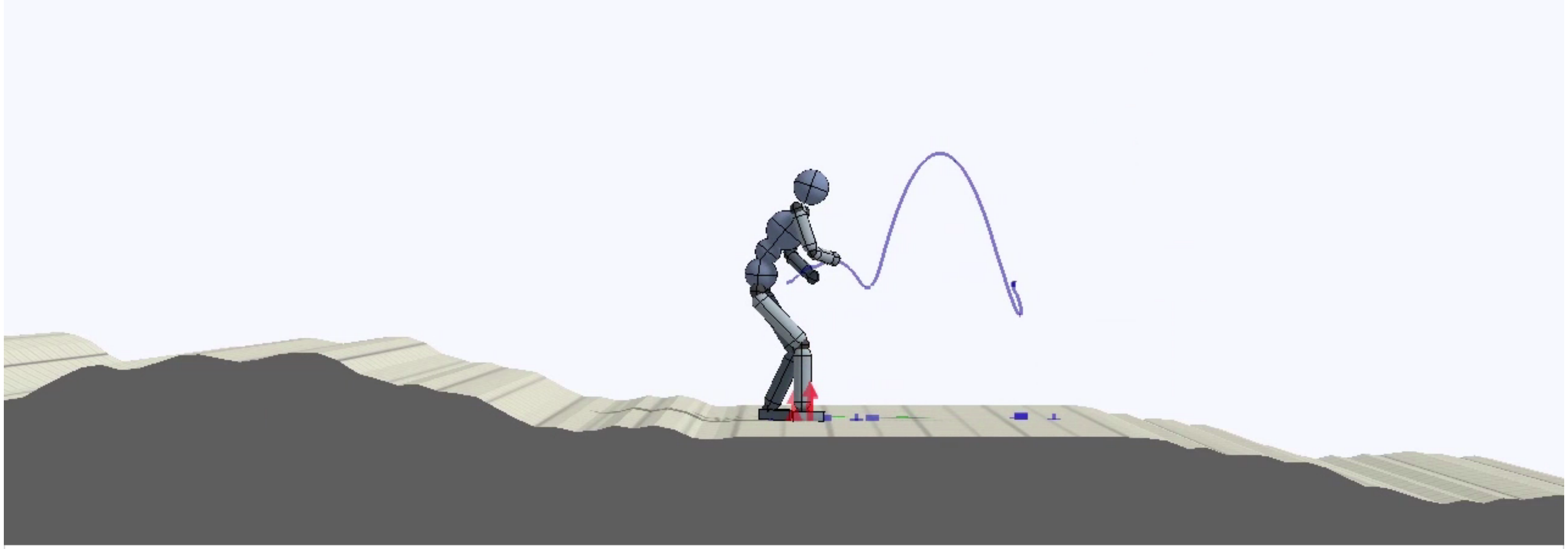
[Science Robotics, 2018]

[DeepLoco: SIGGRAPH 2017]

