HIC 2018

13TH INTERNATIONAL HYDROINFORMATICS CONFERENCE HIC 2018

1-6 JULY 2018 | PALERMO | ITALY

Autonomous Control of Urban Storm Water Networks Using Reinforcement Learning

Abhiram Mullapudi, Branko Kerkez

Department of Civil and Environmental Engineering
University of Michigan, Ann Arbor

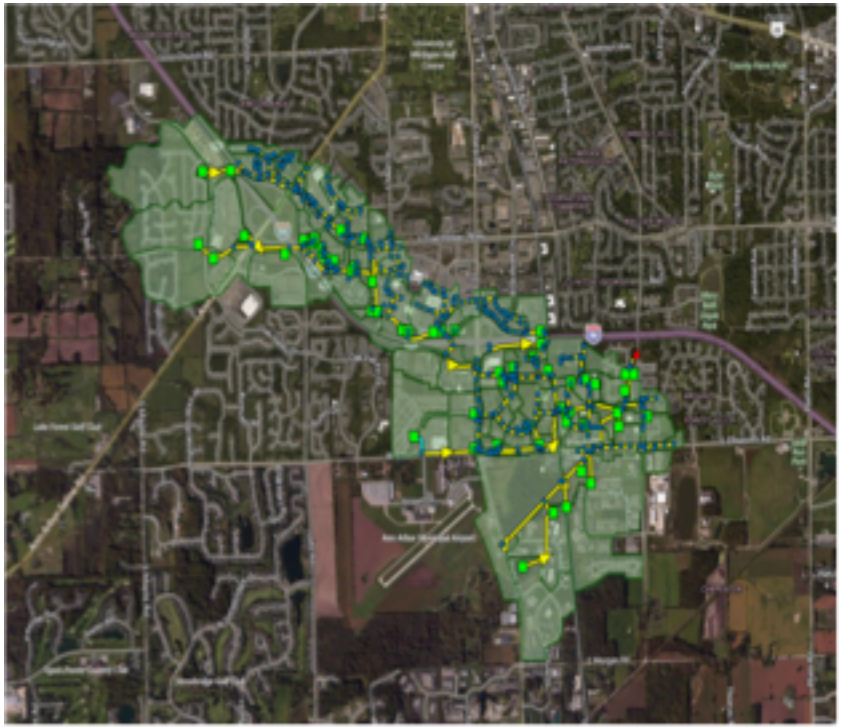
Storm water

May 27 2018
Ellicott City, Md, USA

Lake Erie
USA

nytimes.com/2018/05/27/us/ellicott-city-flooding-maryland.htm
oceanservice.noaa.gov/facts/hab-forecast.html

