amii

UNIVERSITY OF ALBERTA

The Intelligent Robot Learning Laboratory

Professor Matthew E. Taylor (Matt)

Dec 8, 2020

Reinforcement Learning for Compilers and Chip Placement

Professor Matthew E. Taylor (Matt)
Dec 8, 2020

Reinforcement Learning for Compilers and Chip Placement
but Leleh, Nachiket, and Yu already mentioned this ☺

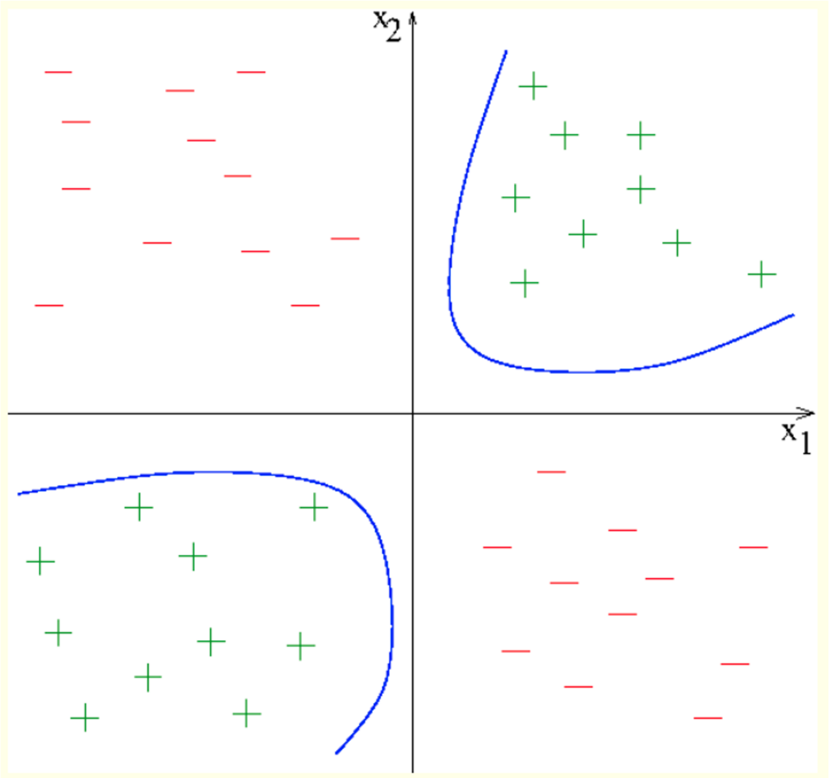
Outline

1. Background on Reinforcement Learning (RL)
2. Examples of RL
3. Next Steps

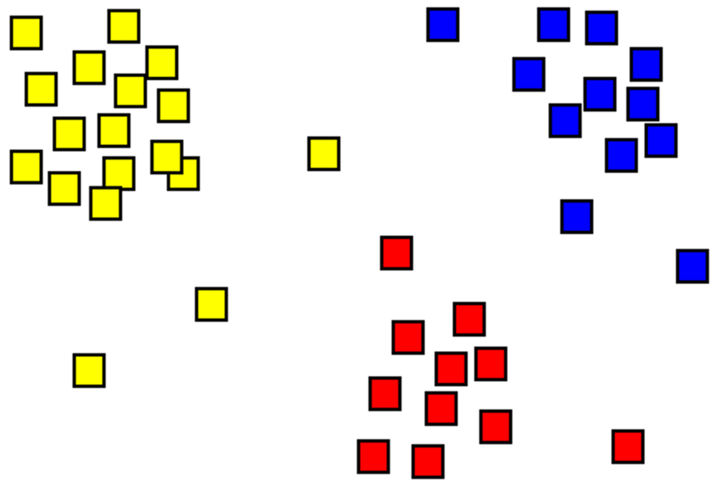
Please ask questions!
Text (voice/video)

Machine Learning (ML)