Walking on Conveyor Belts

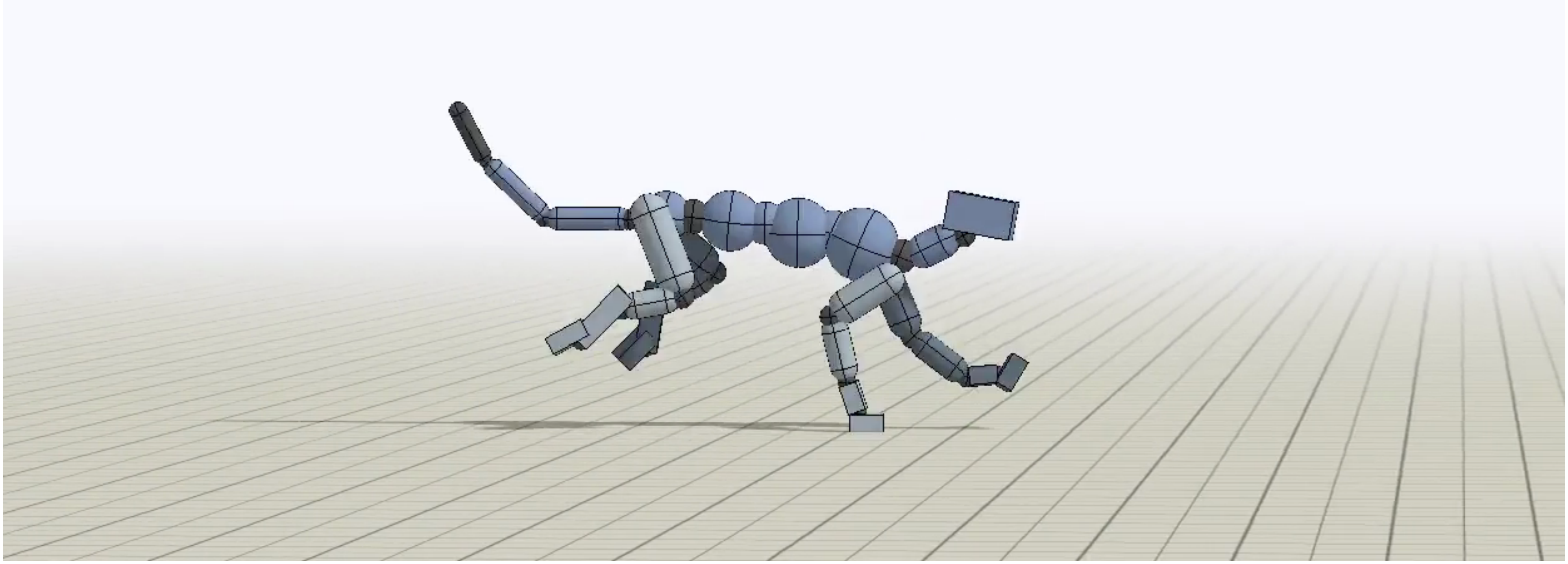


Retargeting



and retarget to different environments.

simulated lion



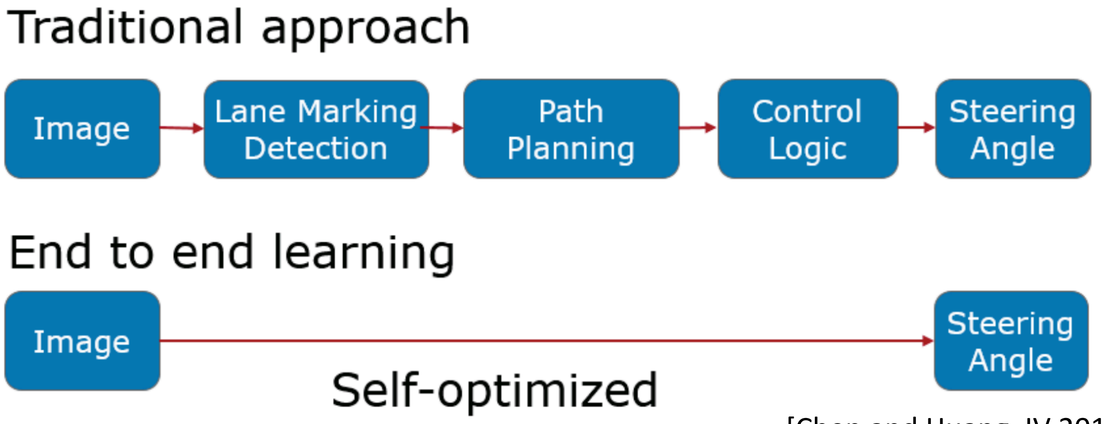
Physics-based Lion (UC Berkeley, UBC, Ziva Dynamics)

[“DeepMimic”, SIGGRAPH 2018]

Cassie:Bipedal Locomotion (UBC, U Oregon, Agility Robotics)

Why Deep RL? (continued)

End-to-end learning: performance benefits



[Chen and Huang, IV 2017]

“pixels to torques”

Good control from rich sensory streams is often the limiting factor

[https://robotik.dfki-bremen.de/en/research/robot-systems/exoskelett-aktiv-ca.html]