Storm water networks



Ann Arbor Storm Water Network

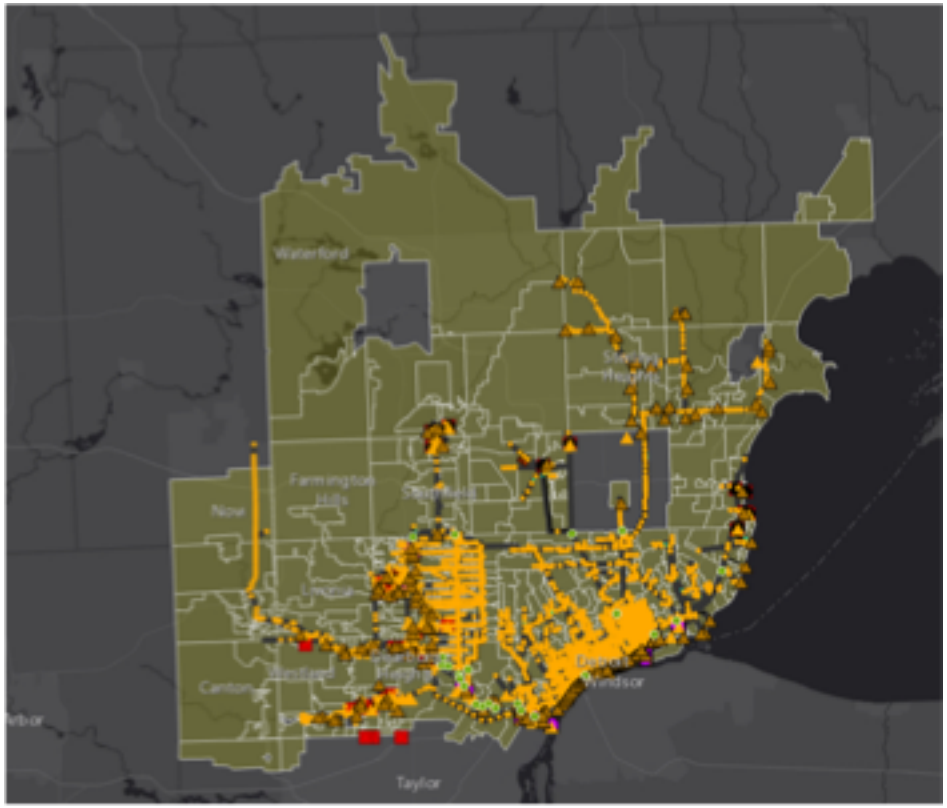
Storm Water Drain,
Chicago, USA

Smarter storm water systems

Kerkez et.al. 2016

Smarter storm water systems

System scale control



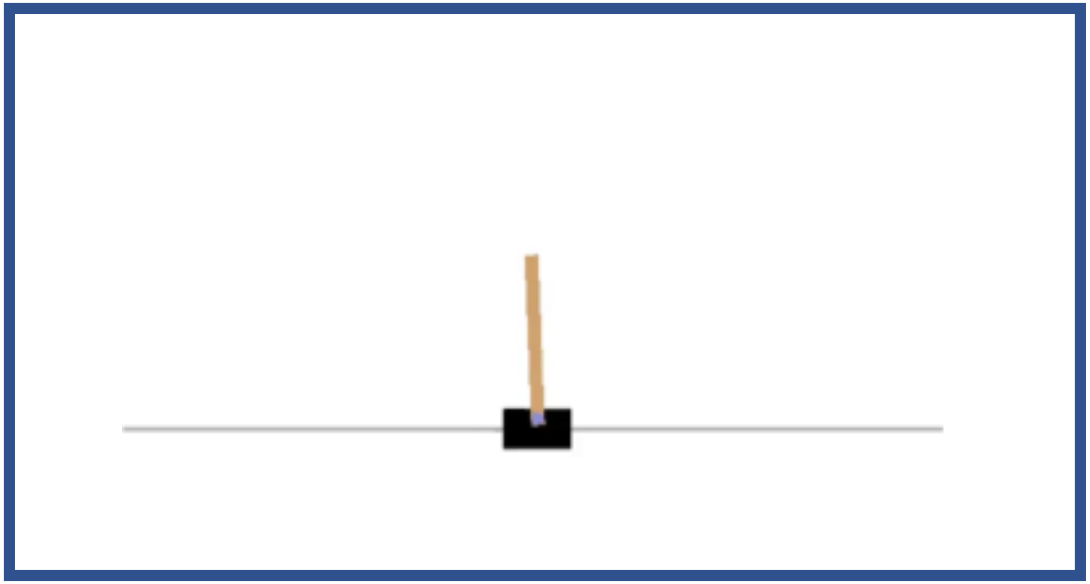
Detroit Storm Water Network

- Coordinate responses between spatially distributed assets
- Account for sensor failure, flash flooding etc.
- Uncertainty
- Real time decisions
- Multi objective formulations

Reinforcement Learning

nature.com
alphagomovie.com/

Reinforcement Learning



Actions:

1. Push cart to right
2. Push cart to left

Observation:

1. Cart position
2. Cart velocity
3. Pole angle
4. Pole velocity at tip

gym.openai.com/envs/CartPole-v0/

Cartpole

Observation:

1. Cart position
2. Cart velocity
3. Pole angle
4. Pole velocity at tip

$\}$ $state(s)$

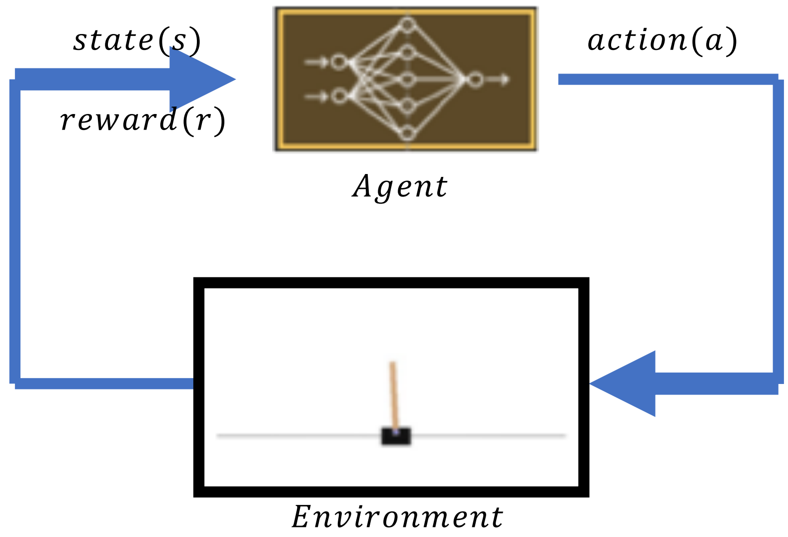
Actions:

1. Push cart to right
2. Push cart to left

$\}$ $action(a)$

Reward:

$$\left.\begin{array}{c} +1\ if\ pole\ (0^0 \pm 12^o) \\ 0\ else \end{array}\right\} reward(r)$$

