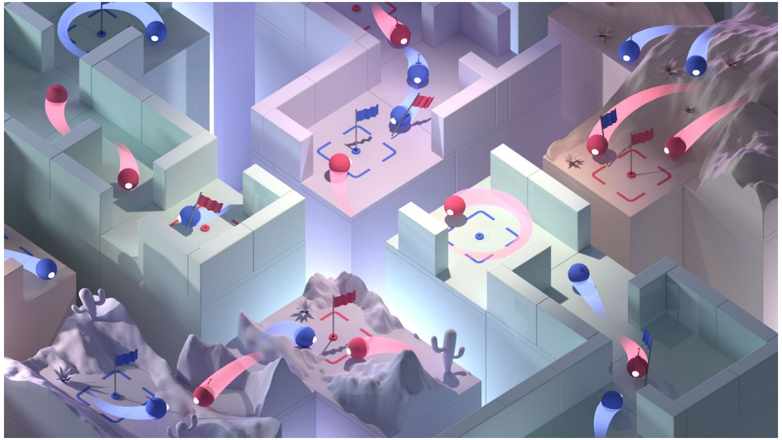
30.11.2020

Reinforcement Learning (RL)

Complex Social Systems HS 2020

Bryan Yu, Anya Heider, Anna Huwyler

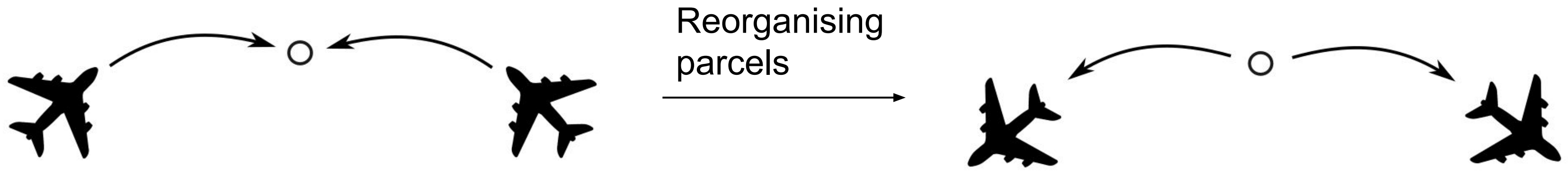
Motivation



→ Need to understand and investigate the influence on actors’ behaviour

Introduction

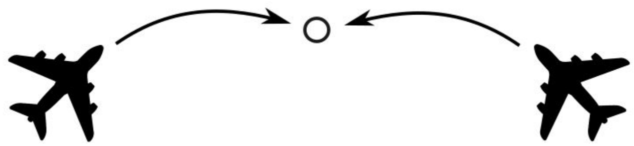
FedEx Story as example:



Delivering parcels with central point of reorganisation of parcels

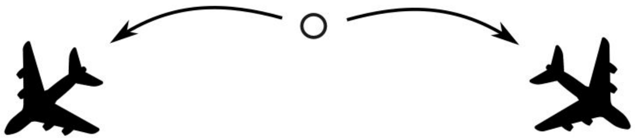
Introduction

FedEx Story as example:



Reorganising parcels

Bottleneck

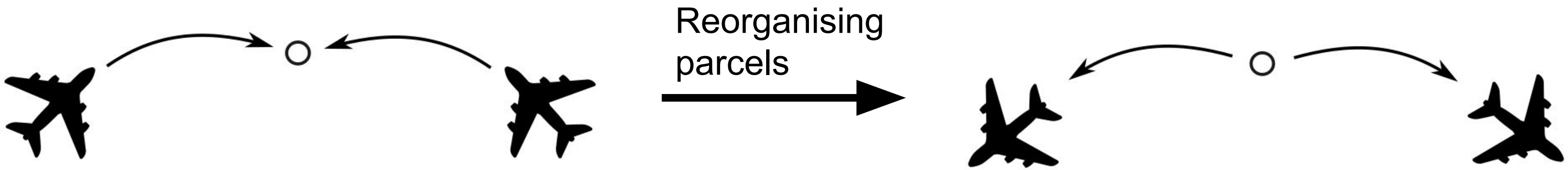


Bottleneck of reorganisation,

difficulties meeting the business targets

Introduction

FedEx Story as example:



Solution:

- Payment not by hour but by successful delivery
- Go home early if done early

Research question:

→ Can this behaviour be modelled with reinforcement learning?

Goals

Evaluate influence of different incentive structures on actors' behaviour

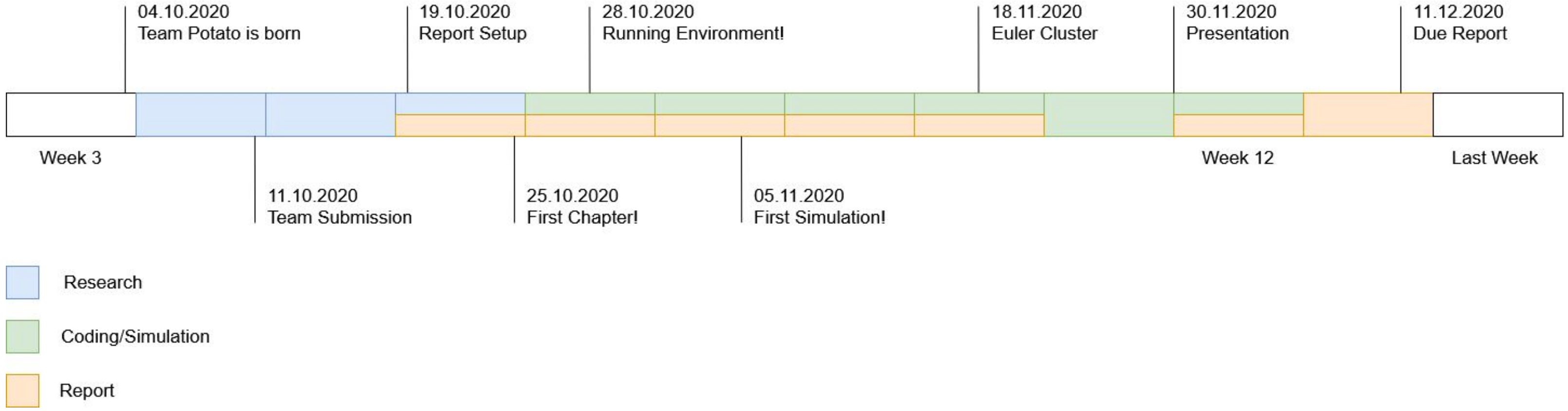
Reward structures:

- Performance dependant
- Time dependant
- At completion of sub tasks or cumulated in the end

In order to do so, set up a suitable model:

- Choose environment
- Choose RL algorithm
- Design reward schemes

Timeline

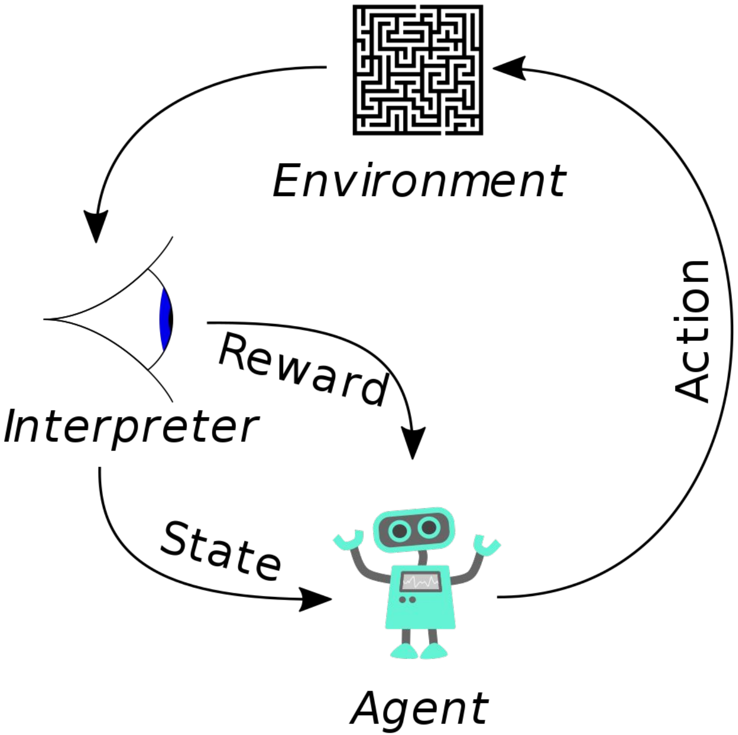


ETH zürich

RL Background + model

Models learning an optimal behavior

- Agent has certain action space within a given environment/state (a_t, s_t)
- Makes a series of sequential decisions
- Based on incentive scheme, receives reward (r_t)
- Goal of the agent is to maximise the total reward by adopting a policy ($\pi(a_t \mid s_t ; \theta)$)



RL Background + model

Markov Decision Process:	R_t	= ∑ γ_k* r_(t+k)	γ_k: discount factor
How do we rank a state?	V(s_t)	= E[R_t|s_t]	
How else can we rank a state?	Q(s_t,a_t)	= E[R_t|s_t ; a_t]	
(i.e. along with its action!)			
Advantage Value:	A(s_t;a_t)	= Q(s_t, a_t) - V(s_t)	
What are we changing to optimize?	θ	π(a_t | s_t ; θ)	
	$J(\theta)$	= E[∑ R_t ; π(θ)]	
Gradient step	$\nabla_\theta J(\theta)$	= 0	

policy gradient methods: A2C, PPO and DQN

RL Background - Policy gradient methods

Deep Q-Networks (DQN):

- Applies convolutional neural network for the agents to learn the control policies
- Introduces experience replay
- Epsilon greedy strategy: Agent occasionally choses some random action instead of most probable action

$$Q^{new}(s_t, a_t) \leftarrow \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \overbrace{\Bigg(\underbrace{\underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}}}_{\text{new value (temporal difference target)}} - \underbrace{Q(s_t, a_t)}_{\text{old value}} \Bigg)}^{\text{temporal difference}}$$

- Reason for choice: Milestone in modern reinforcement learning

RL Background - Policy gradient methods

Advantage Actor Critic (A2C):

- Shared model of multiple parallel workers
- Synchronous version of A3C [turns out the asynchronous updates were not necessary]
- Gradient step: $E[\nabla_\theta \log(\pi(a_t|s_t;\theta_v))A(s_t,a_t;\theta,\theta_v)]$

  A: Advantage function
  π: Policy
  θ: Critic

- Reason for choice: Advantage learning efficiency

RL Background - Policy gradient methods

Proximal Policy Optimization (PPO):

- Introduces trust region
- Idea: Modifications should not cause too much differentiation from existing policy → constraints on the size of the updates
- Objective function:

$$L^{CLIP}(\theta) = \hat{\mathbb{E}}\Big[min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)\Big]$$

  Â: Advantage function
  θ: Policy parameters
  ε: Hyperparameter

- Reason for choice: Advantages due to trust regions, multiple workers and continuous control

Environment

StarCraft II Learning Environment:

- Existing library pysc2 → standardised environment with wide action space

Minigame “Collect Mineral Shards”:

- Two agents
- Simple flat square map
- Randomly distributed minerals (20)
- Refreshed when all minerals are collected
- Goal of agents: collect minerals in a given amount of time
- Action space: Move up, down, left, right

Incentive structures

None:

- Default reward provided at collection of mineral

Time Incentive:

- -20 reward for every step taken

Mineral Thresholding:

- Negative Reward for every time step taken
- Stop negative (Time Incentive) once a certain threshold mineral is collected

Episodic Reward:

- Reward will only be given once after full simulation period

Experiment

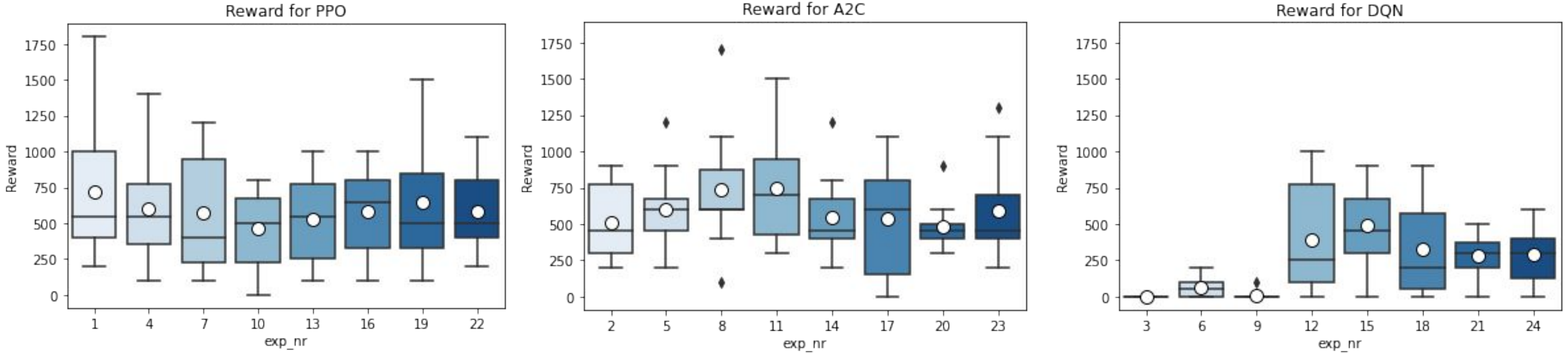
Algorithm	Mineral Reward	Mineral Thresholding	Episodic Reward
DQN, A2C, PPO	Yes, No	Yes, No	Yes, No

→ 24 different constellations

Case Study:

- Run all the constellations
- Training for 10^6 time steps
- Tests with trained models for 10 episodes
- Compare results to baseline random agent

Results - Comparison



- Best values for experiments 1, 8, 11
- DQN seems to go get worse results and take longer for decisions

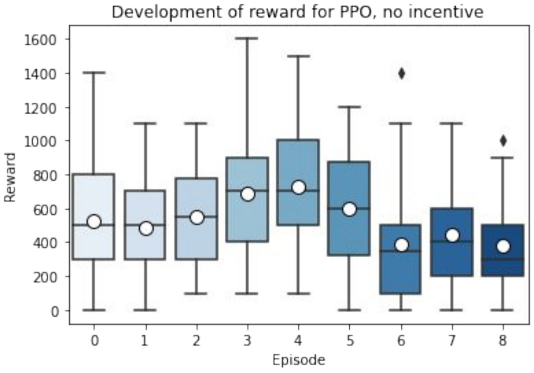
Results - Comparison of best runs

exp_nr	Algorithm	Incentive	Max. Value	Mean Value
1	PPO	no	1800	720
8	A2C	episodic	1700	740
11	A2C	episodic, time incentive	1500	750