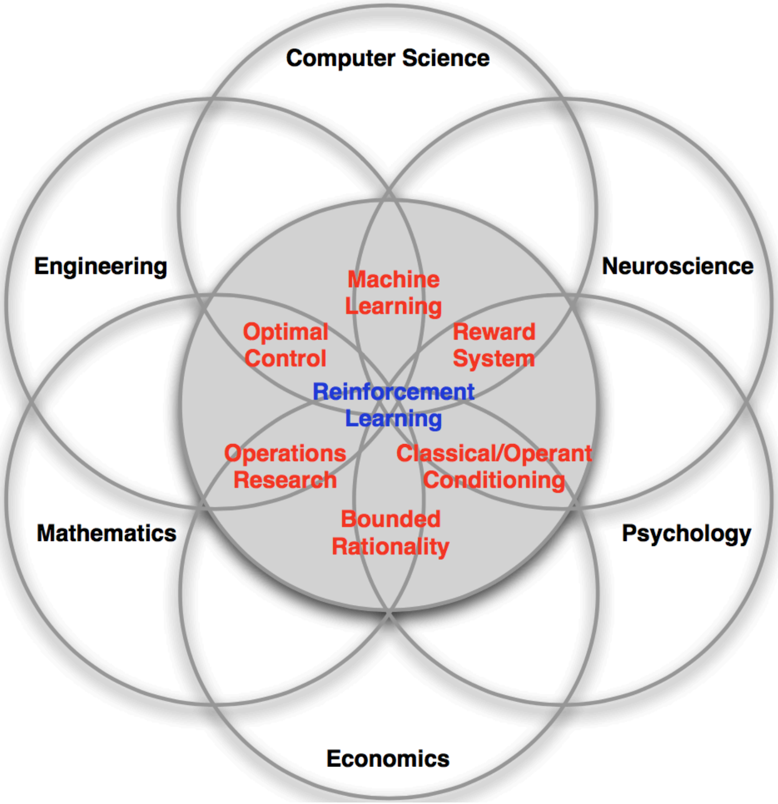
How do humans and animals learn?

Follow the money

- Alphabet invests ~ $2B in DeepMind
- Microsoft invests $1B in OpenAI
- autonomous driving: Mobileye, Wayve, Waymo, Tesla, …

Some Caveats

RL has a long history (by many names)



[Silver]

Generic Algorithm?
Best results often still have domain knowledge

- domain knowledge
  - defining: state, action, reward
  - hyperparameters
  - example expert data

but:

- no-free lunch: *inductive bias* needed for efficient learning
- progress is being made towards generalizable methods !
  - AlphaZero (Go, Chess), DQN (Atari games), OpenAI five (DOTA game)

RL learning often has poor "sample efficiency"

- often far too slow to learn directly in the real world
  - e.g., thousands of years in simulation (OpenAI hand)
- learning requires good simulation models, so not truly "model free"
- many newborn animals can walk within minutes or hours: giraffe, horse foal, piglets, camels, zebra and more

but:

- compute is cheap
- nature: has done much learning on an evolutionary time scale; is similar "transfer learning" possible for RL?