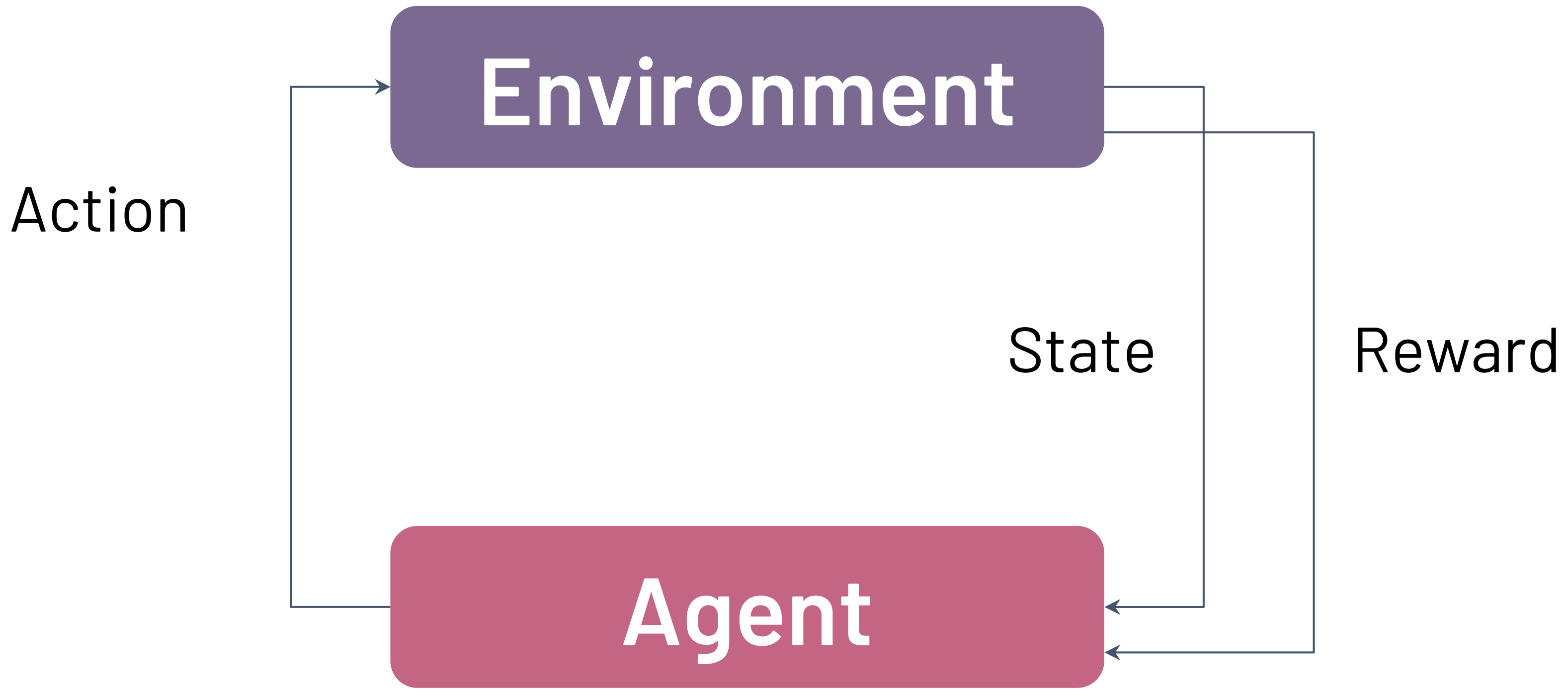
Supervised

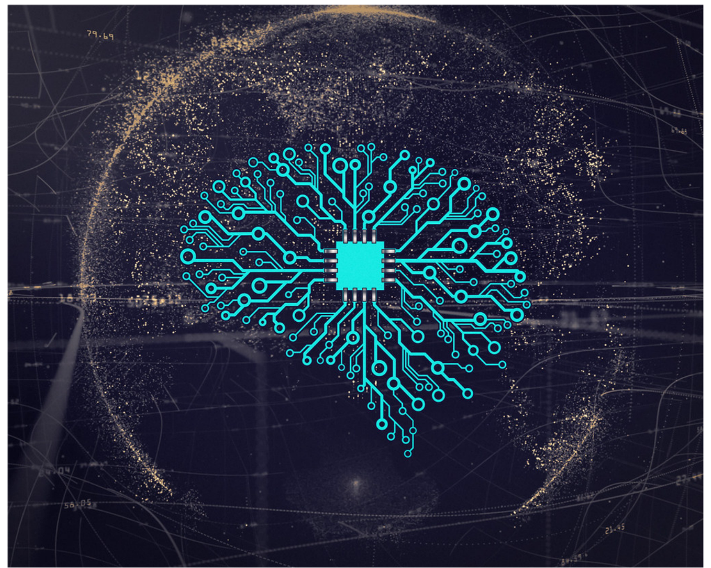


Unsupervised



Reinforcement Learning



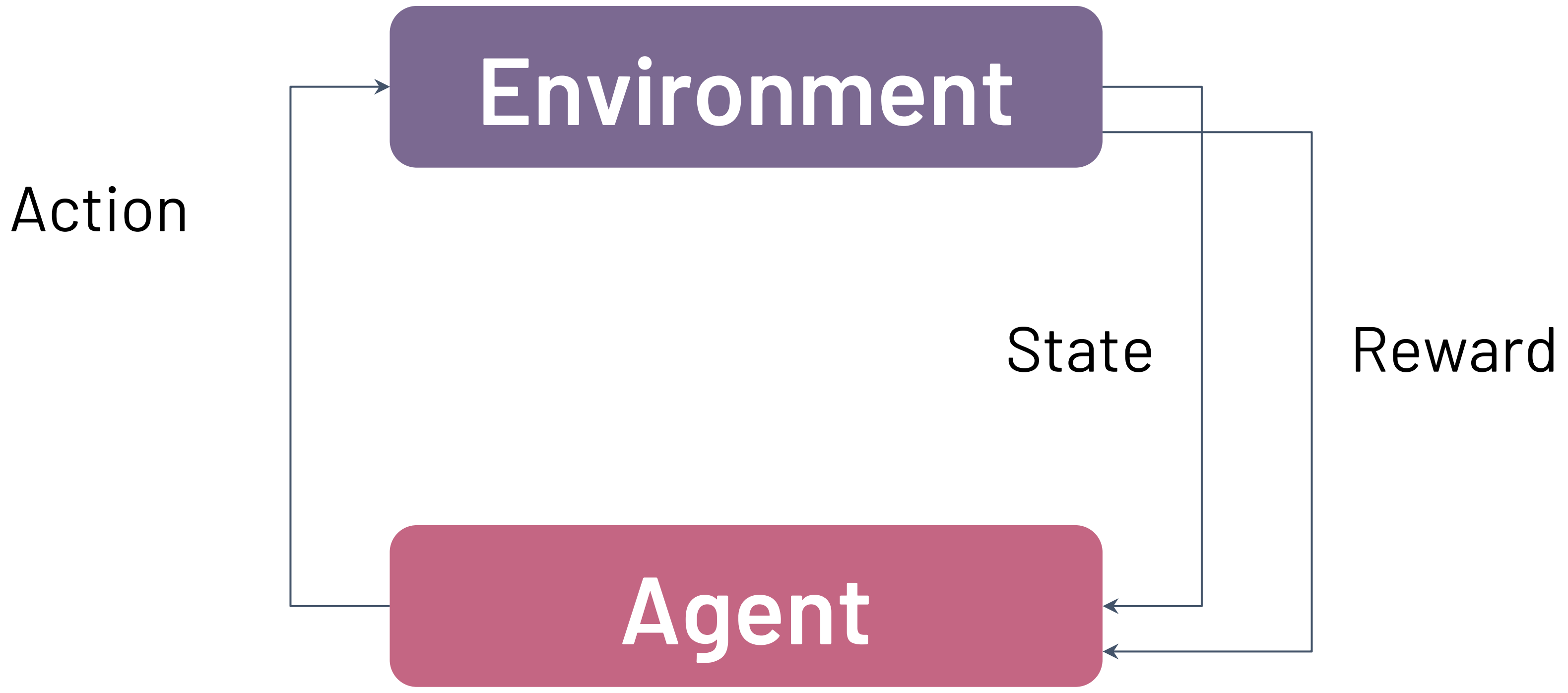


Reinforcement Learning (RL)

No labels: agent never told right or wrong

Agent interacts with environment (simulator or real world)

Typically can gather data, possibly at cost, by interacting with environment

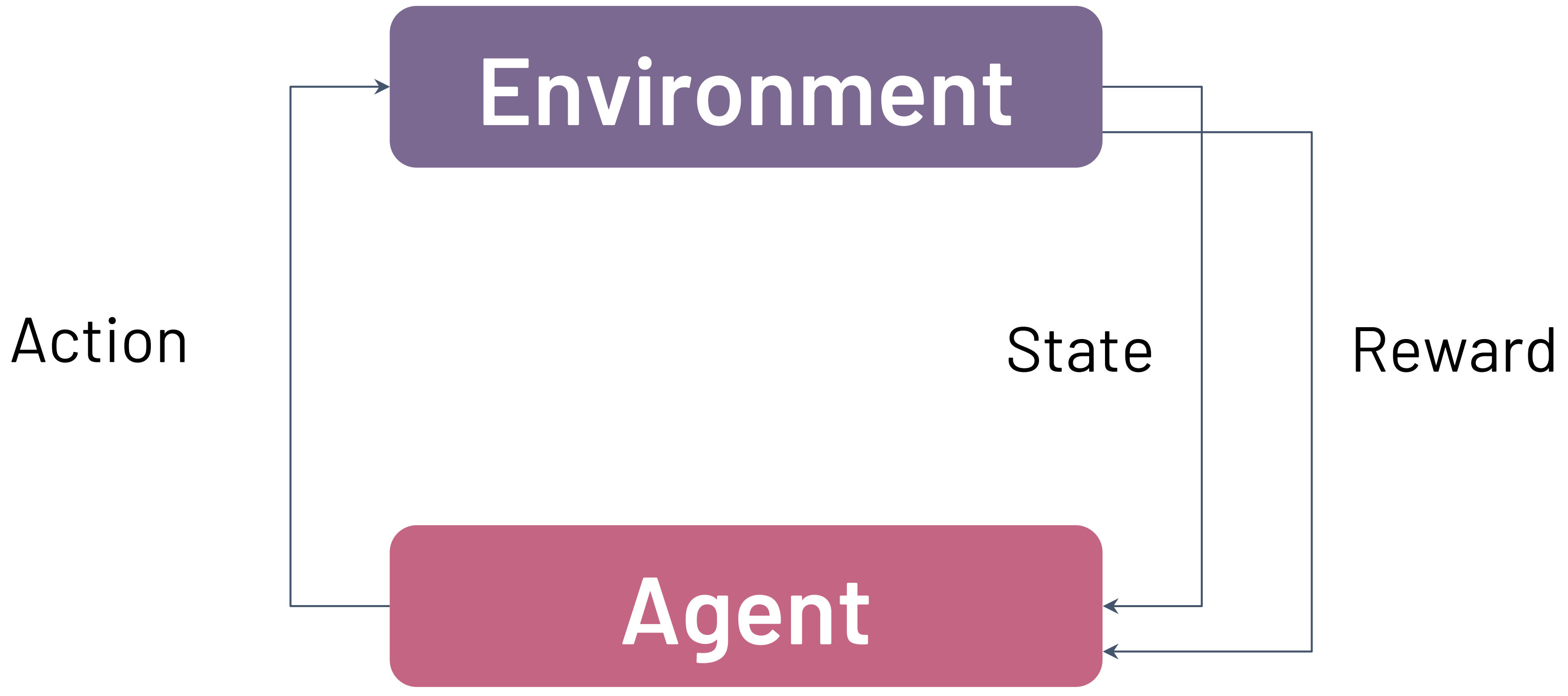


Reinforcement Learning (RL)