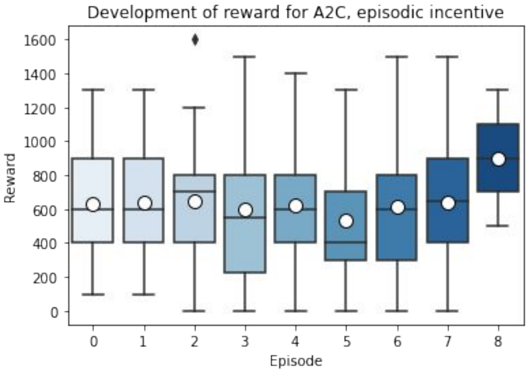
- Normal incentive structure for PPO
- Episodic learning seems to work for A2C

Results - Training

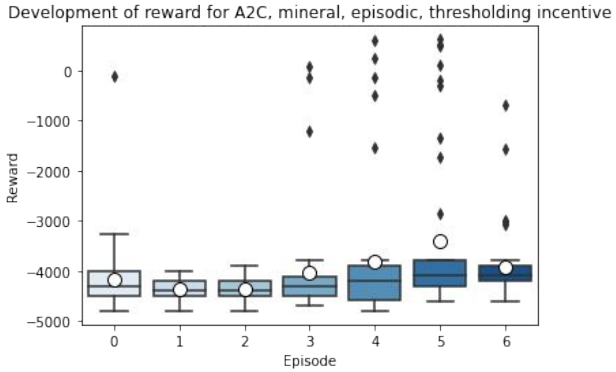
exp_nr: 1



exp_nr: 8

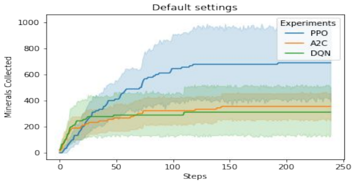


exp_nr: 23

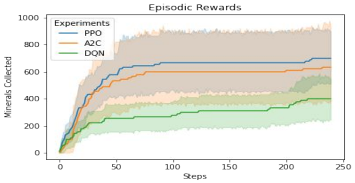


- In general: No clear trend visible (see exp_nr = 1), training period might not be sufficiently high
- Slight learning trend visible for A2C (see exp_nr = 8) [episodic rewards]
- For combination of three incentives, agents seem to become faster in exp_nr = 23

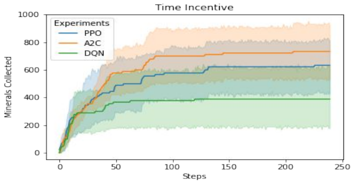
Results



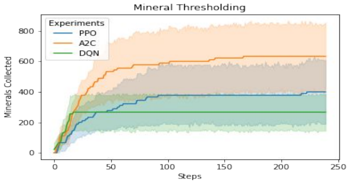
Default settings
Experiments
PPO
A2C
DQN
Minerals Collected
Steps



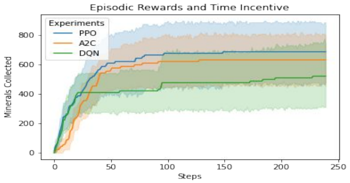
Episodic Rewards
Experiments
PPO
A2C
DQN
Minerals Collected
Steps



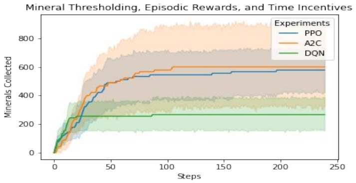
Time Incentive
Experiments
PPO
A2C
DQN
Minerals Collected
Steps



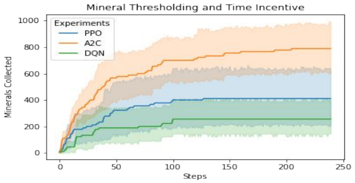
Mineral Thresholding
Experiments
PPO
A2C
DQN
Minerals Collected
Steps



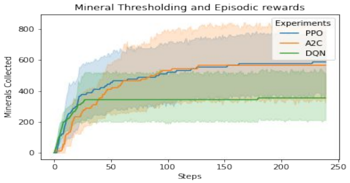
Episodic Rewards and Time Incentive
Experiments
PPO
A2C
DQN
Minerals Collected
Steps



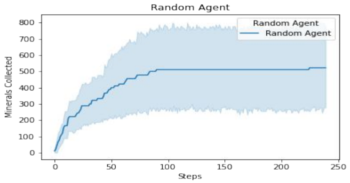
Mineral Thresholding, Episodic Rewards, and Time Incentives
Experiments
PPO
A2C
DQN
Minerals Collected
Steps



Mineral Thresholding and Time Incentive
Experiments
PPO
A2C
DQN
Minerals Collected
Steps



Mineral Thresholding and Episodic rewards
Experiments
PPO
A2C
DQN
Minerals Collected
Steps



Random Agent
Random Agent
Random Agent
Minerals Collected
Steps

Conclusion

- DQN seems to be slower than other algorithms
- Training period might not be sufficiently high → extend training period in order to be able to make final conclusions
- Slight learning trend visible for A2C and PPO → focus on these model?
- Time Incentive appears to improve performance by 200 minerals over the random agent - given 1e6 training episodes. We expect this to improve after we train for a further period of time!