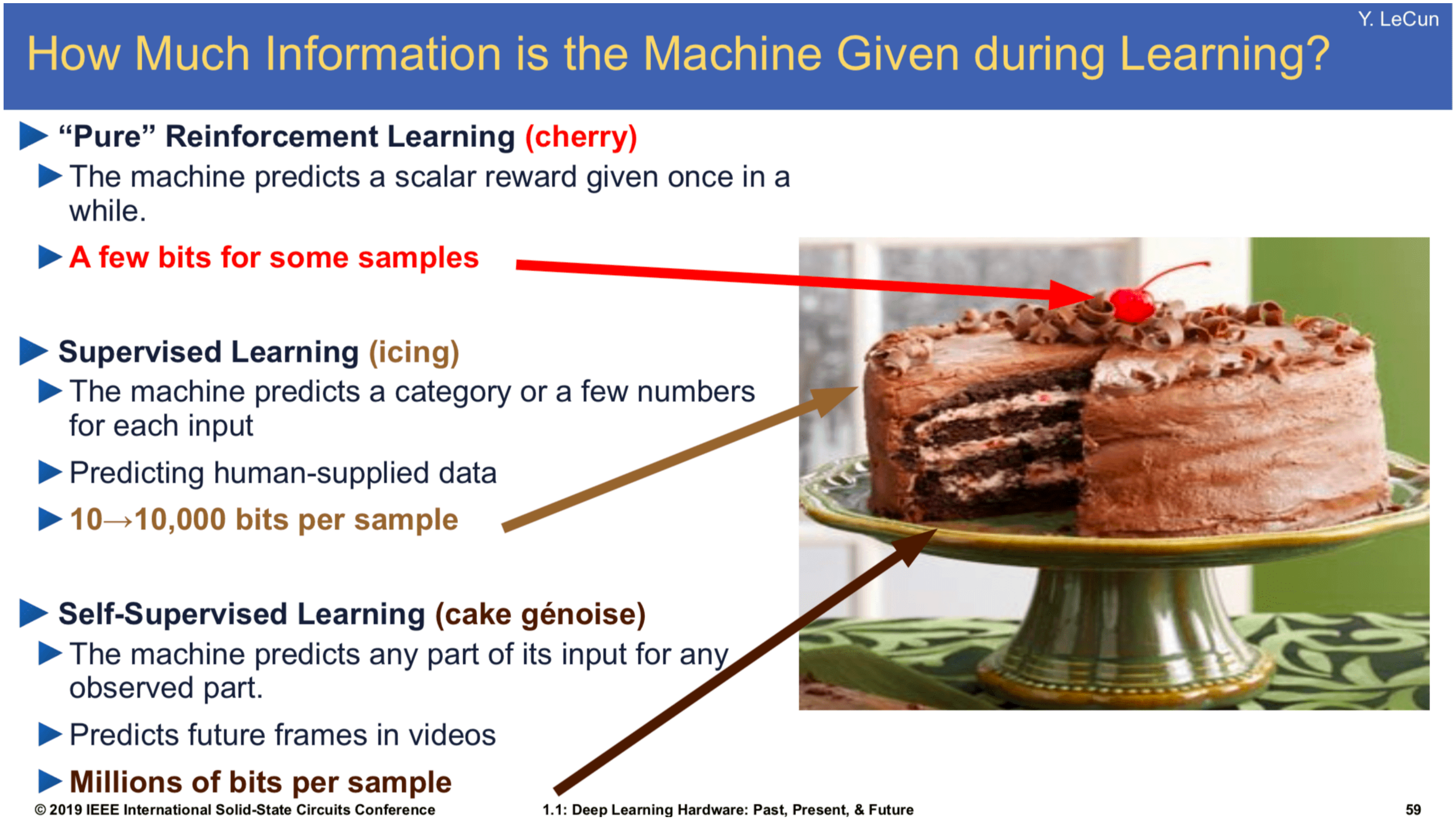
Alternatives to RL

- AI planning: given a known model of how action impacts world
  - e.g., searching through future game states

- Imitation learning
  - reduces RL to supervised learning from an expert
  - collect data, learn policy
  - failure modes exist due to cumulative nature of errors
  - combining imitation + RL is a promising direction

$$\begin{tikzcd}
\{(s_i, a_i)\} \arrow[d, Rightarrow] \\
a = \pi(s)
\end{tikzcd}$$

RL has very limited data to learn from



Y. LeCun

How Much Information is the Machine Given during Learning?

- **"Pure" Reinforcement Learning (cherry)**
  - The machine predicts a scalar reward given once in a while.
  - **A few bits for some samples**
- **Supervised Learning (icing)**
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - **10→10,000 bits per sample**
- **Self-Supervised Learning (cake génoise)**
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**