The agent typically learns via exploring vs. exploiting

Possible goals include

- Automation
- Improvement
- Enabling novel processes



RL Applications

(Un)Supervised learning performs well for many real-world applications

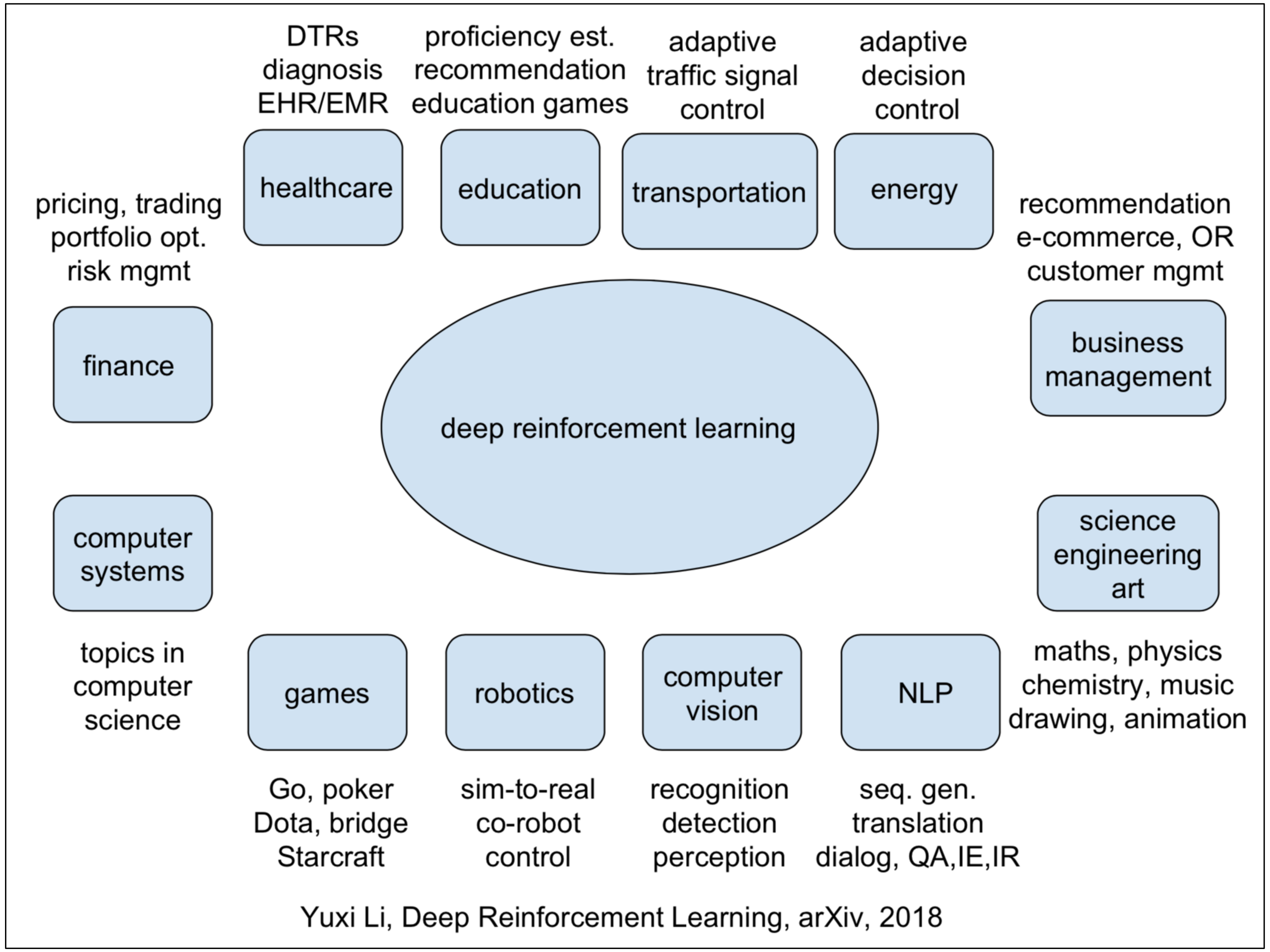
OpenAI Five - Dota

AlphaStar - StarCraft

AlphaGO

RL Applications

(Un)Supervised learning performs well for many real-world applications

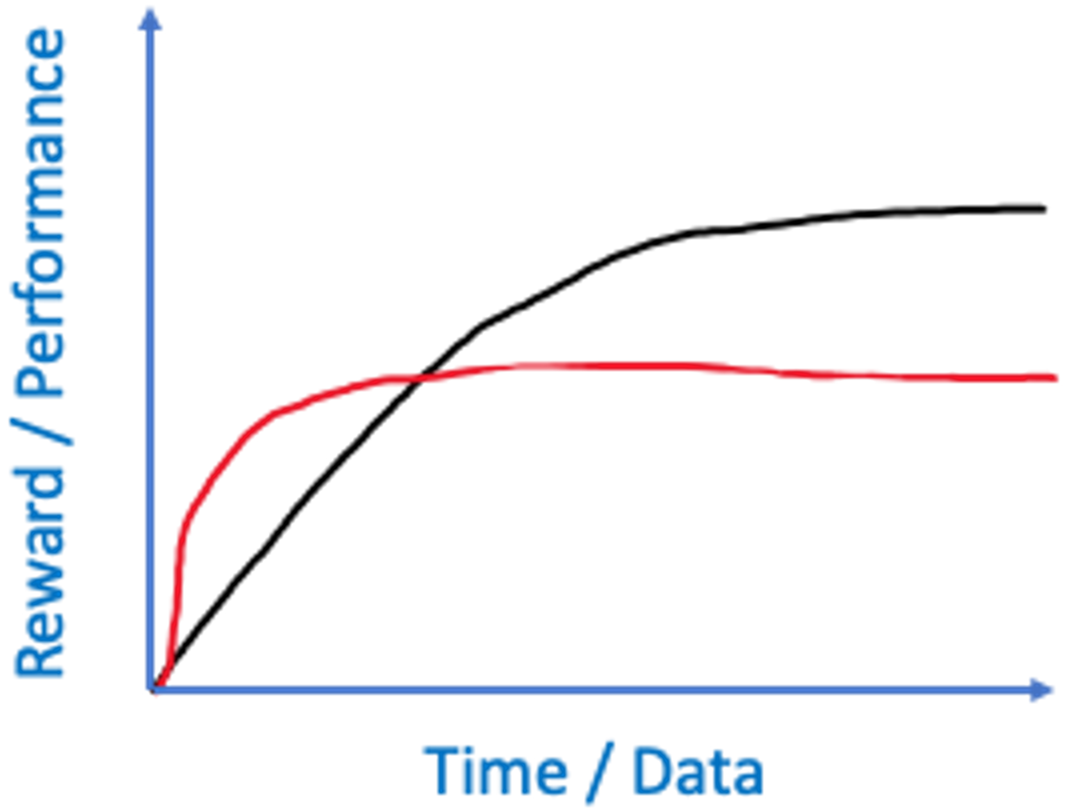


1. RL is mature
2. You should know if/where it applies

RL Goals

Learn to maximize real-valued reward signal (ideally)

- With maximal final performance
- With little data
- Reducing human effort
- Discovering novel solutions
- Handling non-stationary environments