Conclusion

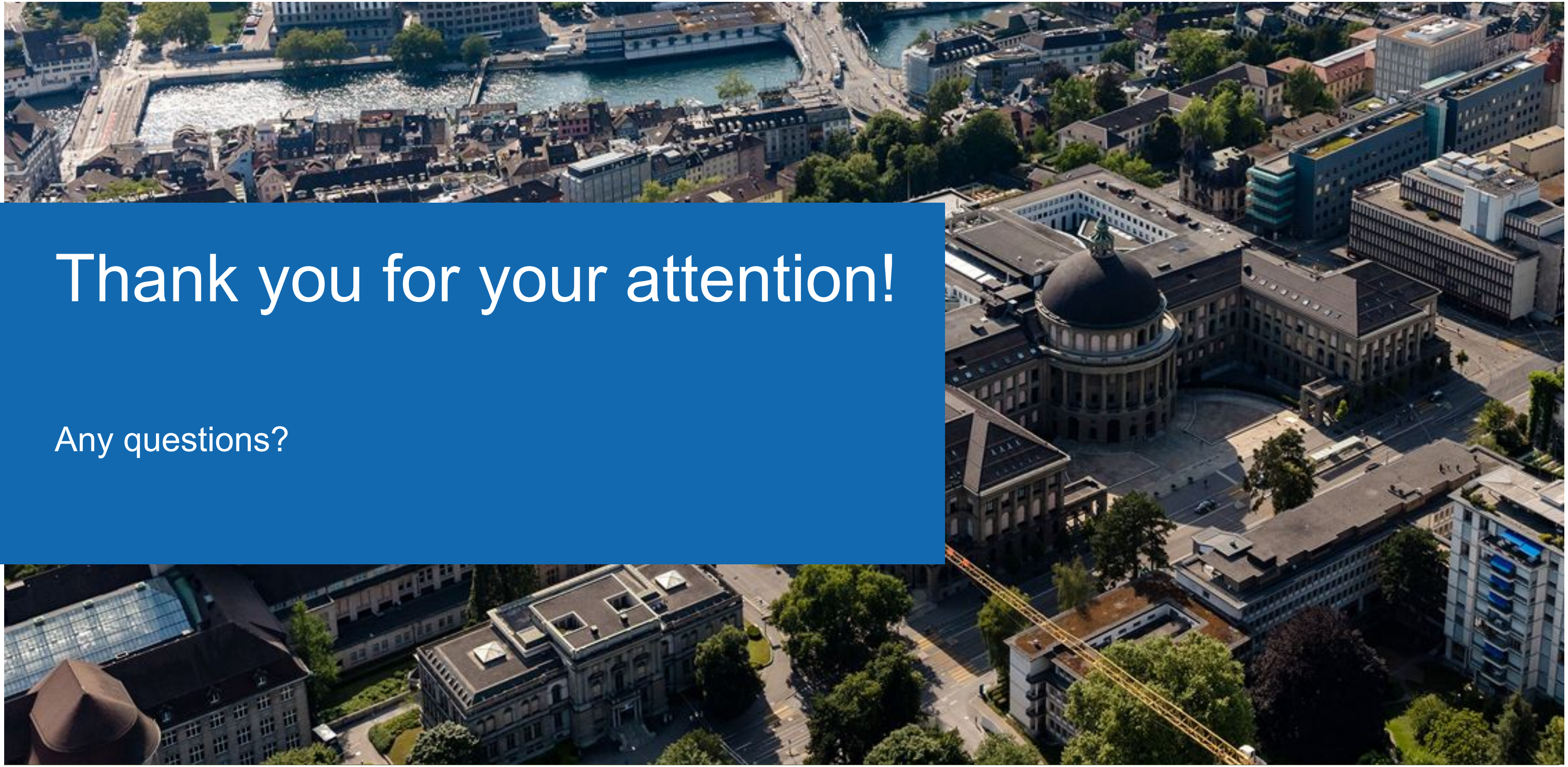
- DQN seems to be slower than other algorithms
- Training period might not be sufficiently high → extend training period in order to be able to make final conclusions
- Slight learning trend visible for A2C and PPO → focus on these model?

Outlook

Finish testing:

- Choose algorithm that performs best
- Extend the training periods for the three incentive structures
- Compare results
- Possibly extend to multi-agent (i.e. one controller per marine)

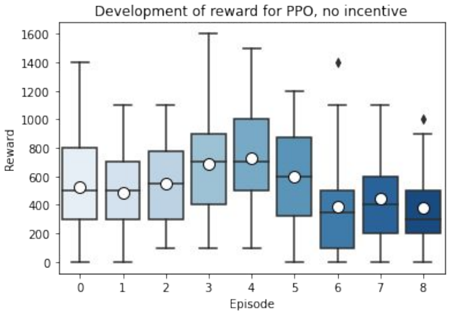
30.11.2020

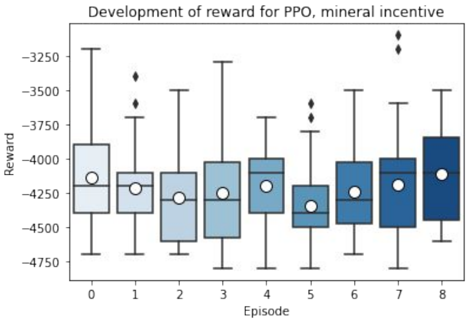


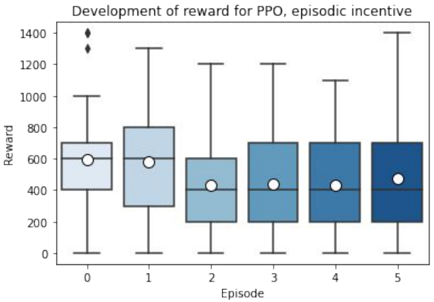
Thank you for your attention!

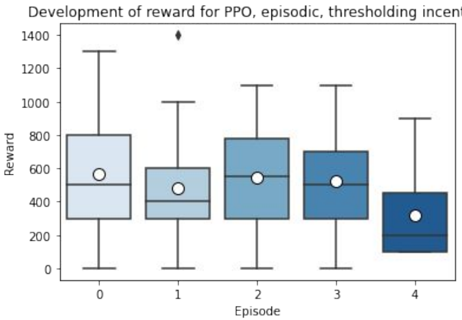
Any questions?

