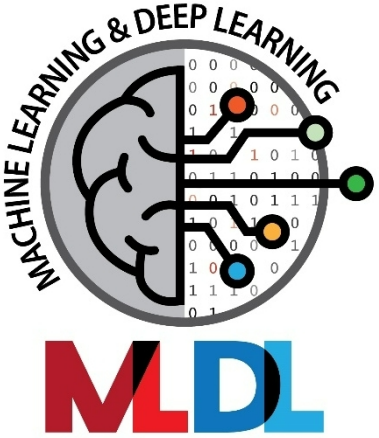
SAND2022-9851C



MLDL

Machine Learning and Deep Learning Conference 2022

Hybrid Deep Reinforcement Learning For Online Distribution Power System Optimization and Control

- Nicholas Corrado 8722
- Michael Livesay 8722
- Jay Johnson 8812
- Tyson Bailey 5683
- Drew Levin 8721

- Funding Source: LDRD

Motivation

- Decentralization in the power industry makes power systems more vulnerable to attacks.

- Prior work on grid resilience primarily uses optimization techniques
  - May not scale to large systems
  - Not designed to defend against an active adversary

- Can a reinforcement learning (RL) agent defend a distribution power system by controlling a collection of utility-owned distributed energy resources?

Contributions

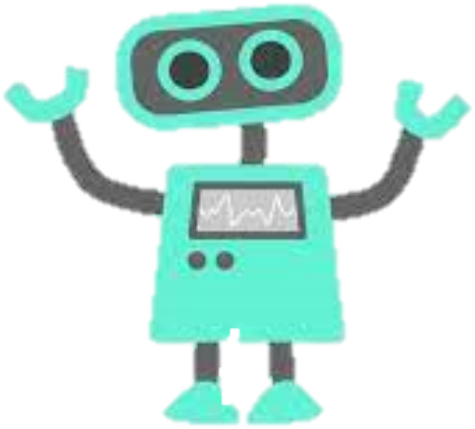
- Prior work on RL-based grid resilience focus on discrete-action settings or continuous-action settings. We are the first to consider a parameterized-action setting, a more natural setting for grid resilience tasks.
  - Agent can learn optimal DER setpoints as well as the optimal path to the optimal setpoints.
- We introduce a deterministic greedy algorithm, and find that it performs quite well.
- We empirically demonstrate that RL agents can successfully regulate distribution systems and outperform the greedy algorithm.
- We evaluate several RL algorithms and observe that algorithms specially designed for parameterized action tasks are significantly more data efficient.
- This work takes an additional step towards a more realistic multi-player distribution system control game

Reinforcement Learning Interaction Protocol

Policy π

Action	Left	Down	Up	Right
Probability	0.4	0.1	0.2	0.3

agent
(wants to maximize reward)