RL: Setting

States: 12 in total

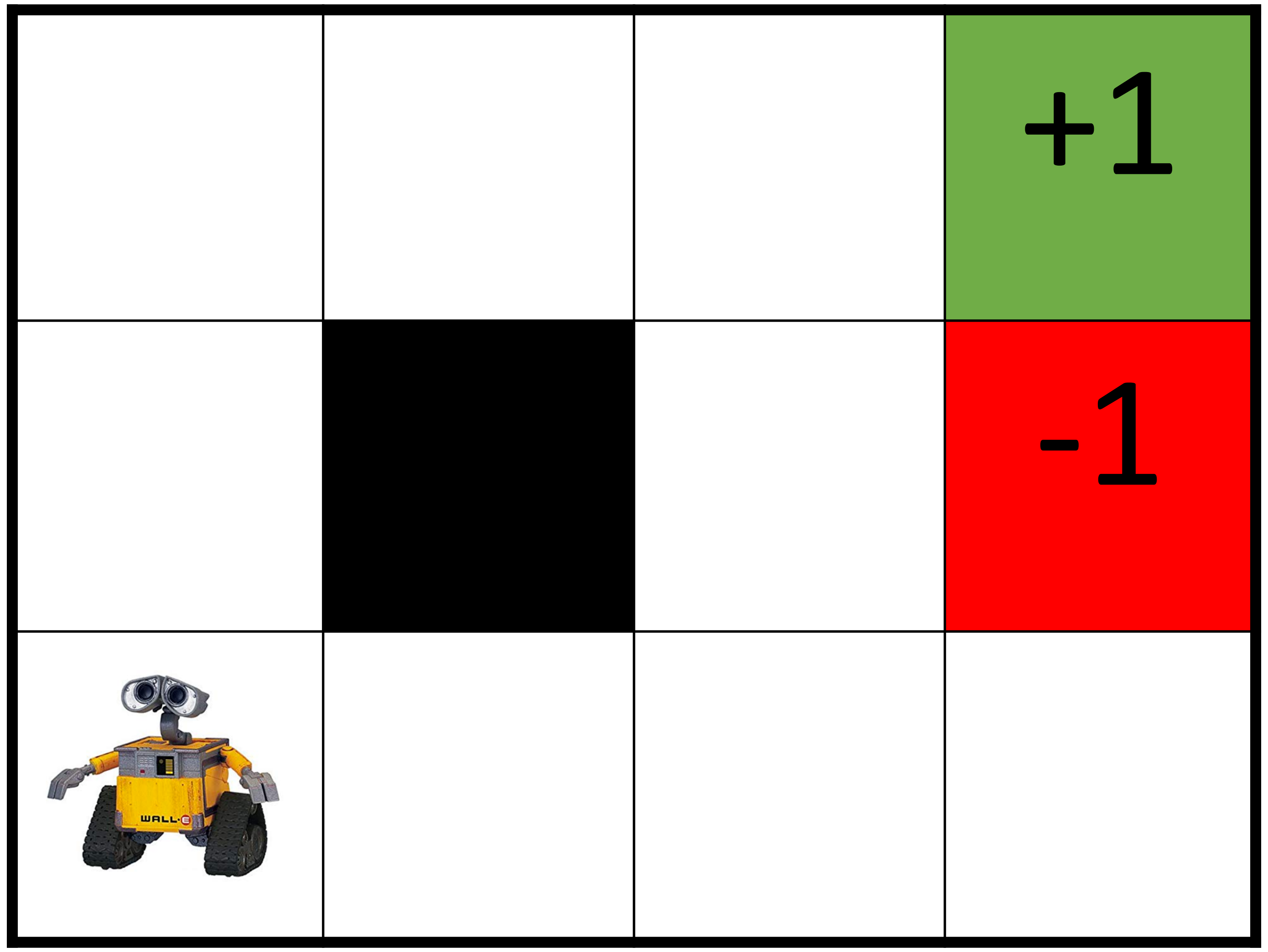
Actions: 4 cardinal directions

Transition Function: can't move through walls,

small chance of slipping to the side

Reward Function: 2 end states + step penalty

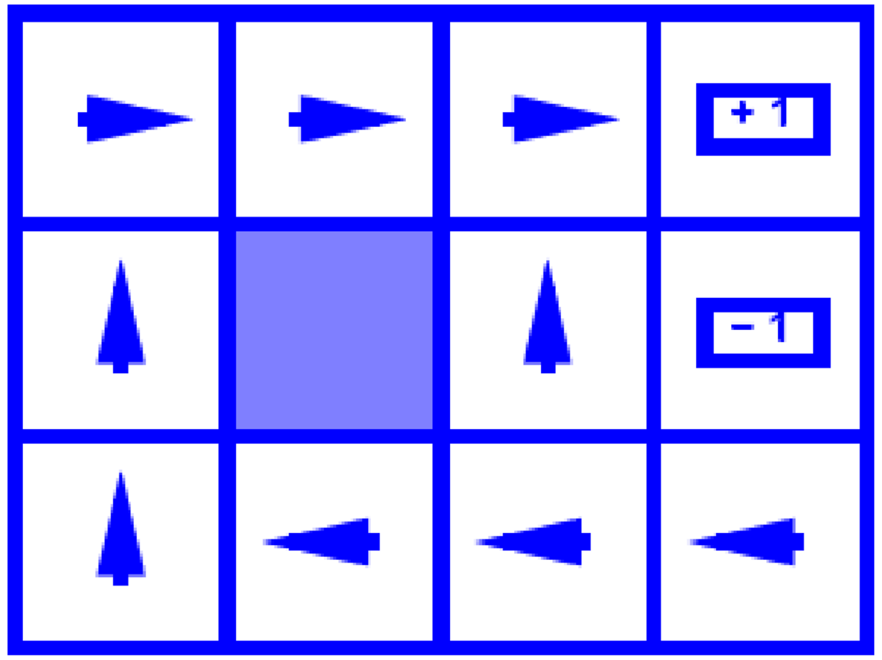
Policy: How to act (states → actions)