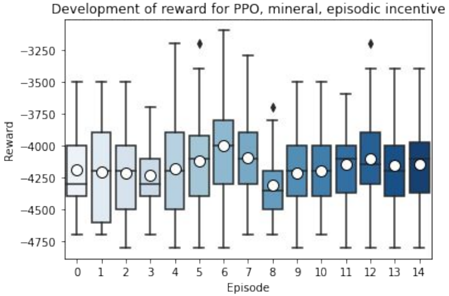
Results - PPO

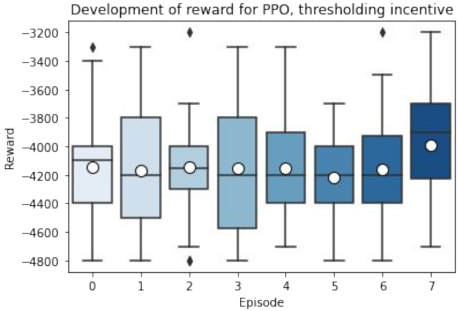

Development of reward for PPO, no incentive
Reward
Episode

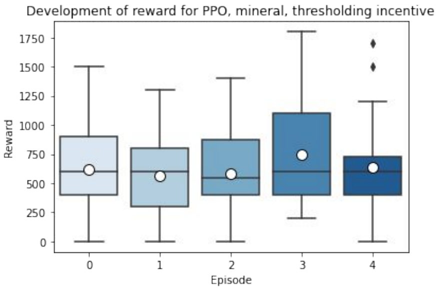

Development of reward for PPO, mineral incentive
Reward
Episode

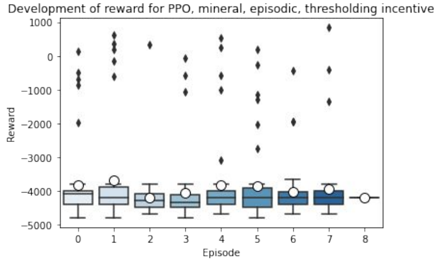

Development of reward for PPO, episodic incentive
Reward
Episode


Development of reward for PPO, episodic, thresholding incent
Reward
Episode

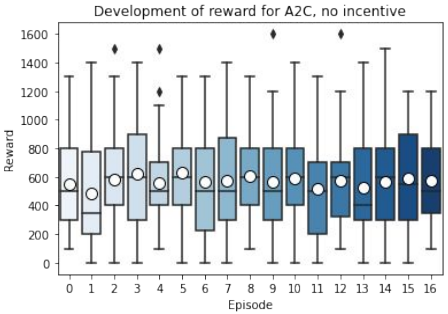

Development of reward for PPO, mineral, episodic incentive
Reward
Episode

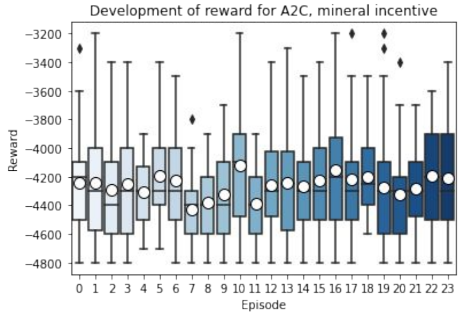

Development of reward for PPO, thresholding incentive
Reward
Episode

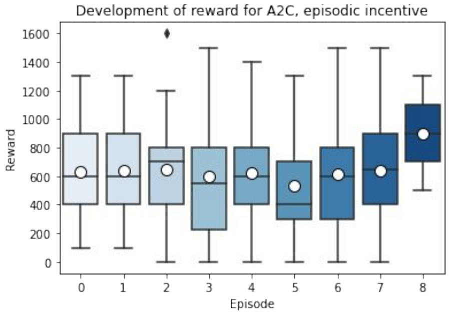

Development of reward for PPO, mineral, thresholding incentive
Reward
Episode

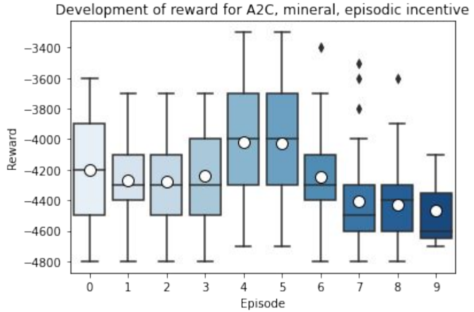

Development of reward for PPO, mineral, episodic, thresholding incentive
Reward
Episode

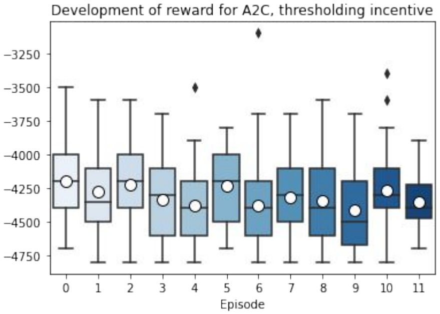
Results - A2C

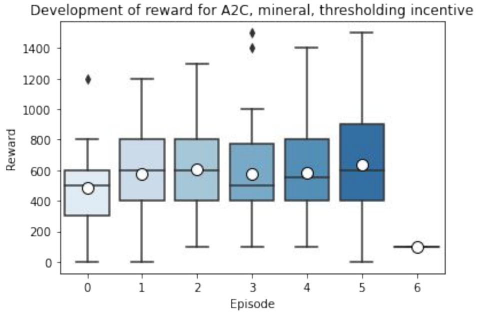

Development of reward for A2C, no incentive
Reward
Episode

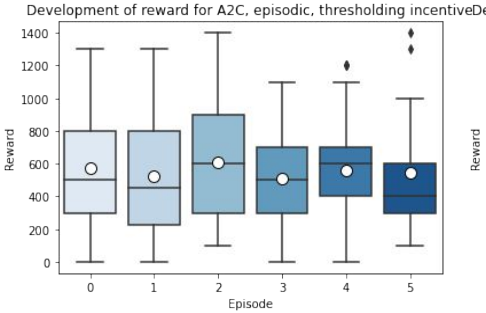

Development of reward for A2C, mineral incentive
Reward
Episode

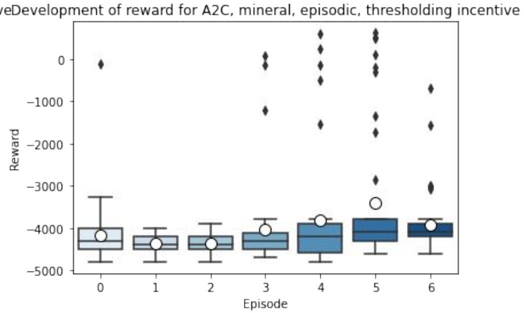

Development of reward for A2C, episodic incentive
Reward
Episode


Development of reward for A2C, mineral, episodic incentive
Reward
Episode

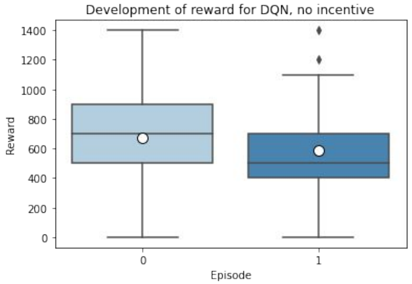

Development of reward for A2C, thresholding incentive
Episode

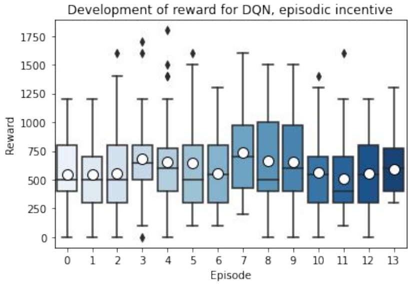

Development of reward for A2C, mineral, thresholding incentive
Reward
Episode

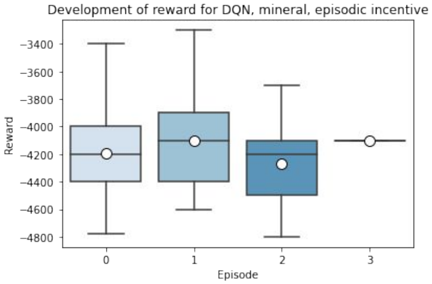

Development of reward for A2C, episodic, thresholding incentive
Reward
Episode

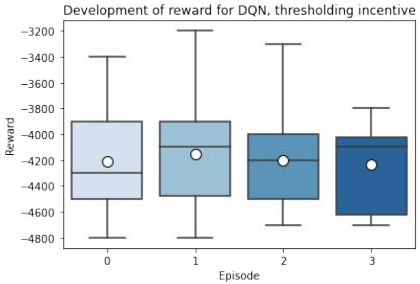

Development of reward for A2C, mineral, episodic, thresholding incentive
Reward
Episode