action

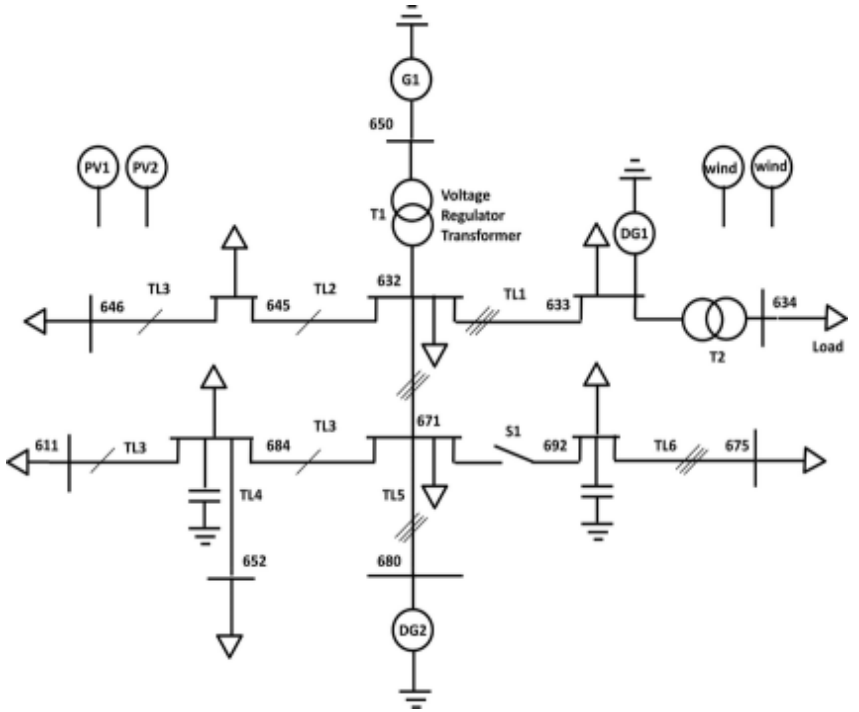
observation (state),
reward

environment

Power System 1: IEEE 13-bus Model

- 14 controllable DERs
- Agent controls the active and reactive power of each DER
- On-load tap changing transformers (LTCs) adjust the number of windings on the transformer to correct low/high voltages. We assume the agent makes decisions very quickly, allowing us to ignore the LTC dynamics.

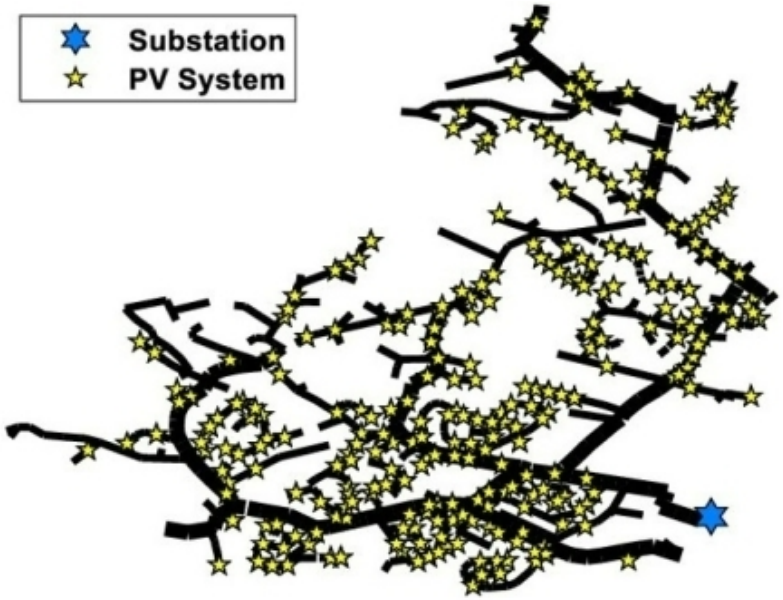
- **IEEE-balanced:** LTCs are tapped to default values.
- **IEEE-unbalanced:** LTCs are tapped to the 0.95 pu state to produce a severe voltage condition in which all bus voltages are less than 0.95 pu.



Power System 2: EPRI Ckt5 Model

- 701 total controllable DERs
- Agent controls the power factor of each DER
  - Power factor = ratio of active and reactive power

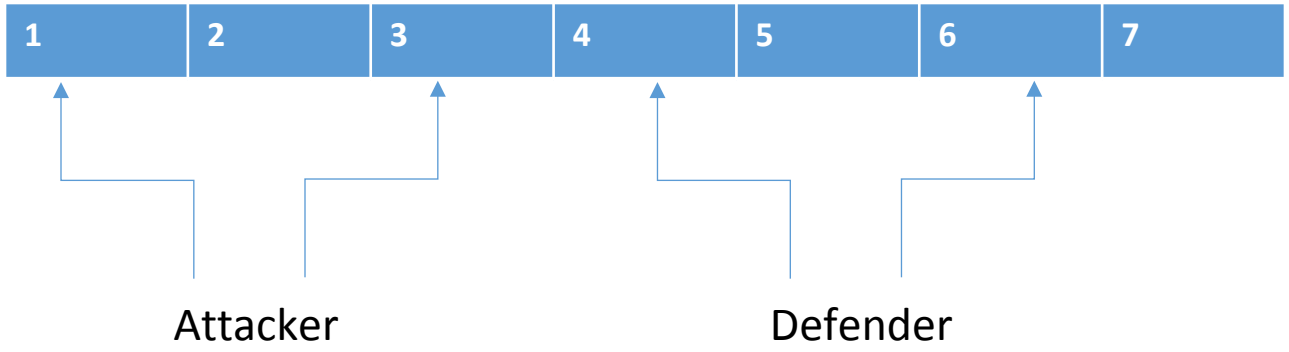
- **EPRI-14:** Agent only controls DERs with the 14-largest power ratings.
- **EPRI-32:** Agent only controls DERs with the 32-largest power ratings.