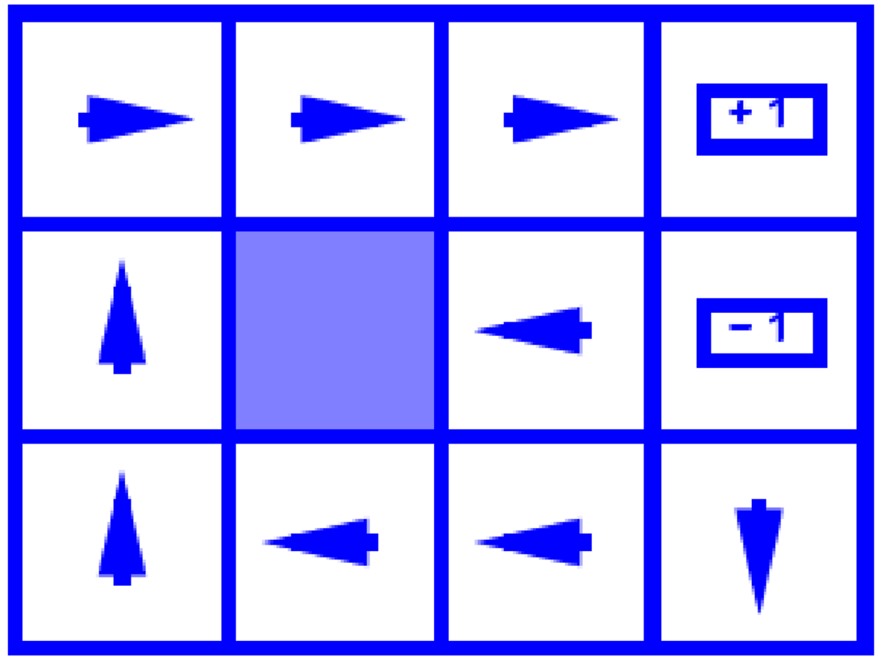
RL: Optimal Policies

Can solve via planning

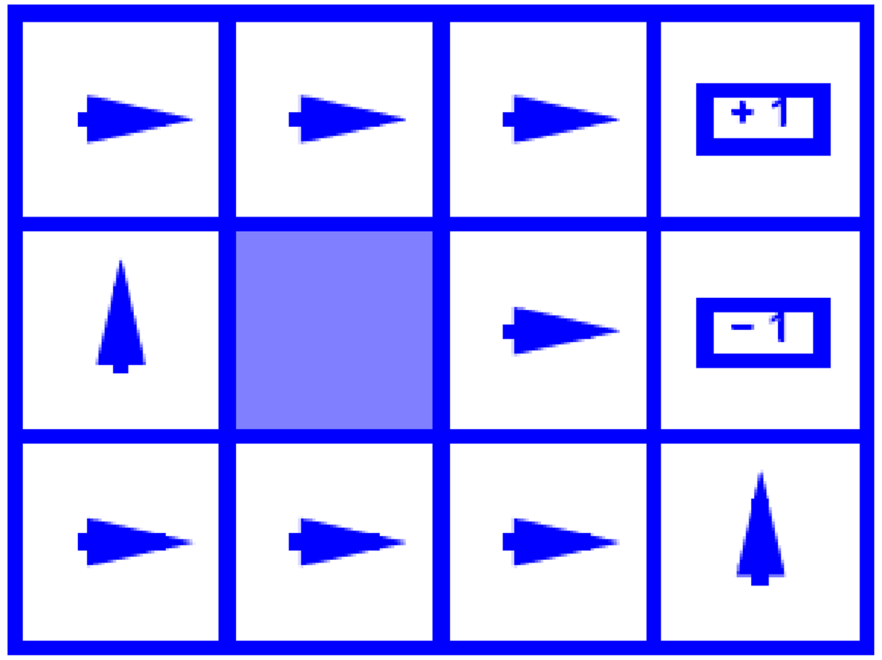
Reward function determines behavior



R(s) = -0.03



R(s) = -0.01

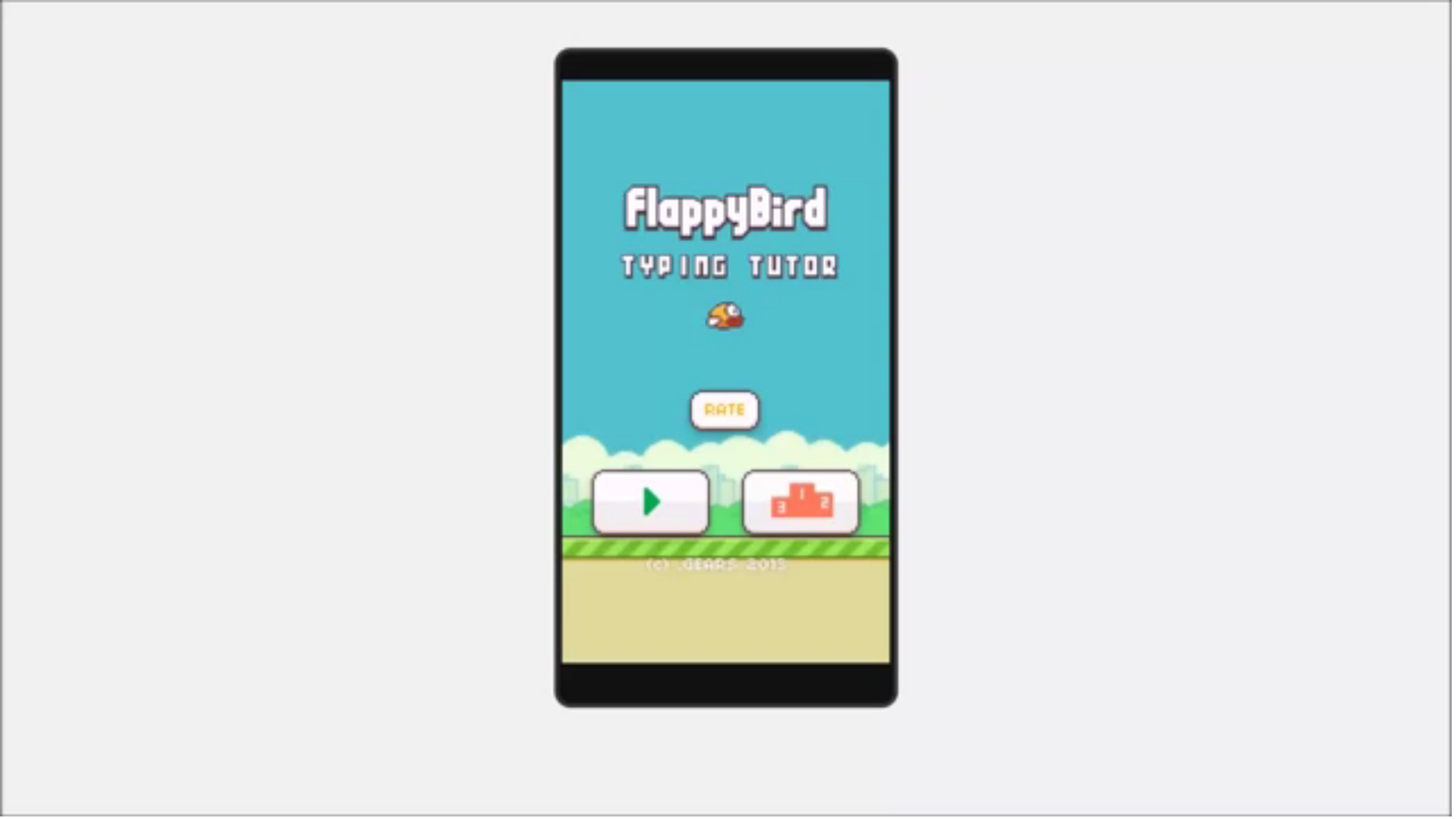


R(s) = -2.0

Outline

1. Background on RL
2. Examples of RL
3. Next Steps

https://www.flickr.com/photos/



- 2-3 days of development
- 2013 release, 2014 reportedly making $50k/day
- Then, removed because "too addictive"

https://www.youtube.com/watch?v=OJw4HTWvGdY http://sarvagyavaish.github.io/FlappyBirdRL/

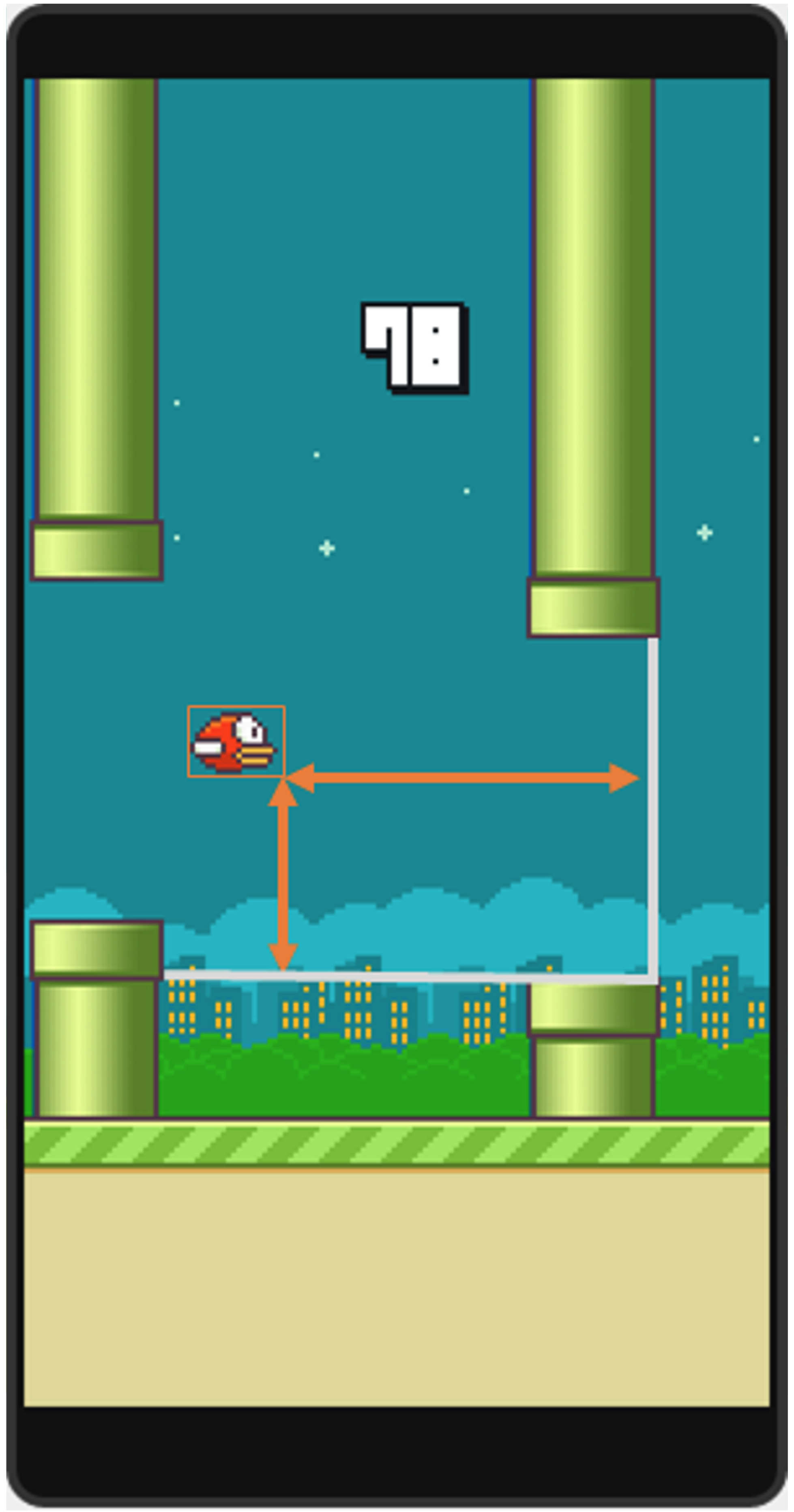
Example 1: Flappy Bird

Transition function: controlled by game

Action?