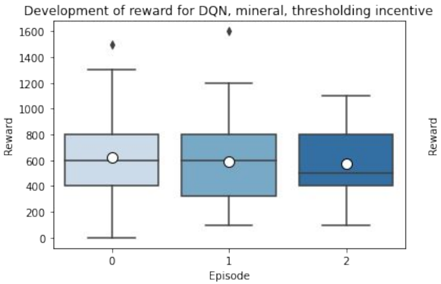
Results - DQN

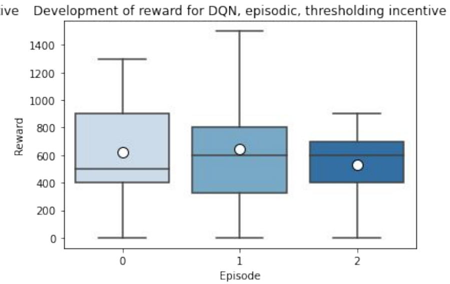


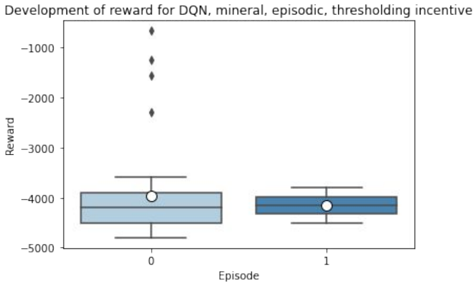






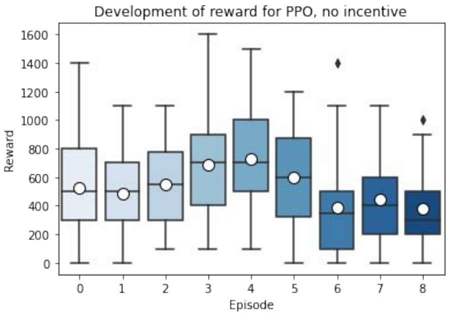


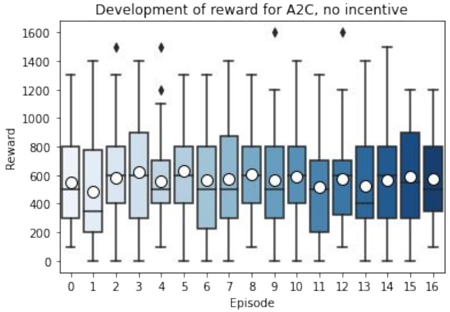


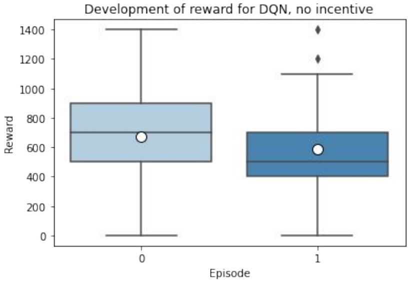


ETH zürich

Results - no incentive

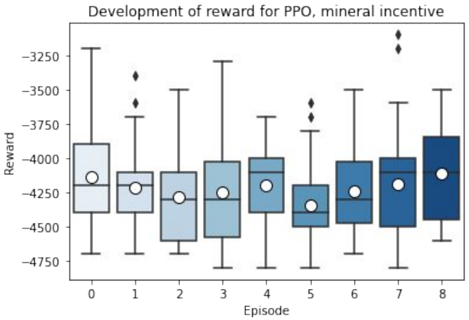


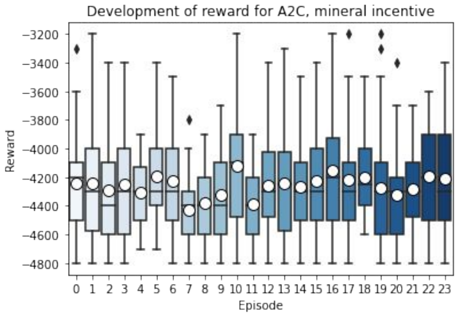




- Values around 500-700
- DQN seems to go through less episodes → Slower?
- No significant improvement visible (even worse for PPO)