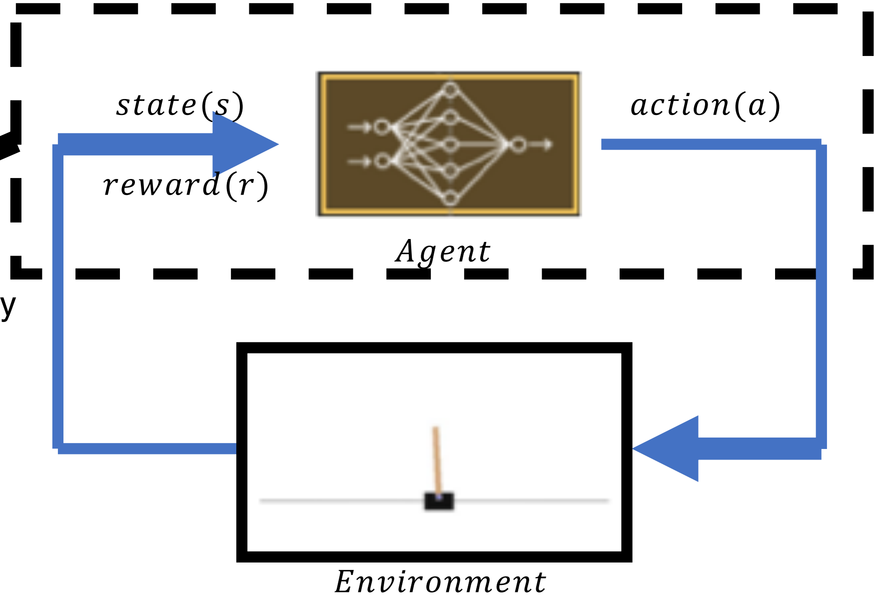


$$\boxed{f: s \rightarrow a}$$

Cartpole – deep neural network

$$NN(state)[\mathbb{R}^m] \rightarrow q(s,a)[\mathbb{R}^n]$$

$$action = \max_a q(s,a)$$



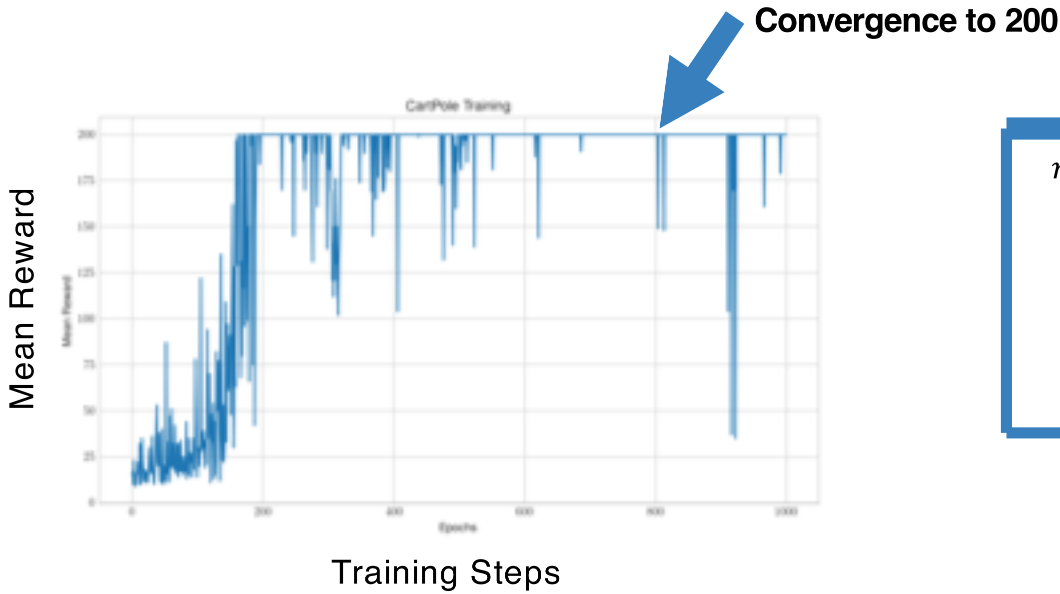
1. Predict $action(a)$ based on the $state(s)$
2. Store the reward, action, state and next state to memory
3. After T steps, start learning

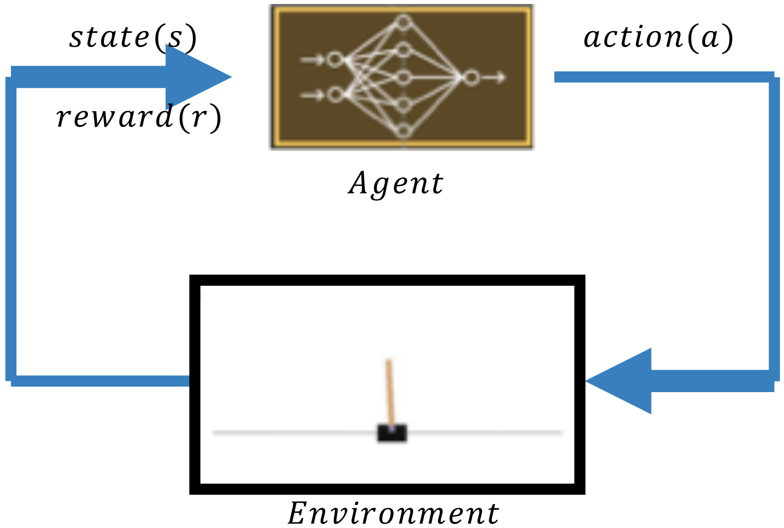
- Sample from memory (random)
- $Loss = |q(s,a) - (r + \gamma \max q(s_{t+1}, a_{t+1})|^2$
- Backpropagate the loss to update NN