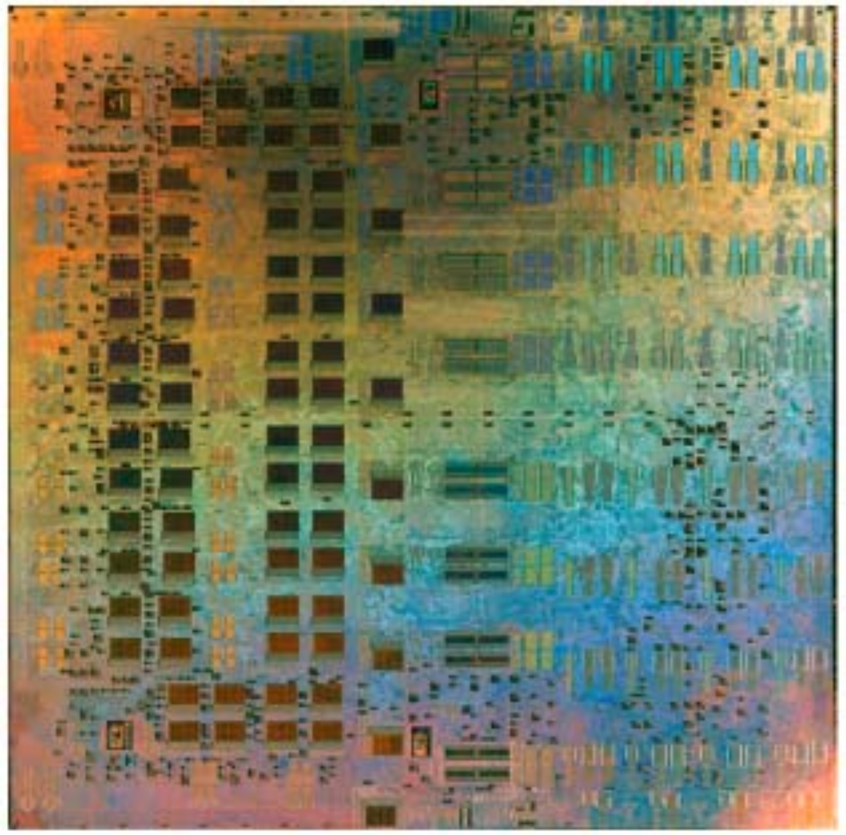
Reward?

State representation?



Example 2: Compiler Optimization

PACT-08



TRIPS: Tera-op, Reliable, Intelligently adaptive Processing System

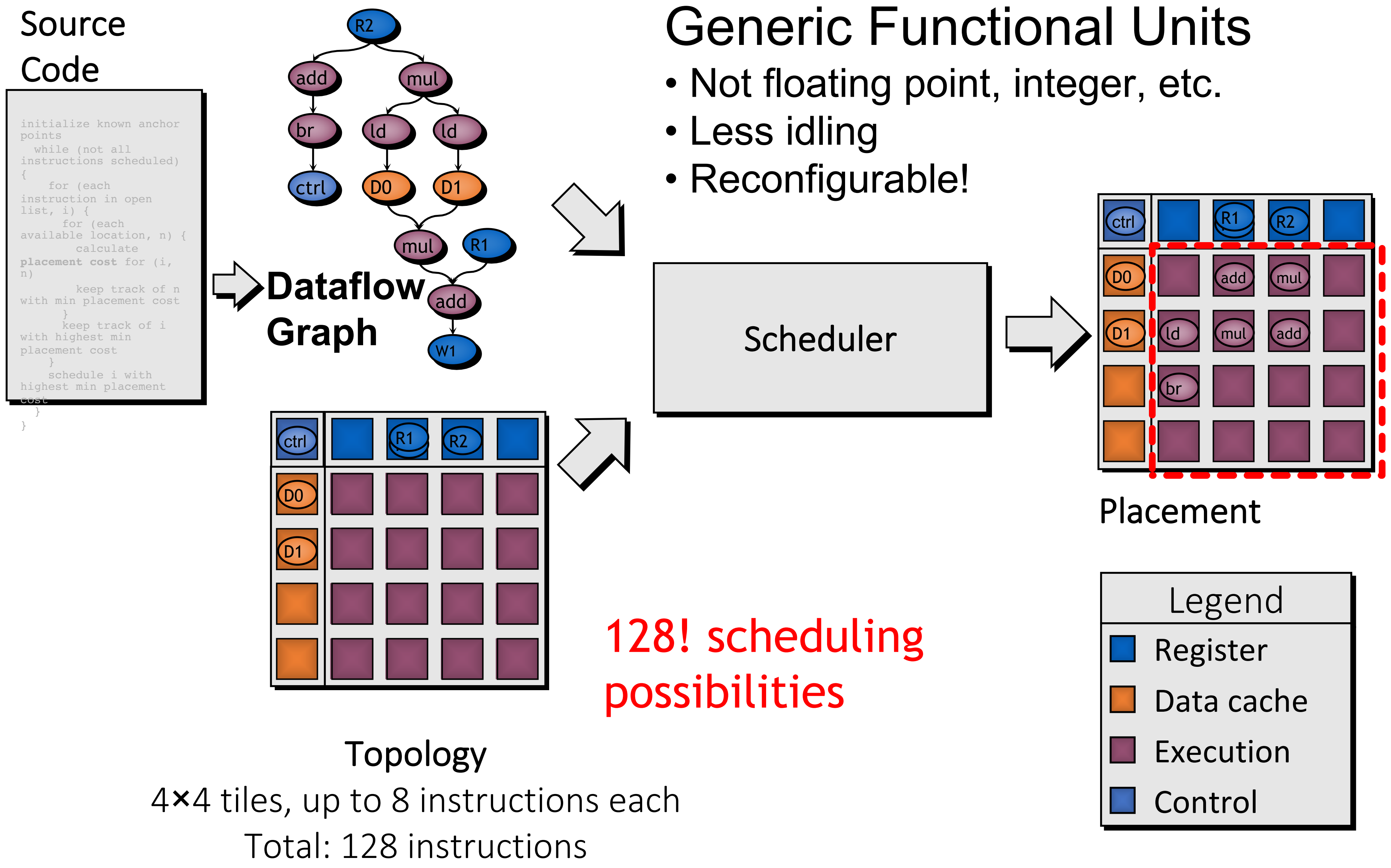
SPS scheduler: 2006

UT-Austin:
Kathryn S. McKinley &
Doug Berger

Scheduling Overview

Generic Functional Units

- Not floating point, integer, etc.
- Less idling
- Reconfigurable!