RL can be difficult to apply in practice

Reinforcement Learning for Real Life

ICML 2019 Workshop

June 14, 2019, Long Beach, CA, USA

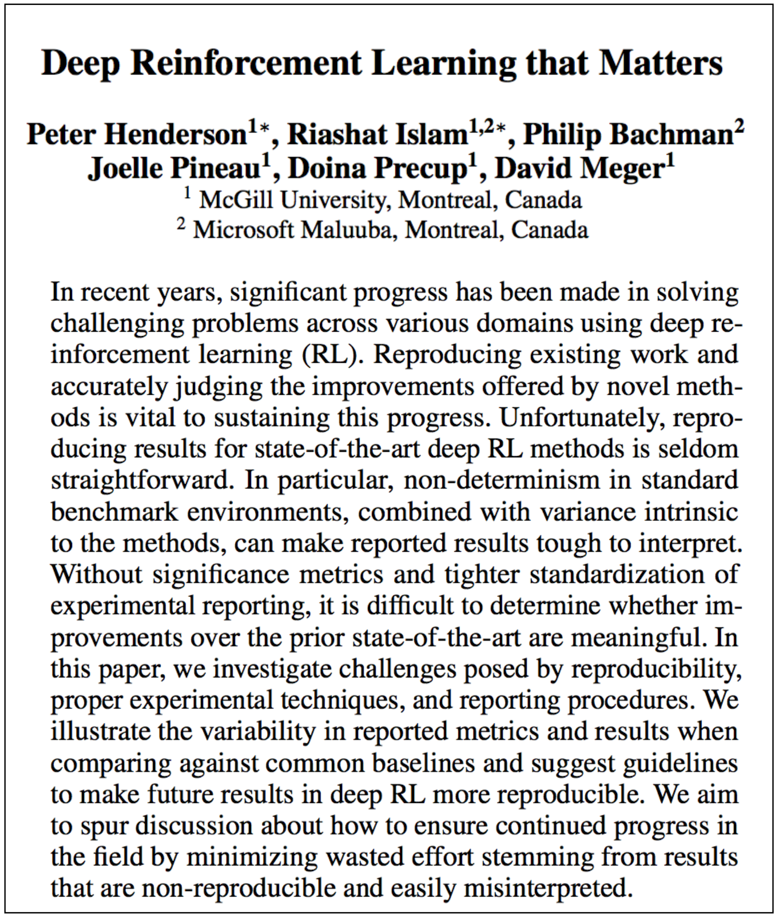
- Production systems
- Autonomous driving
- Business management
- Chemistry
- Computer Systems
- Energy
- Healthcare
- Robotics/manufacture

“Challenges of Real-World
Reinforcement Learning”
[Dulac-Arnold et al.]

“ … the research advances in RL are often hard to leverage in real-world systems due to a series of assumptions that are rarely satisfied in practice”

1. Training off-line from the fixed logs of an external behavior policy.
2. Learning on the real system from limited samples.
3. High-dimensional continuous state and action spaces.
4. Safety constraints that should never or at least rarely be violated.
5. Tasks that may be partially observable …
6. Reward functions that are unspecified, multi-objective, or risk-sensitive.
7. System operators who desire explainable policies and actions.
8. Inference that must happen in real-time at the control frequency of the system.
9. Large and/or unknown delays in the system actuators, sensors, or rewards.

Reprodicibility and Brittle Results



Deep Reinforcement Learning that Matters

Peter Henderson[1]*, Riashat Islam[1,2]*, Philip Bachman[2]
Joelle Pineau[1], Doina Precup[1], David Meger[1]
[1] McGill University, Montreal, Canada
[2] Microsoft Maluuba, Montreal, Canada

In recent years, significant progress has been made in solving challenging problems across various domains using deep reinforcement learning (RL). Reproducing existing work and accurately judging the improvements offered by novel methods is vital to sustaining this progress. Unfortunately, reproducing results for state-of-the-art deep RL methods is seldom straightforward. In particular, non-determinism in standard benchmark environments, combined with variance intrinsic to the methods, can make reported results tough to interpret. Without significance metrics and tighter standardization of experimental reporting, it is difficult to determine whether improvements over the prior state-of-the-art are meaningful. In this paper, we investigate challenges posed by reproducibility, proper experimental techniques, and reporting procedures. We illustrate the variability in reported metrics and results when comparing against common baselines and suggest guidelines to make future results in deep RL more reproducible. We aim to spur discussion about how to ensure continued progress in the field by minimizing wasted effort stemming from results that are non-reproducible and easily misinterpreted.

“Unfortunately, reproducing results for state-of-the-art deep RL methods is seldom straightforward.”