$$q(s,a) \rightarrow Value\ of\ taking\ action\ a\ in\ state\ s$$

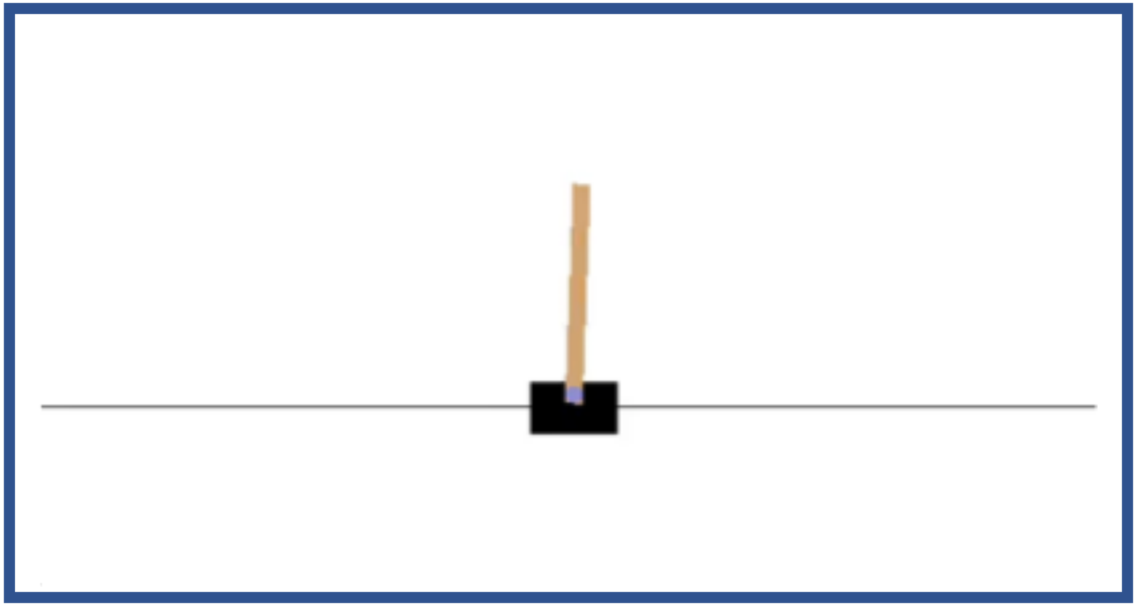
$$q(s_t,a_t) = \mathrm{E}_\pi\ [r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots |s,a] = E_\pi\left[\sum_{i=0}^{\infty} \gamma^i r_{t+i}\,|s_t,a_t\right]$$