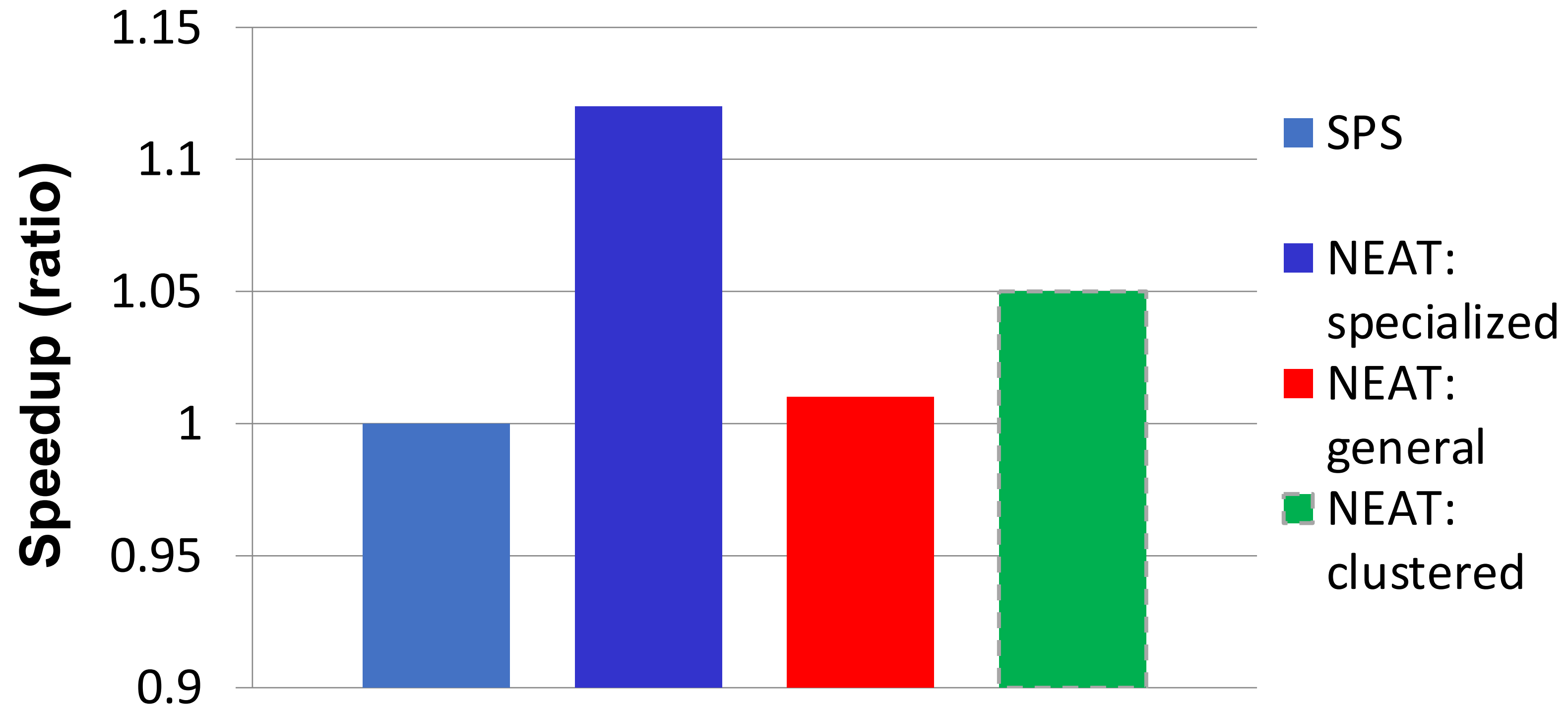


Example 2: Compiler Optimization

PACT-08

State: 11 features based on current instruction & already placed
Action: Place an instruction
Reward: 0 until all instructions placed, then, what's the speedup?

Heuristics → Learned scheduler heuristics
Per benchmark or general
47 small benchmarks

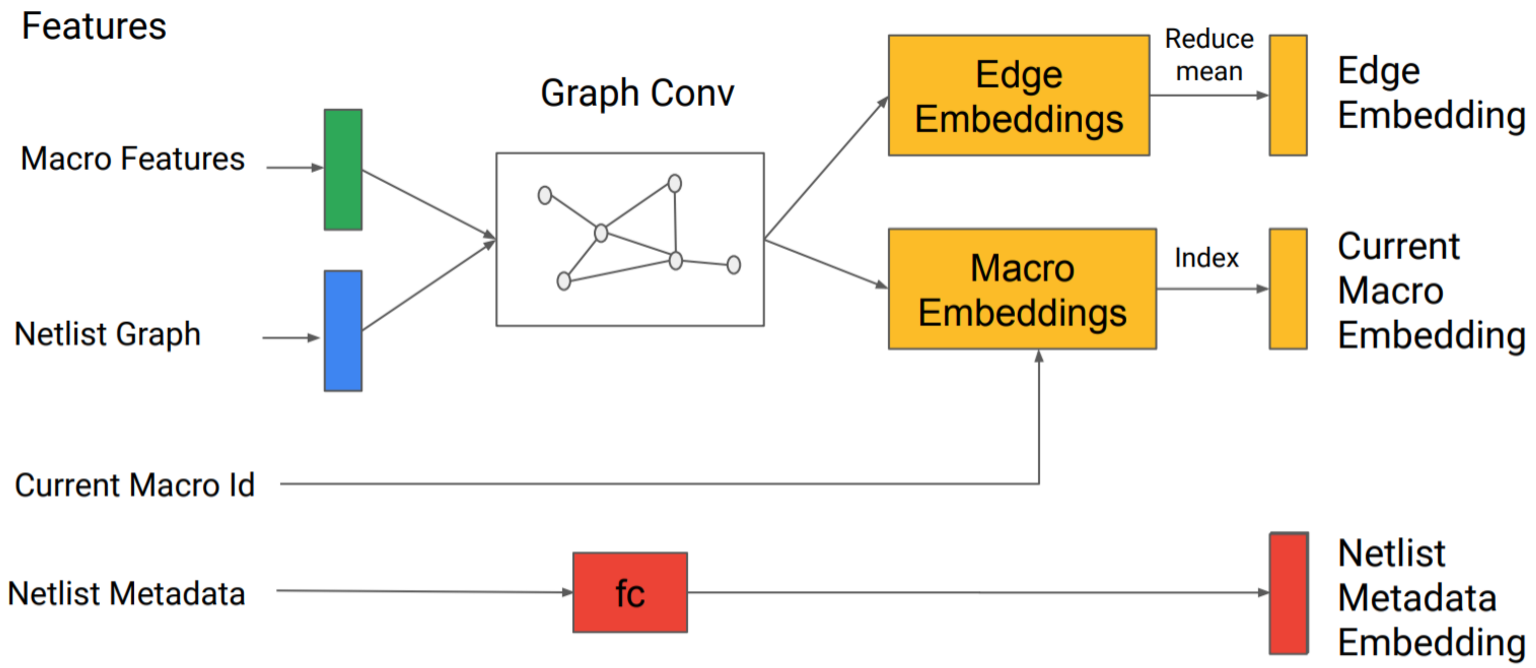


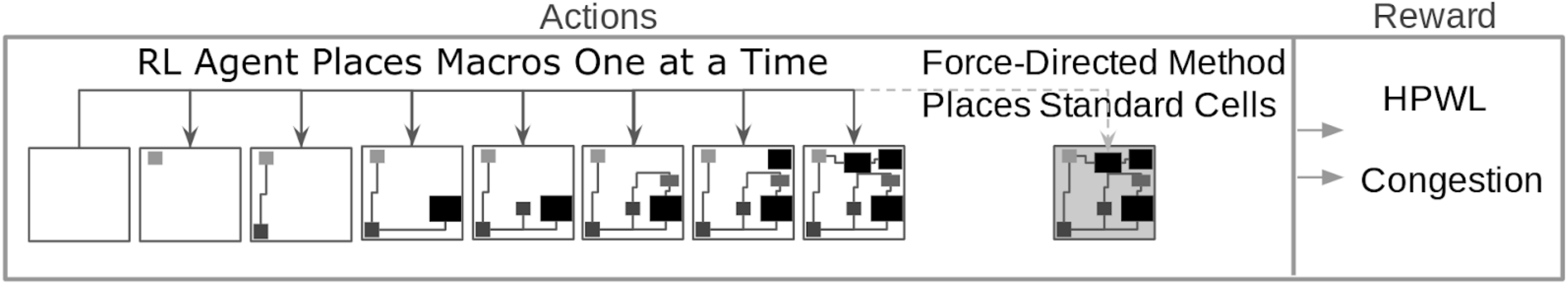
Example 3: Google's Chip Placement with Deep RL