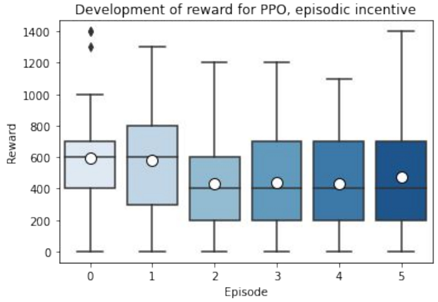
Results - mineral incentive

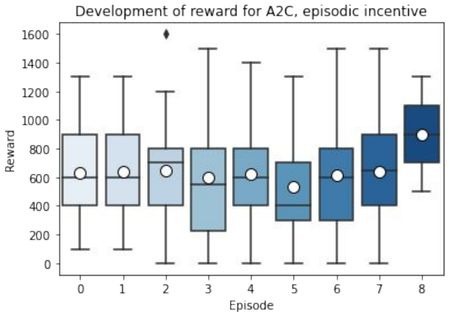


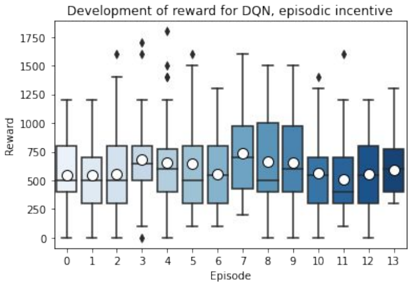


- Values around -4200
- No significant increase visible

Results - episodic incentive

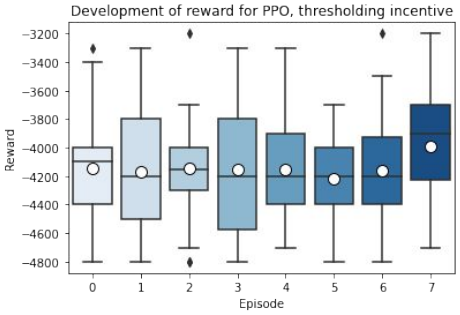


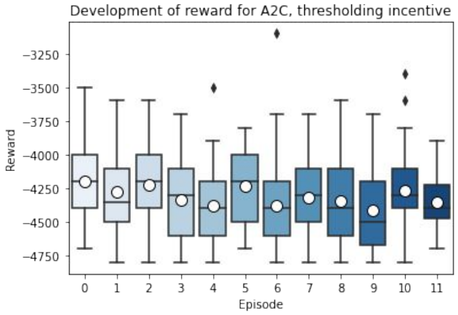


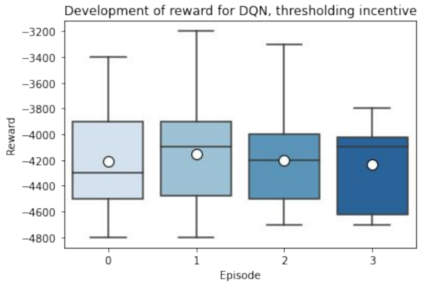


- Values between 400 and 900
- A2C seems to improve performance over time

Results - thresholding incentive

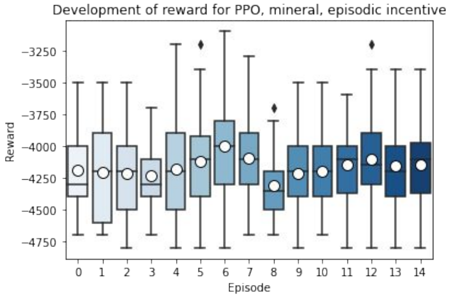


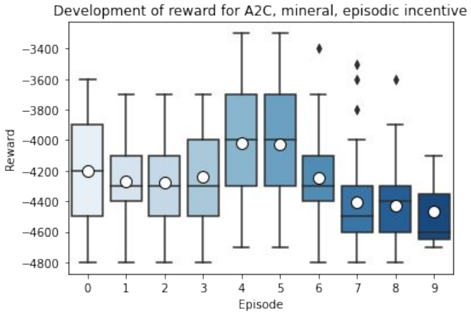


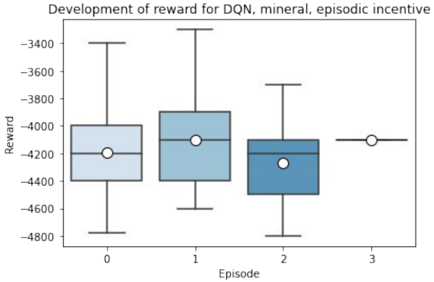


- Values around -4200
- PPO seems to slightly improve performance over time

Results - episodic mineral incentive

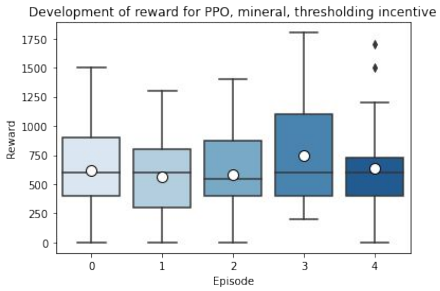


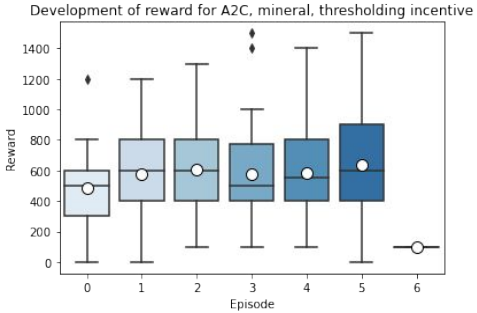


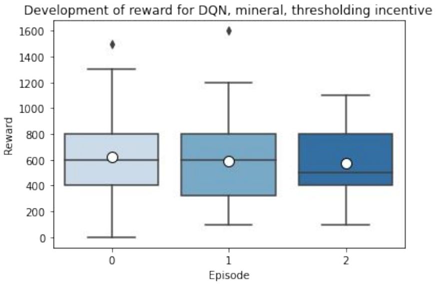


- Values around -4200
- Performance drop for A2C

Results - thresholding mineral incentive

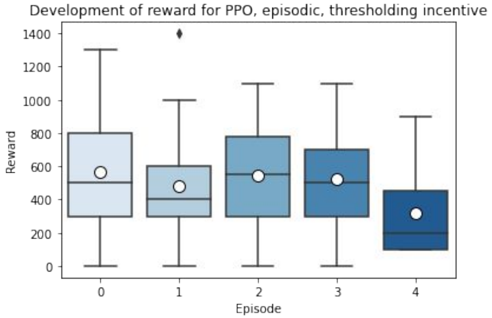


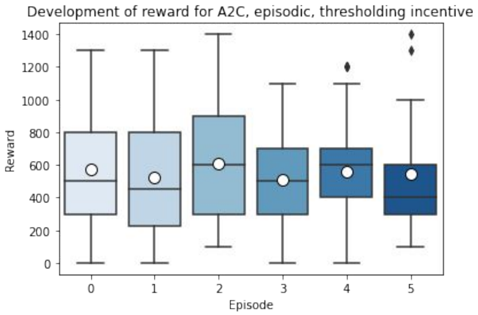


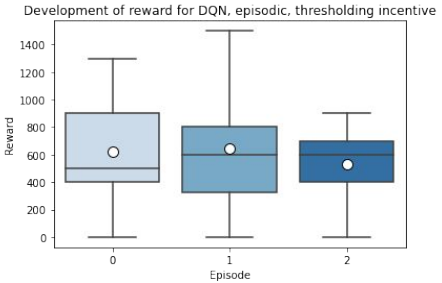


- Values around 600

Results - episodic thresholding incentive

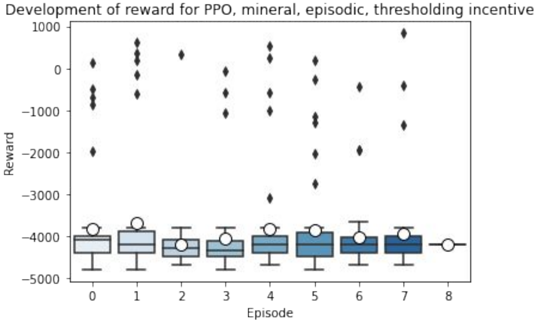


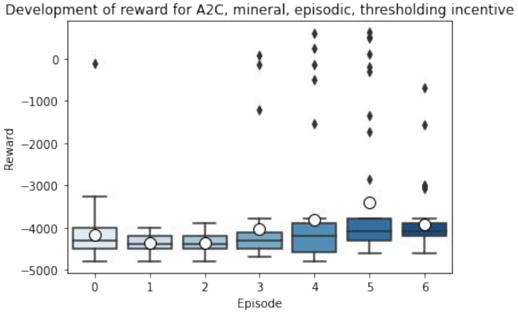


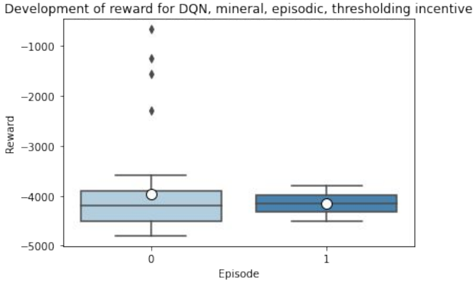


- Values around 600

Results - episodic thresholding mineral incentive

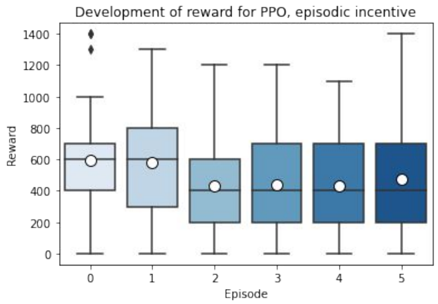


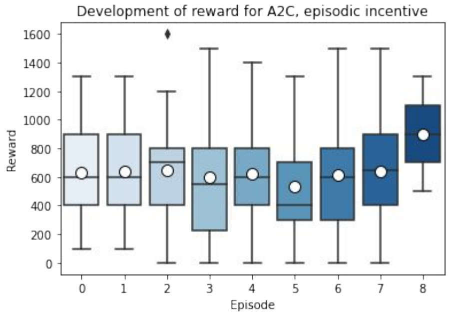


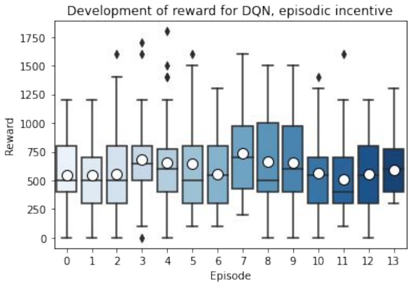