Discrete Environment Experimentation

- As part of a discrete version of the environment we experimented with splitting the range of busses that a given agent can control.
- With training they often found equilibriums where each agent would settle on a small set of actions 1-2 it would repeat alternating between turns.
- An example score after training:
  - Good Agent: -0.45708
  - Bad Agent: 0.04969*

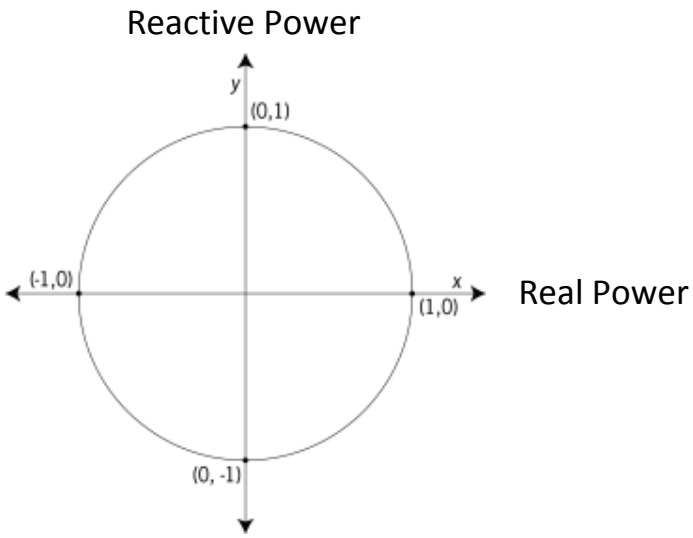
*The Bad Agent was able to get the best reward in this case.



- Future Work: Training the agent to act optimally with any subset of busses.

Continuous Environment Experimentation

- Let n = number of controllable DERs (i.e. number of discrete actions)
- **Parameter space $\mathcal{X}$:** The set of possible setpoints for a single DER
  - IEEE: $\mathcal{X}$ = unit disk
  - EPRI: $\mathcal{X} = [-1, +1]$
- **Action space:** $\{0, \dots, n-1\} \times \mathcal{X}$
  - For the EPRI model, action $(7, 0.4)$ changes the setpoint of DER 7 to 0.4
  - For the IEEE model, action $(7, (0.4, -0.2))$ changes the setpoint of DER 7 to $(0.4, -0.2)$
- **State space:** The current setpoints of all DERs (i.e. a list of n points in $\mathcal{X}$)
- **Initial state distribution:** All bus states are initialized to a point in or on the unit circle uniformly at random



Parameter space for the IEEE model

Parameter space for the EPRI model

Continuous Environment Experimentation

- **Reward:** negative sum of squared errors of bus voltages compared to nominal voltage values.

$$r = -\sum_{i=0}^{n-1} (V_i - V_i^*)^2$$

where V_i and V_i^* are the current voltage and nominal voltage of DER i, respectively.

- **Objective:** Maximize the expected discounted reward

$$J = \mathrm{E}_\pi \left[\sum_{t=1}^{T} \gamma^t r_t \right]$$

where $\gamma \in (0,1)$ is a discounting factor and $T = 100$ is the horizon

- **Objective Interpretation:** Stabilize the system by bringing voltages as close to nominal as possible.

Parameterized Action Spaces

- At each step, the agent chooses which DER to modify (a discrete action) and a new setpoint for the chosen DER (continuous parameters).

- We generalize continuous-action RL algorithms to handle parameterized actions using the technique introduced in [https://arxiv.org/pdf/1511.04143.pdf]:
  - Choosing a discrete action: Use n output weights followed by a softmax activation, and then a sample from the resulting distribution.
  - Choosing the continuous parameters: output continuous parameters for all discrete actions, and then select the parameters corresponding to the choses discrete action.
  - This is an ad-hoc technique: the agent must learn that only the continuous parameters corresponding to the chosen discrete action affect the environment.

- The Multi-Pass Deep Q-Network (MPDQN) algorithm is specially designed to handle parametrized actions
  - Special architecture lets the agent know that only the continuous parameters corresponding to the chosen discrete action affect the environment.