Cartpole

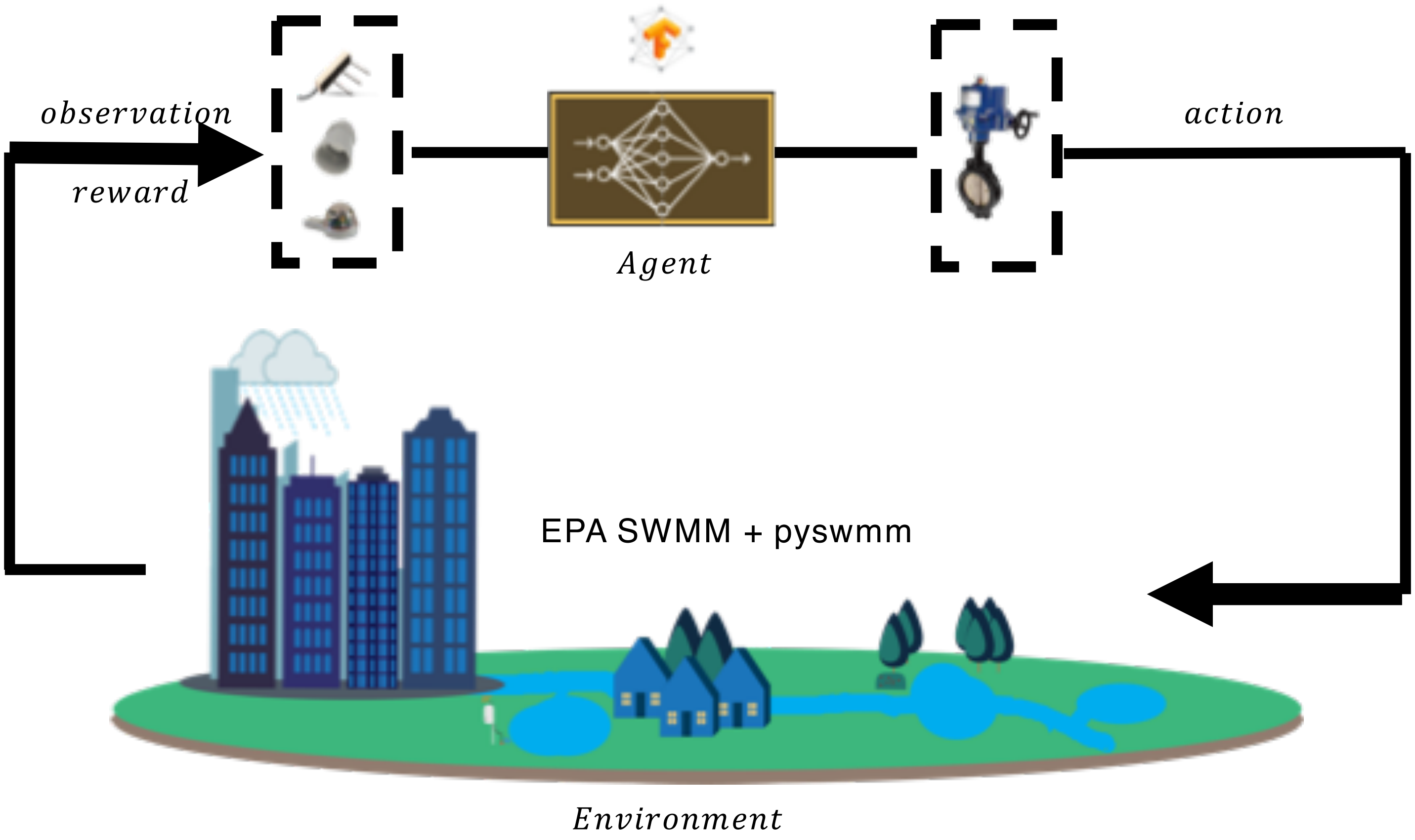

Convergence to 200
Mean Reward
Training Steps


$state(s)$
$reward(r)$
$action(a)$
$Agent$
$Environment$

Cartpole - solved

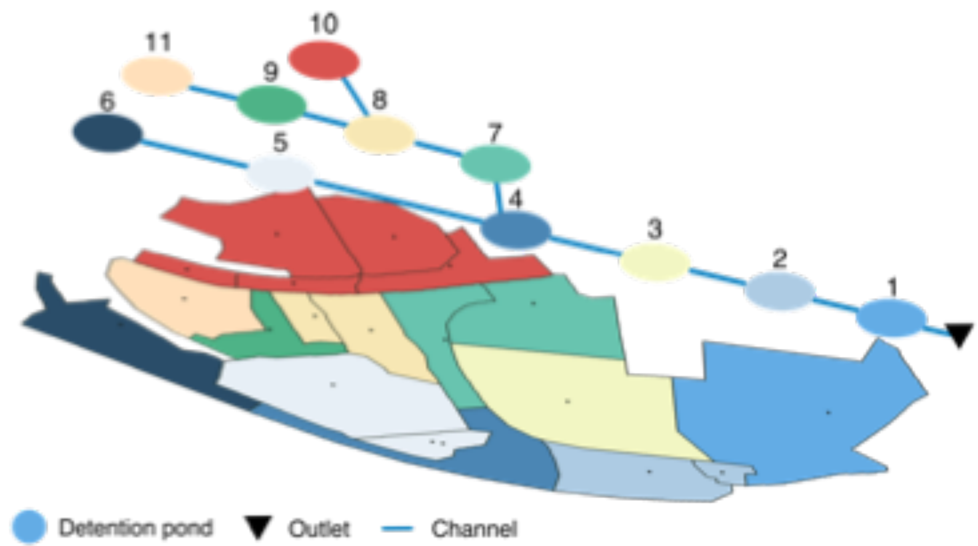


RL – storm water control



Characterizing RL

- Convergence - reward function (Single Pond)
  - 3 different reward functions of same objective
- Deep Neural Networks(System Scale Control)
  1. Generic Neural Networks
  2. Generic Neural Networks with batch normalization