- Values around -4000
- For A2C agents seem to become faster

Results - episodic incentive

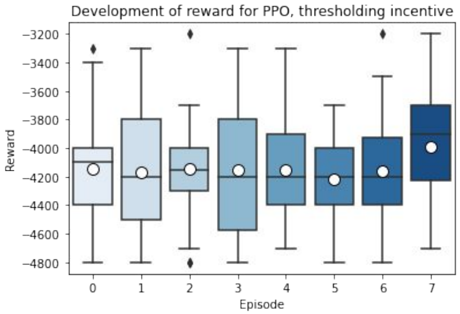


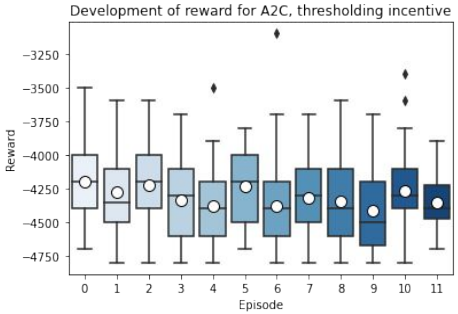


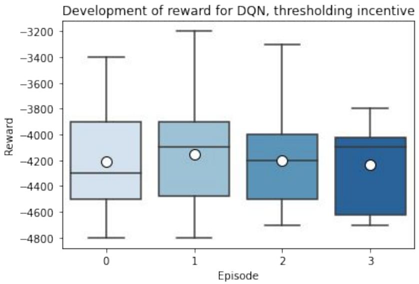


- Values between 400 and 900
- A2C seems to improve performance over time

Results - thresholding incentive

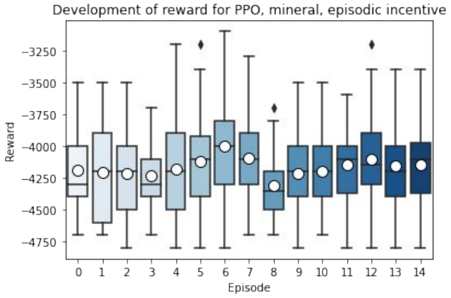


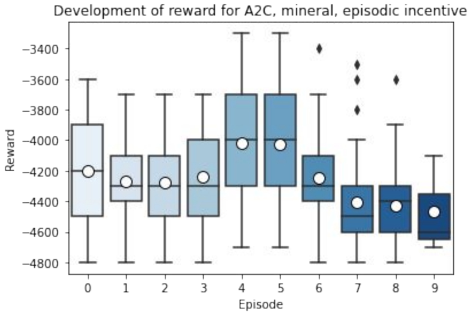


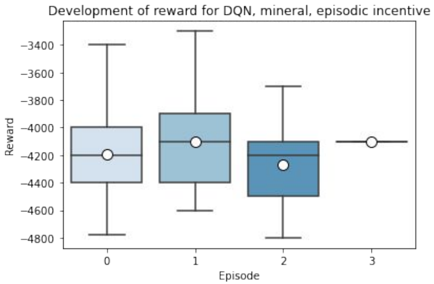


- Values around -4200
- PPO seems to slightly improve performance over time

Results - episodic mineral incentive

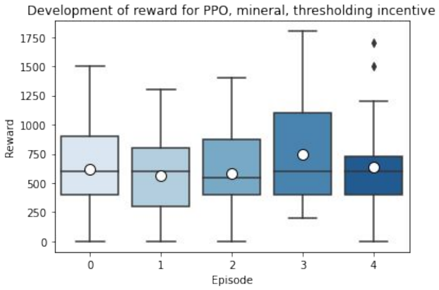


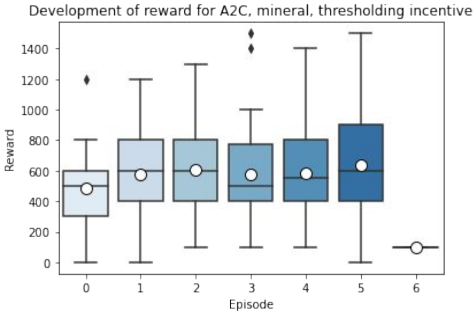


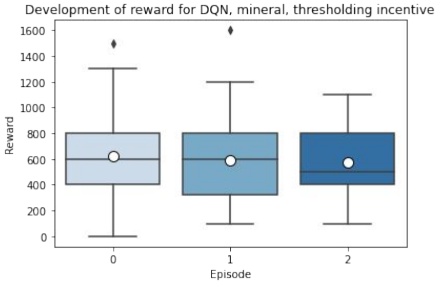


- Values around -4200
- Performance drop for A2C

Results - thresholding mineral incentive

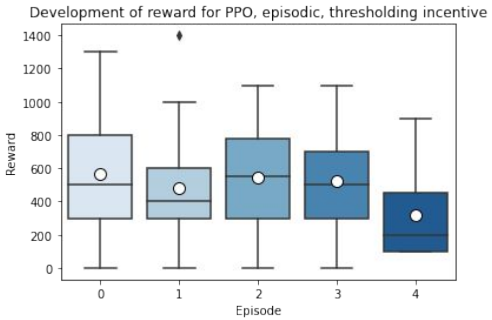


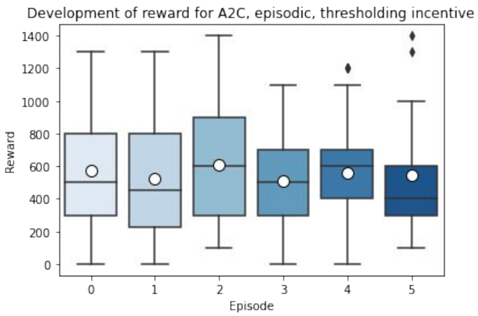


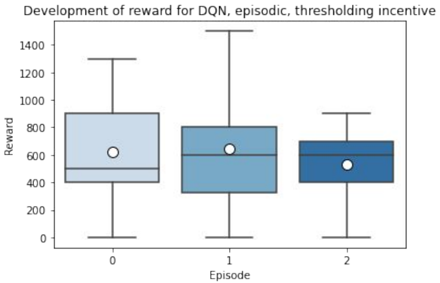


- Values around 600

Results - episodic thresholding incentive

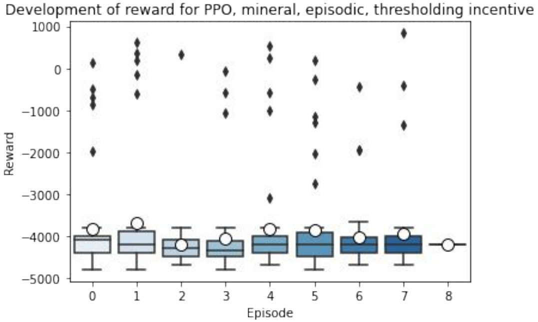


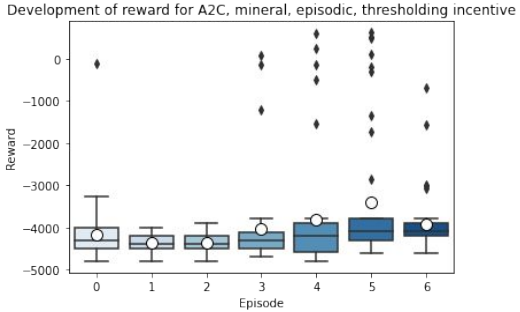


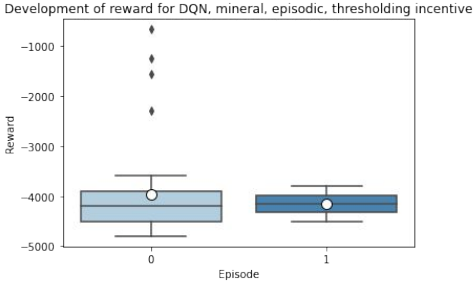


- Values around 600

Results - episodic thresholding mineral incentive







- Values around -4000
- For A2C agents seem to become faster

Team?

Anya Heider: Renewable Energy Engineering background, PhD on “Power System Flexibility in the German Energy System” in cooperation with the Reiner Lemoine Institut gGmbH seated in Berlin