Greedy Algorithm

- Coordinate descent-based approach

- At each step, the algorithm identifies a set of promising actions—one for each DER—and then chooses the action from this set that maximally increase its immediate reward.

- Define:
  - s = current state
  - s_i = current state of DER i
  - $r_i(s, x)$ = immediate reward for changing the setpoint of DER i to x in state s.
- For each DER, we approximate the gradient $\nabla_x r_i(s, x)$.
- Let $x_i' = s_i + \eta \nabla_x r_i(s, x)$, where η is a small step size parameter
- The agent then chooses the action (i, x_i') with the maximum immediate reward

Experiments: Setup

- **RL algorithms:**
  - Proximal Policy Optimization (PPO)
  - Deep Deterministic Policy Gradient (DDPG)
  - Soft Actor-Critic (SAC)
  - Multi-Pass Deep Q-Network (MPDQN)

- **IEEE Model:** Train agents over 10k episodes, evaluate performance every 100 episodes

- **EPRI Model:** Train agents over 50k episodes, evaluate performance every 1k episodes.

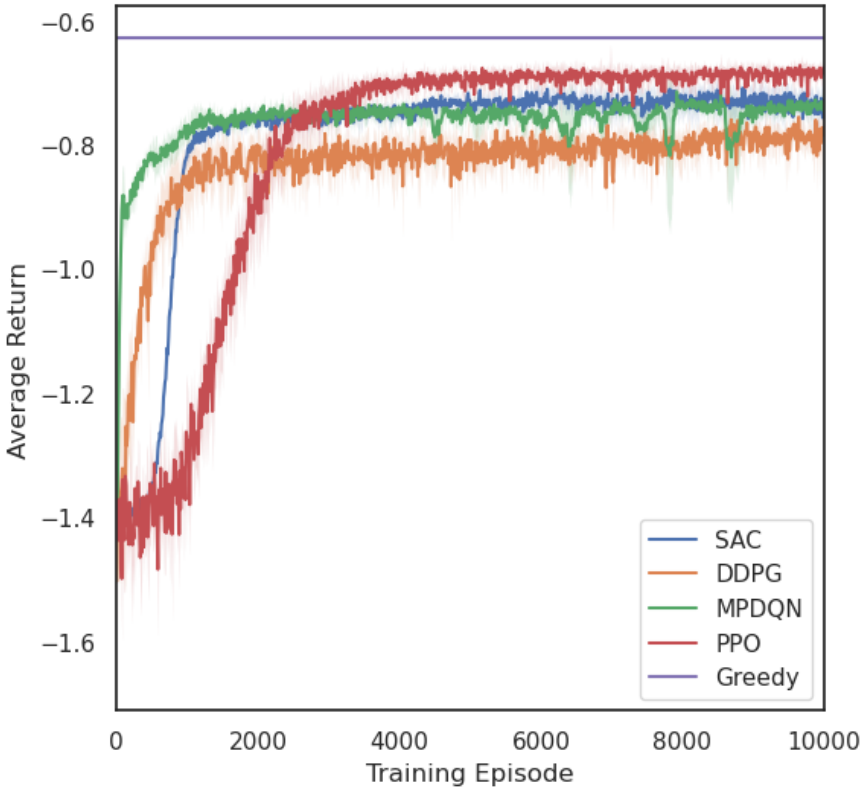
Experiments: Evaluation Metrics

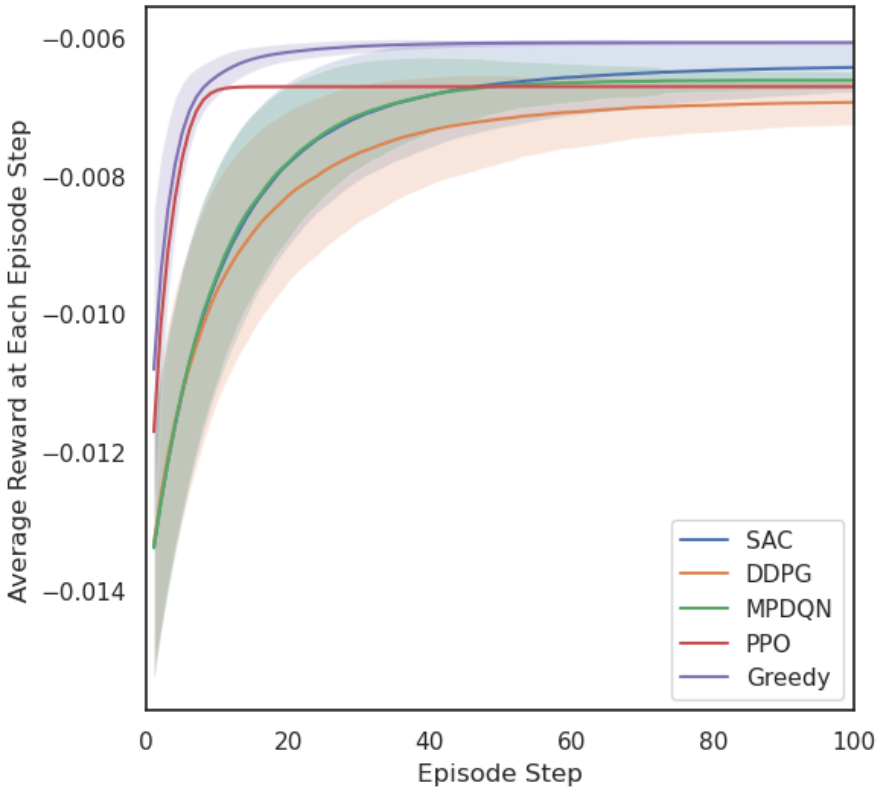
1. **Data efficiency:** How many environment interactions are required to train each agent to convergence?

2. **Final state reward:** How good is the final state achieve by each agent?

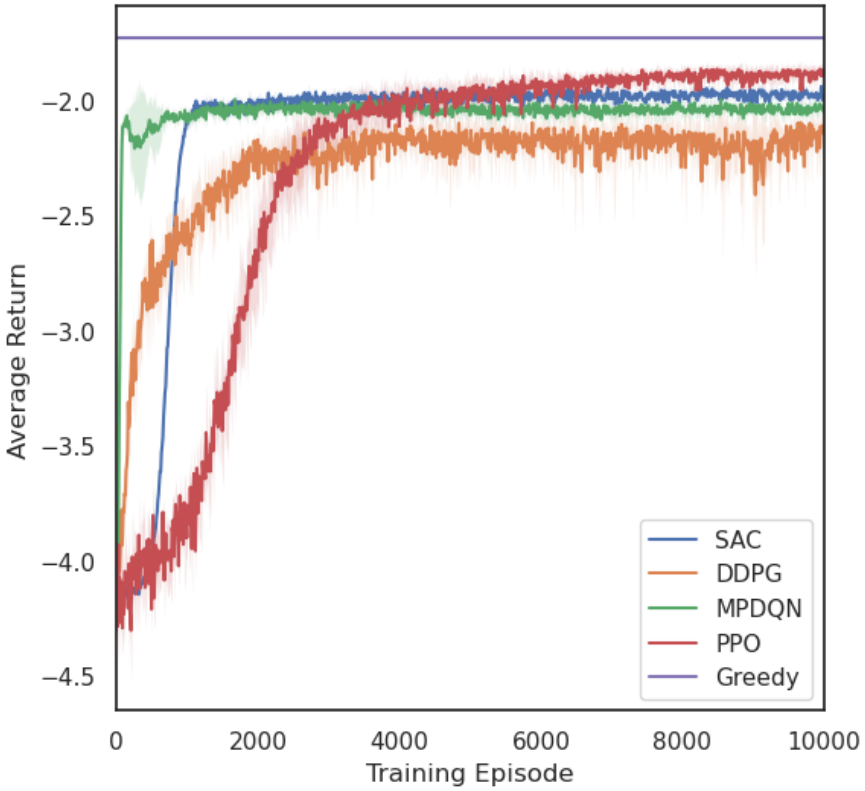
3. **Path to final state:** In a episode, how many steps does it take the agent to reach it's final solution?