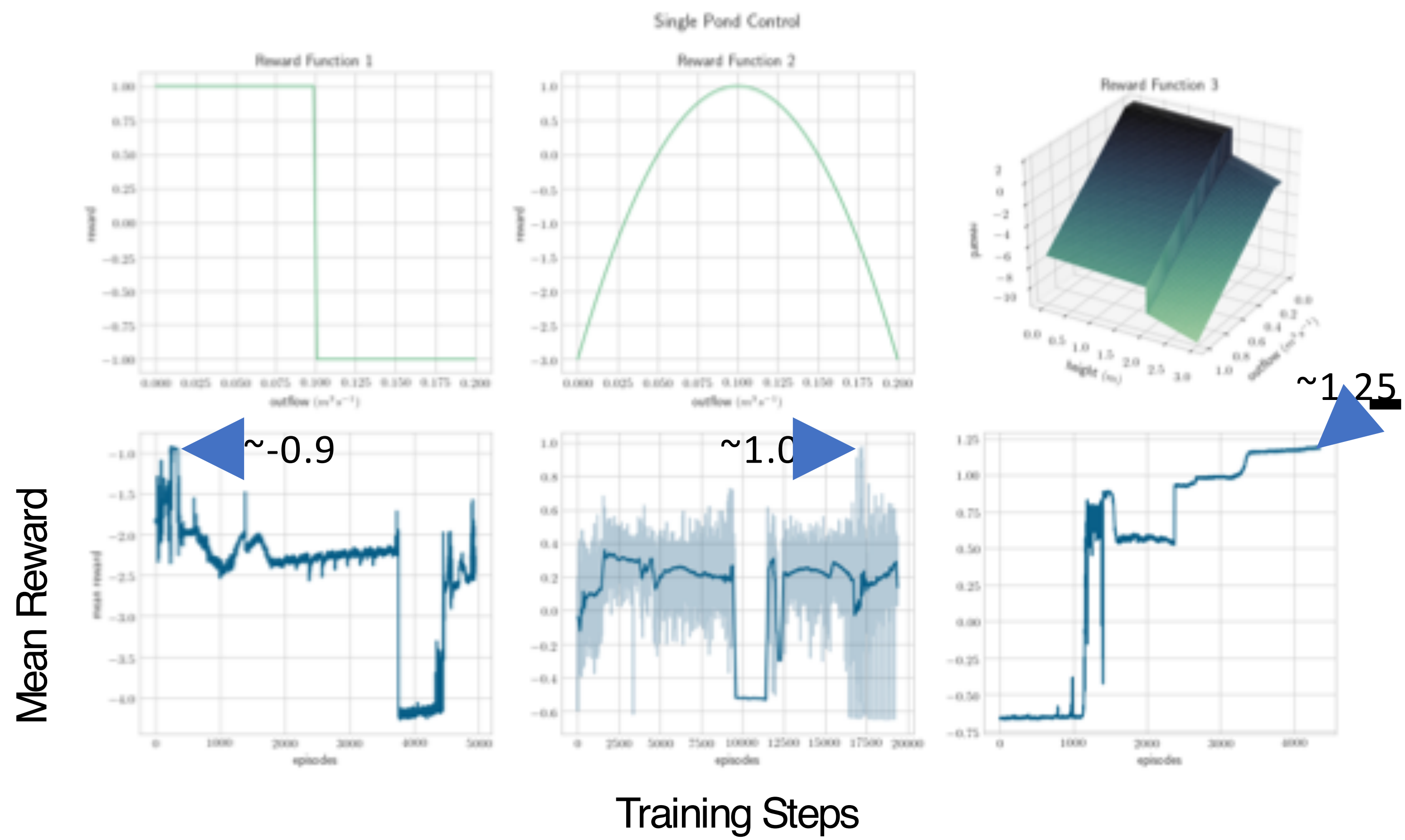


Simulated Water Shed

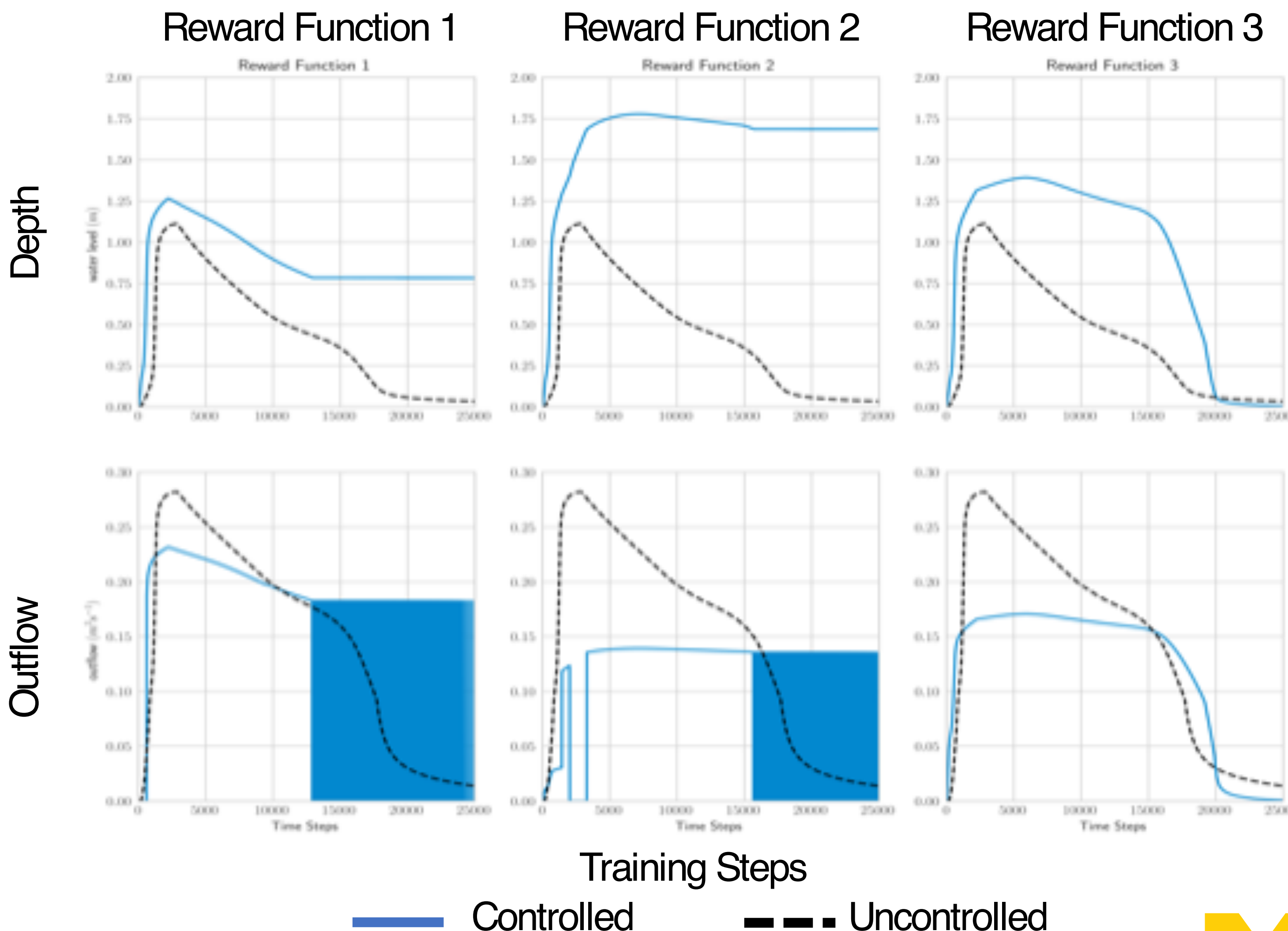
HIC 2018

13TH INTERNATIONAL HYDROINFORMATICS CONFERENCE HIC 2018
1-6 JULY 2018 | PALERMO | ITALY

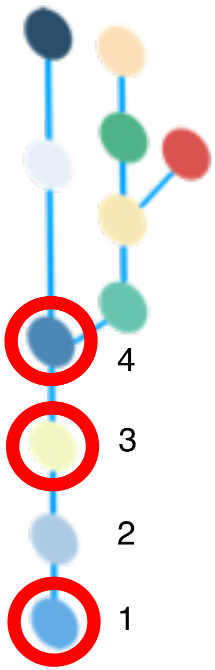
Reward Design - Convergence

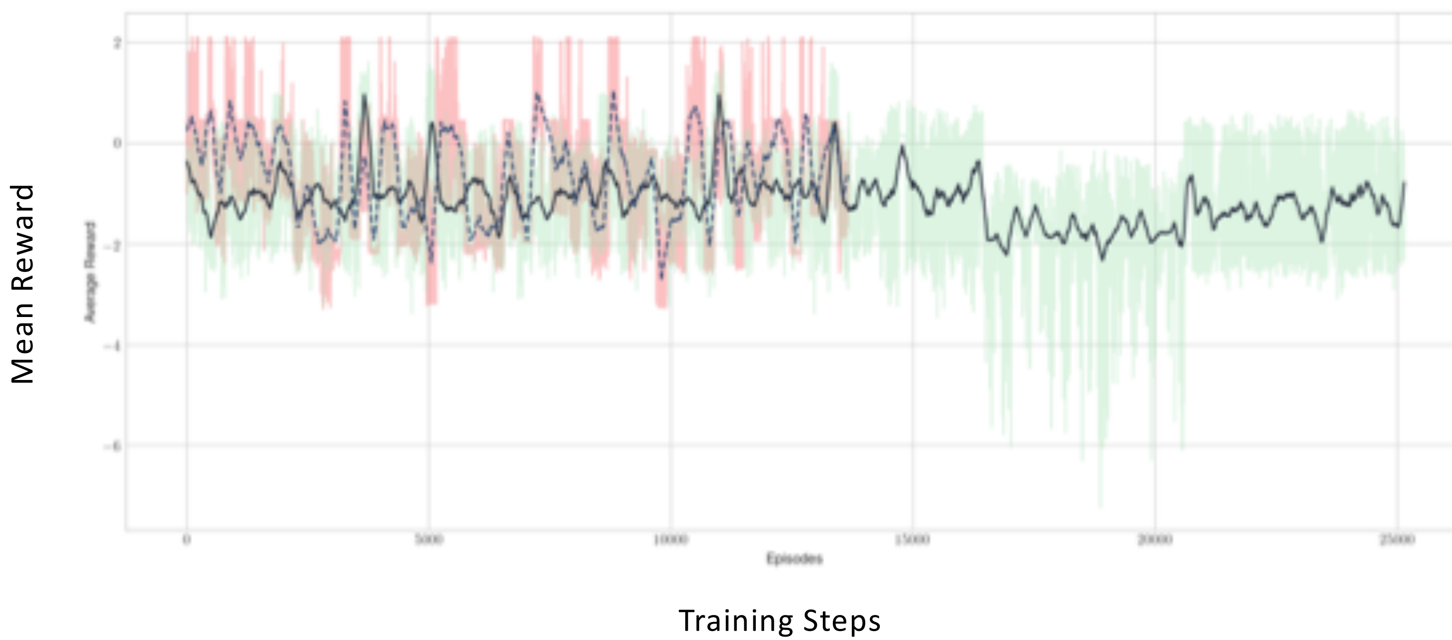


Reward Design - Responses



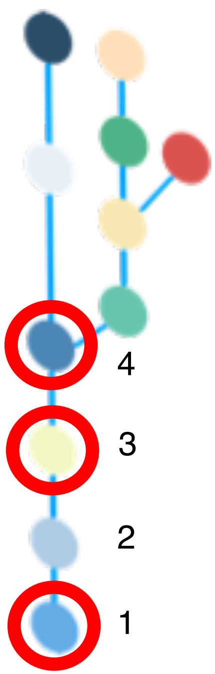
System Scale Control - Training

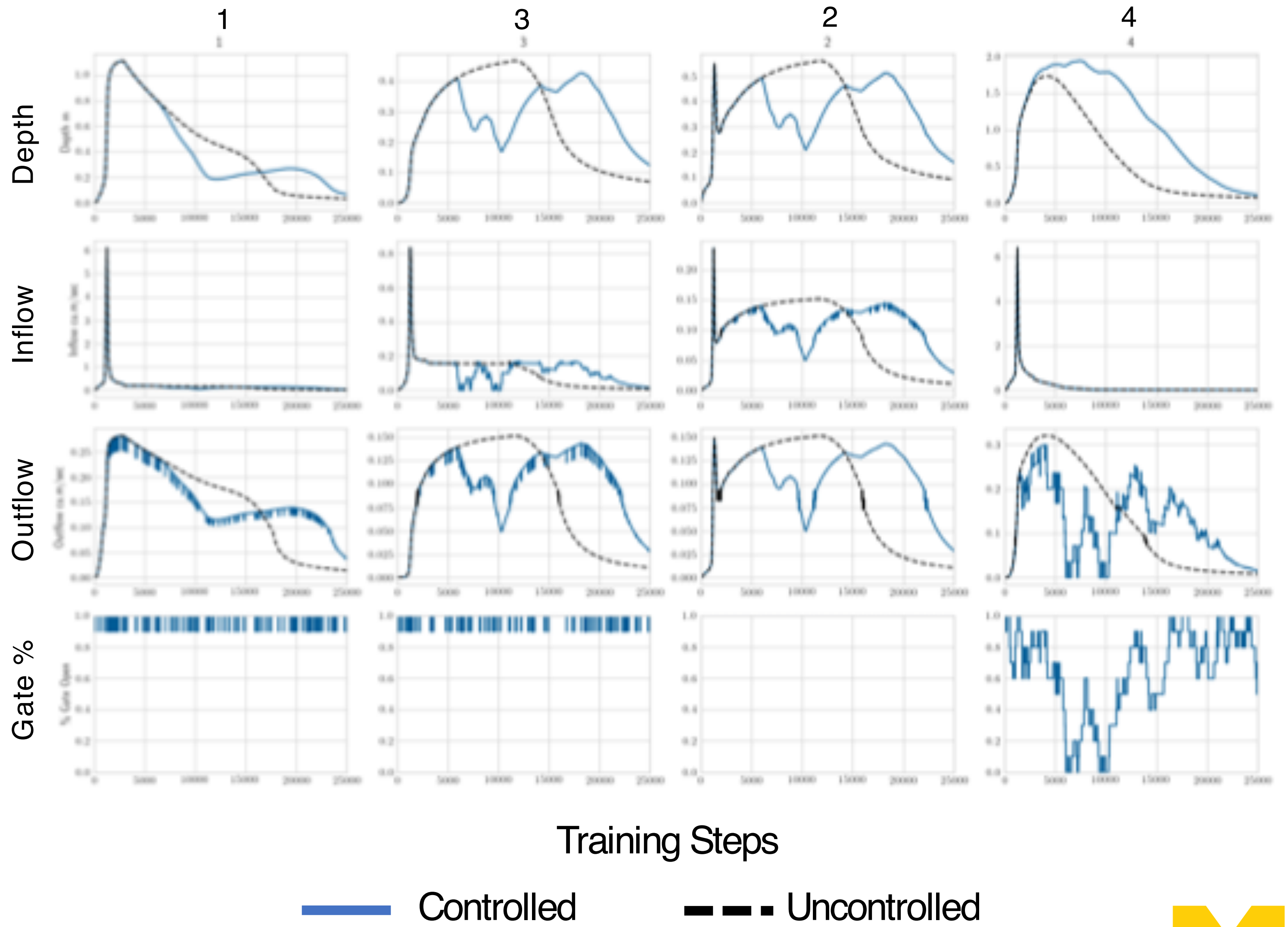




Generic NN
Batch Normalization NN

System Scale Control - Response





Training

- Training RL controller
  - Computational Cost:
    - 2 weeks to 1 month on K40 tesla GPUS
  - Fine-tuning neural networks layers/hyper parameters

System Scale : ~750,000,000 (7.5 Million Steps)
Single Assent : ~125,000,000 (1.25 Million Steps)

High Performance Clusters

applieddatasystem.com/solutions_hpcclusters.aspx

Summary

- Though RL can be used to control the storm water networks, we are far from solving system scale control of storm water networks.
  - Reward design
  - Deep neural networks
  - Randomness of control
- Model free/ Model based reinforcement learning
  - Adding storm water domain knowledge to controller
- RL can be used to address more fundamental questions about the control of storm water networks

Real Time Water Systems Lab

Thank you

Open-Storm.org