Azalia+, 2020: arXiv:2004.10746v1

TPUs & ASIC

Netlist: graph of circuit components





Example 3: Google's Chip Placement with Deep RL

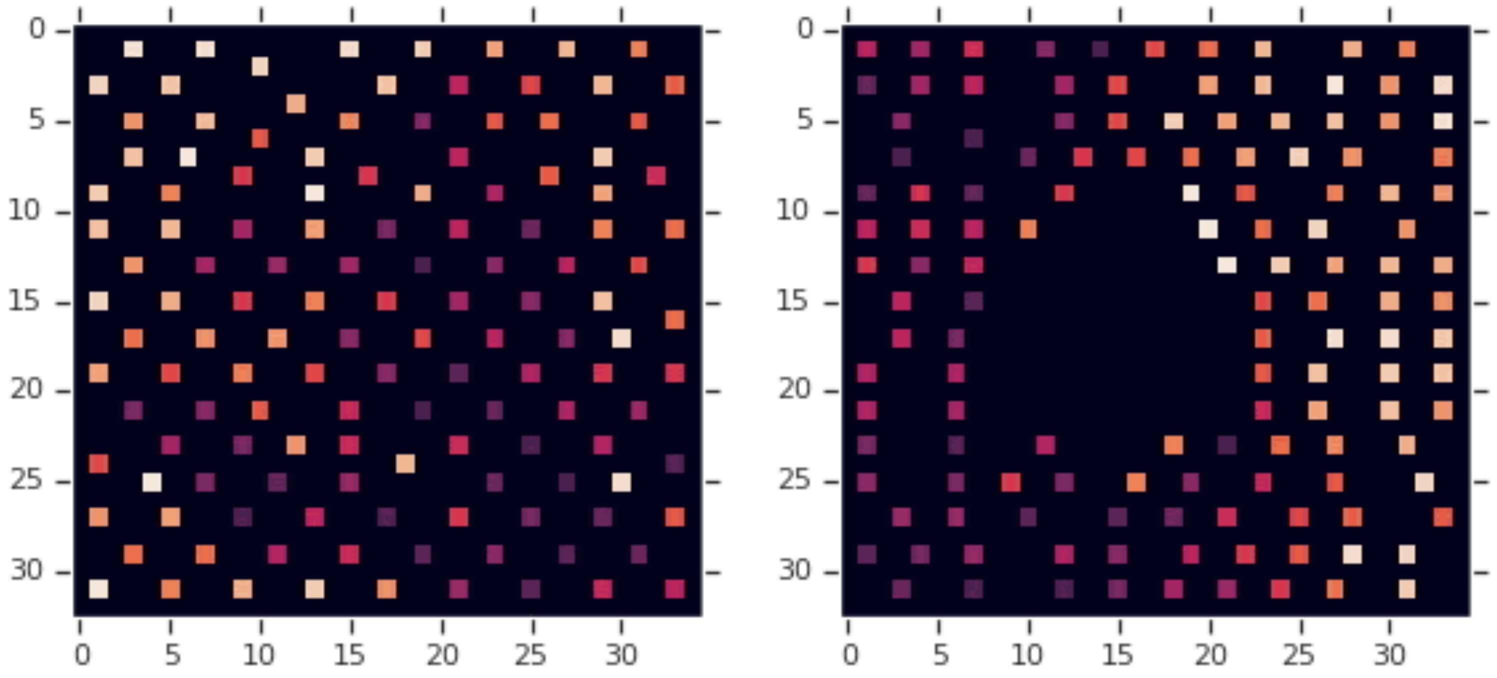
States: Every possible partial placement of netlist onto chip canvas

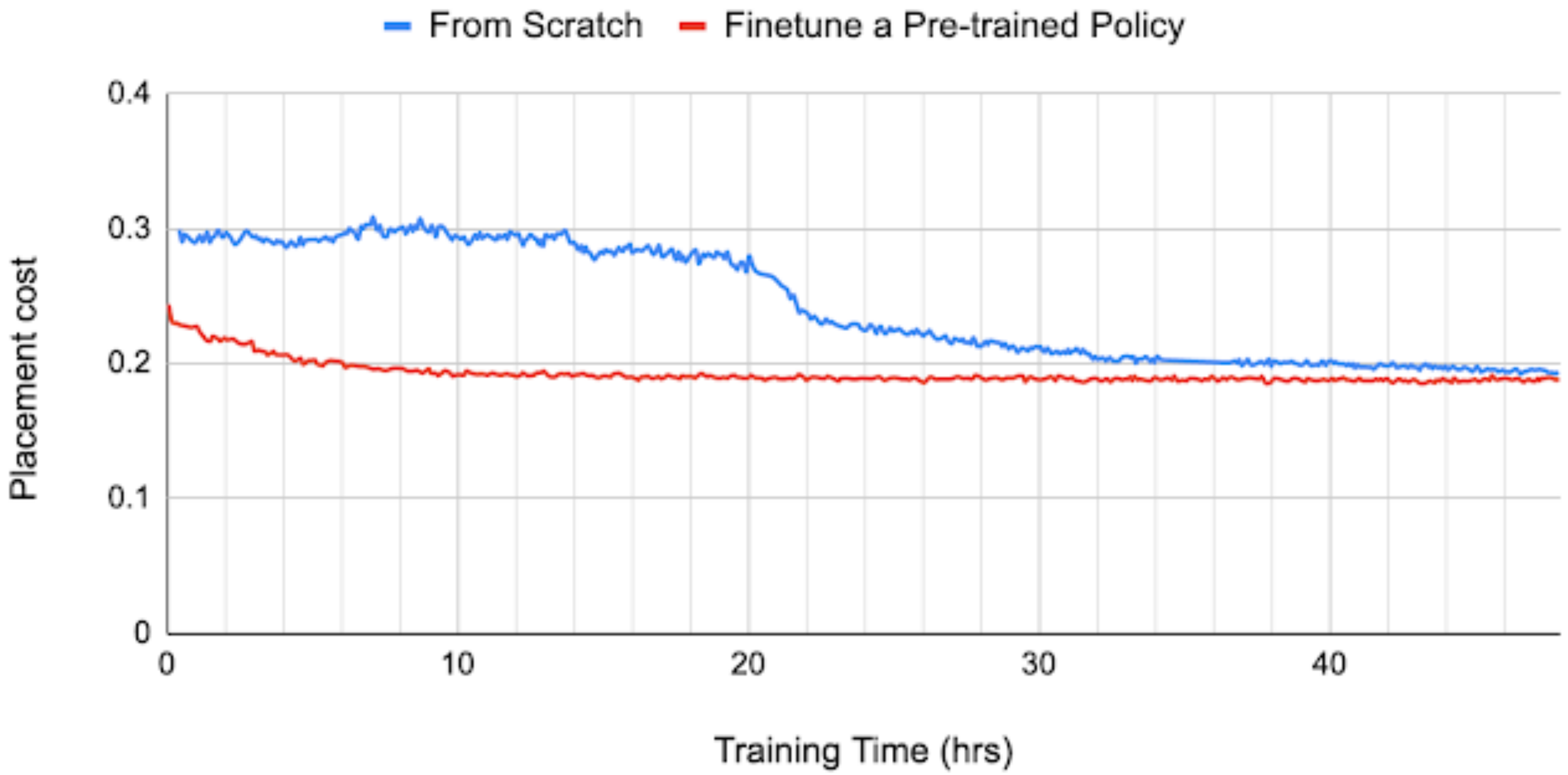
Actions: Place current macro at any location on discrete canvas space

- Don't violate hard constraints

Reward: 0 for all actions except last action

- Negative weighted sum of proxy wirelength & congestion





Example 4: Water Treatment

Developed @ UofA

ISL Adapt, UofA, and Amii

No ground truth
Raw water from North Saskatchewan River
State: Sensors added to filtration plant
Actions: Changes like chemicals, backwash cleaning, etc.
Reward: Environmental and fiscal benefits

bit.ly/3ouscLO

Water Treatment: Drayton Valley

Outline

1. Background on RL
2. Examples of RL
3. Next Steps

RL Strengths

Agent can autonomously learn to maximize rewards

Programmer just specifies goals

Often much less work than directly programming

Can achieve superhuman performance

Can handle unanticipated changes in the environment

RL Weaknesses

Agent maximizes rewards whether it's what you actually wanted or not!

- Example: agent collects points in a game, rather than completing level

Can require lots of computation and/or interaction with the real world

- Interacting with world can have cost in time, money, wear, etc.

Solutions are often black box: explainability is not well understood (yet)

Initial performance could be very poor

Additional Resources

RL

- Coursera RL specialization from U Alberta (White & White)
  - https://www.coursera.org/specializations/reinforcement-learning
- Udacity class from Georgia Tech
  - https://www.udacity.com/course/reinforcement-learning--ud600
- THE book on RL (Sutton & Barto, 2018)
  - http://www.incompleteideas.net/book/the-book-2nd.html
- Csaba Szepesvári: Algorithms in Reinforcement Learning
  - https://sites.ualberta.ca/~szepesva/rlbook.html

Deep RL

- Class on YouTube from UCL/Deepmind
  - https://www.youtube.com/playlist?list=PLqYmG7hTraZDNJre23vqCGIVpfZ_K2RZs
- OpenAI Spinning Up in Deep RL
  - https://spinningup.openai.com/en/latest/

Thank you! Questions?

Matt Taylor:
IRLL.ca
Amii.ca

RL
- Coursera RL specialization from U Alberta
  - https://www.coursera.org/specializations/reinforcement-learning
- Udacity class from Georgia Tech
  - https://www.udacity.com/course/reinforcement-learning--ud600
- THE book on RL (Sutton & Barto, 2018)
  - http://www.incompleteideas.net/book/the-book-2nd.html
- Csaba Szepesvári: Algorithms in Reinforcement Learning
  - https://sites.ualberta.ca/~szepesva/rlbook.html

Deep RL
- Class on YouTube from UCL/Deepmind
  - https://www.youtube.com/playlist?list=PLqYmG7hTraZDNJre23vqCGIVpfZ_K2RZs
- OpenAI Spinning Up in Deep RL
  - https://spinningup.openai.com/en/latest/