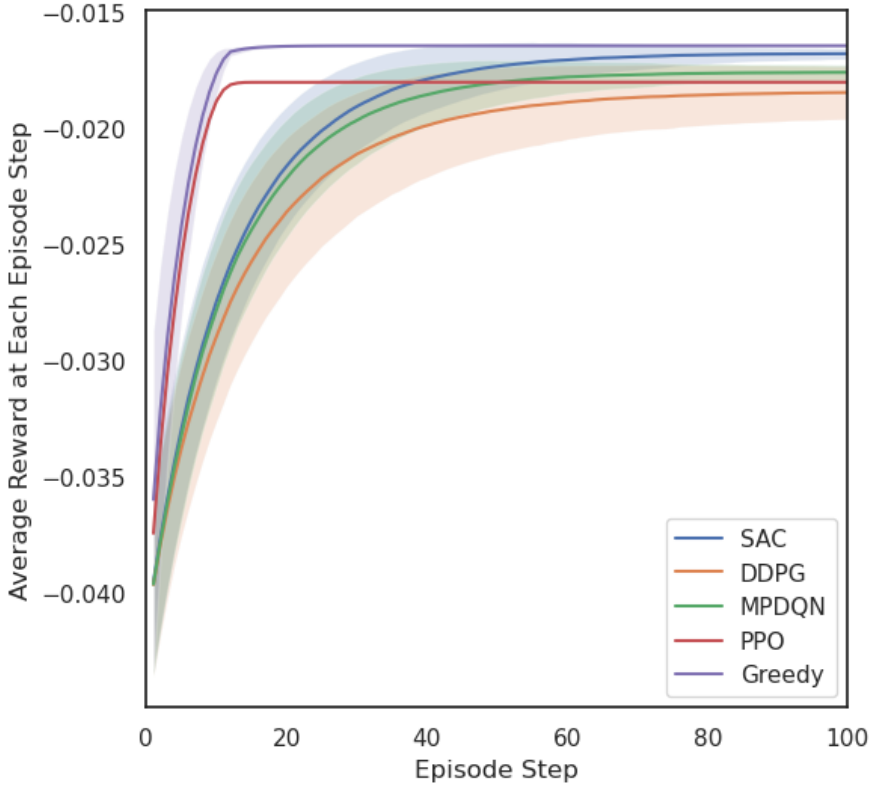
IEEE 13-bus: Balanced



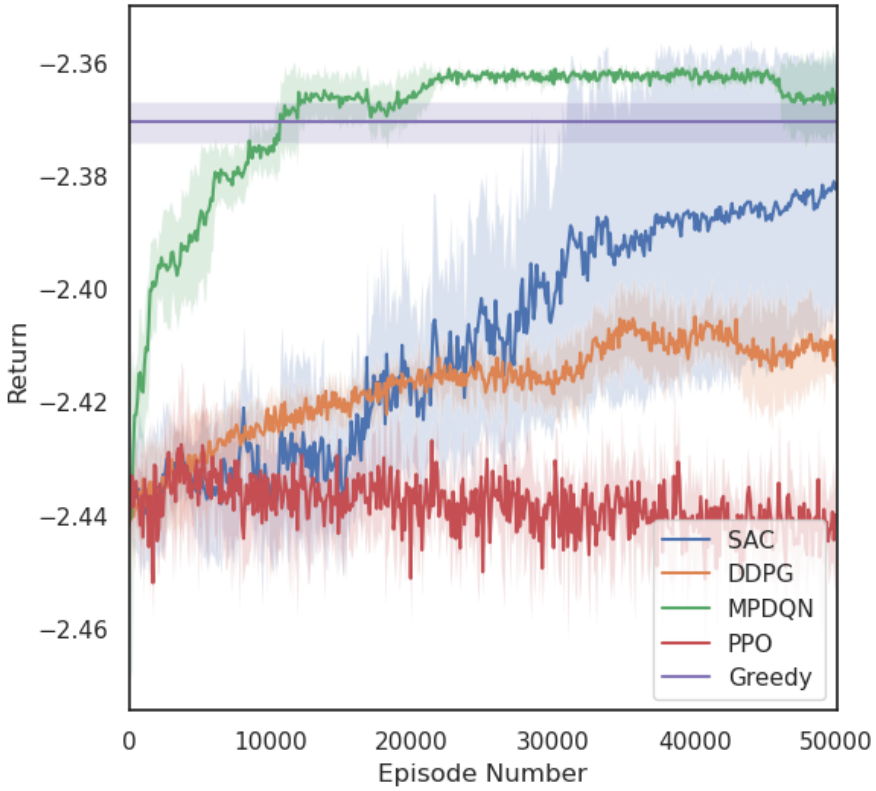


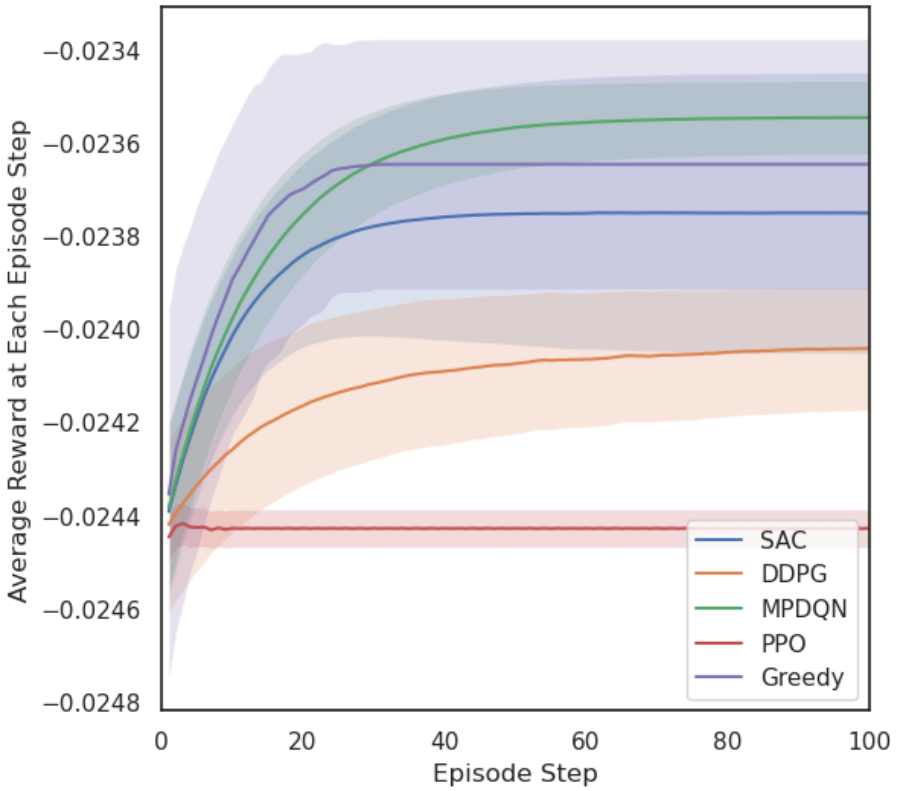
IEEE 13-bus: Unbalanced



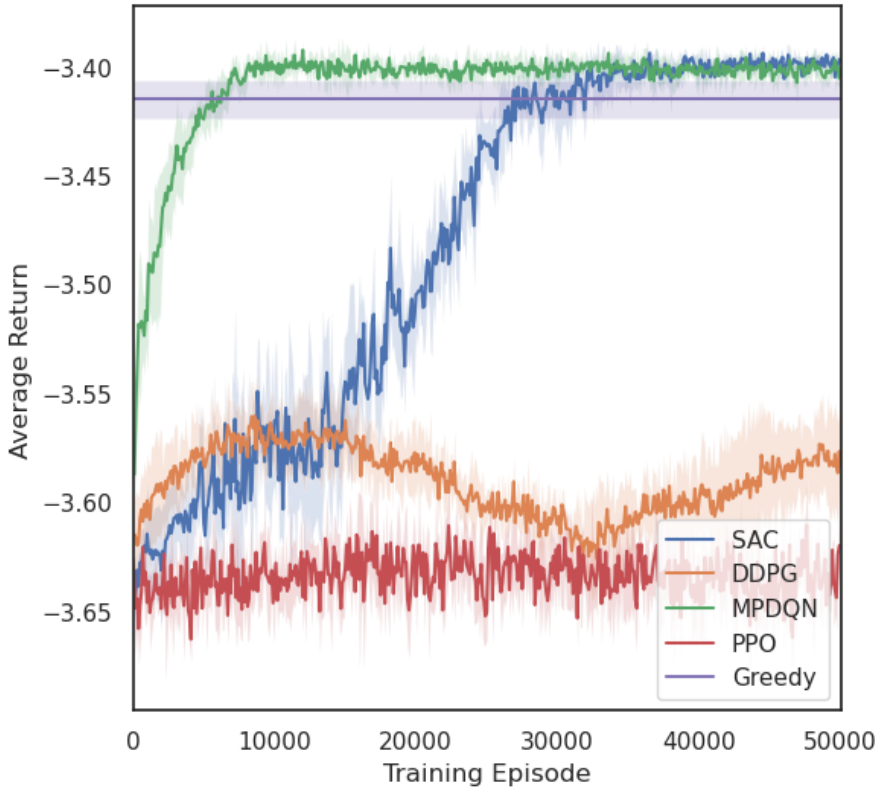


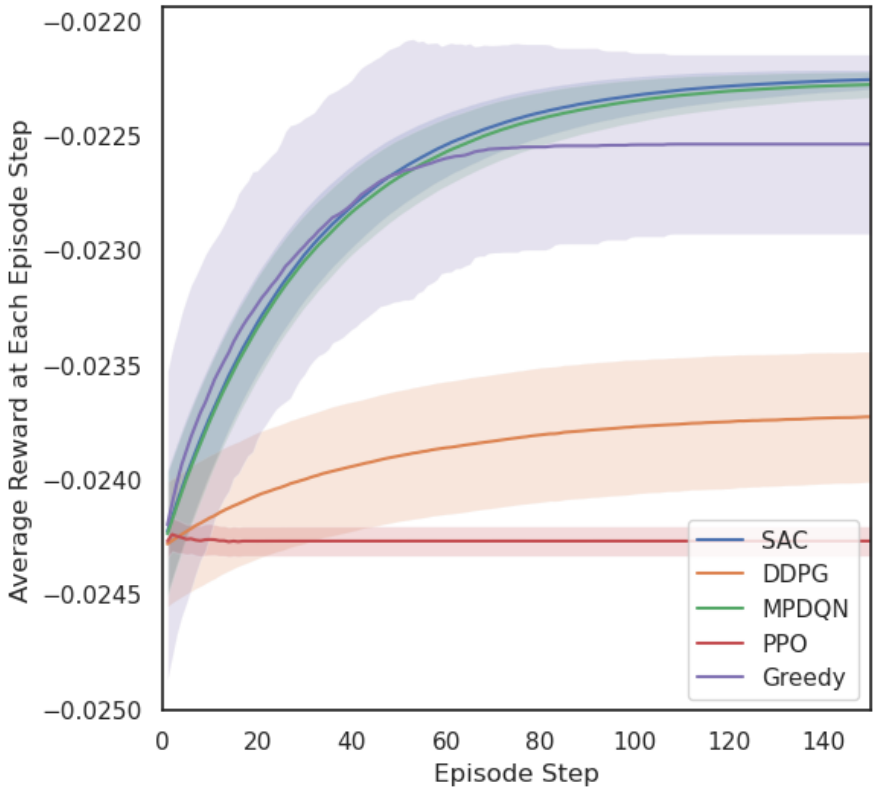
PV-14 Model


−2.36
−2.38
−2.40
−2.42
−2.44
−2.46
Return
0
10000
20000
30000
40000
50000
Episode Number
SAC
DDPG
MPDQN
PPO
Greedy


−0.0234
−0.0236
−0.0238
−0.0240
−0.0242
−0.0244
−0.0246
−0.0248
Average Reward at Each Episode Step
0
20
40
60
80
100
Episode Step
SAC
DDPG
MPDQN
PPO
Greedy

PV-14 Model


Average Return
−3.40
−3.45
−3.50
−3.55
−3.60
−3.65
0
10000
20000
30000
40000
50000
Training Episode
SAC
DDPG
MPDQN
PPO
Greedy


Average Reward at Each Episode Step
−0.0220
−0.0225
−0.0230
−0.0235
−0.0240
−0.0245
−0.0250
0
20
40
60
80
100
120
140
Episode Step
SAC
DDPG
MPDQN
PPO
Greedy

Results Summary

- MPDQN is significantly more data efficient than the other RL algorithms and finds a good final state in all tasks.
- SAC can find a slightly better solution, but requires 4x as much data
- DDPG and PPO can only stabilize the simpler IEEE model
- MPDQN and SAC can outperform the greedy algorithm on the more complex EPRI Ckt5 model

Conclusions

- DRL agents can learn to stabilize distribution power systems in a parameterized-action environment.

- This works marks an additional step towards more realistic multi-player distribution system control game which could train an agent to defend the power grid under a potential cyberattack.

- Future considerations:
  - Study larger systems
  - Study simple two-player settings