SAND2021-8185C

MLDL

Machine Learning and Deep Learning Conference 2021

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

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

- Funding Source: LDRD

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

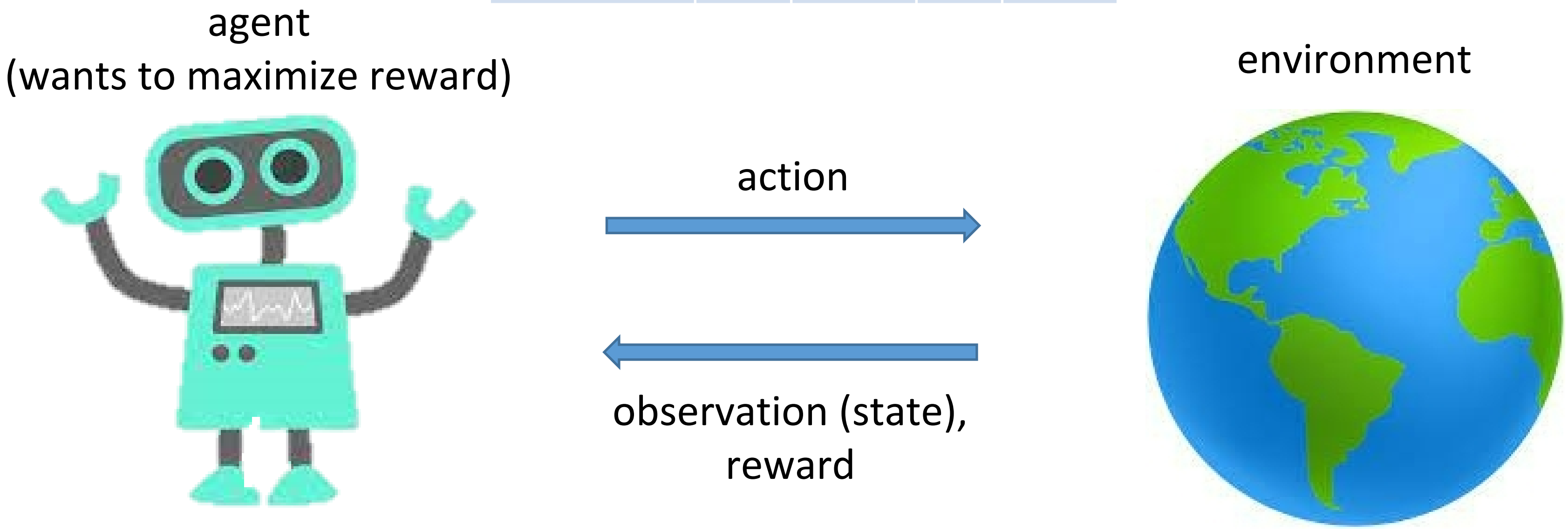
Motivation

- A trend towards decentralization in the power industry opens up new vulnerabilities.
- We investigate the possibility of defending distribution power systems using a deep reinforcement learning DRL agent who has control of a collection of utility-owned distributed energy resources (DER).
- Prior work showed that a DRL agent can learn to stabilize a modified version of the IEEE 13 -bus system with coarsely discretized bus states.
  - Discrete state and action space
- In this work, we allow the agent to set bus states to values over a continuous range.
  - Continuous state space
  - Hybrid action space (both discrete and continuous actions
- We train a DRL agent to stabilize the bus system using several continuous space algorithms and compare performance with agents trained on the discretized system.
- 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

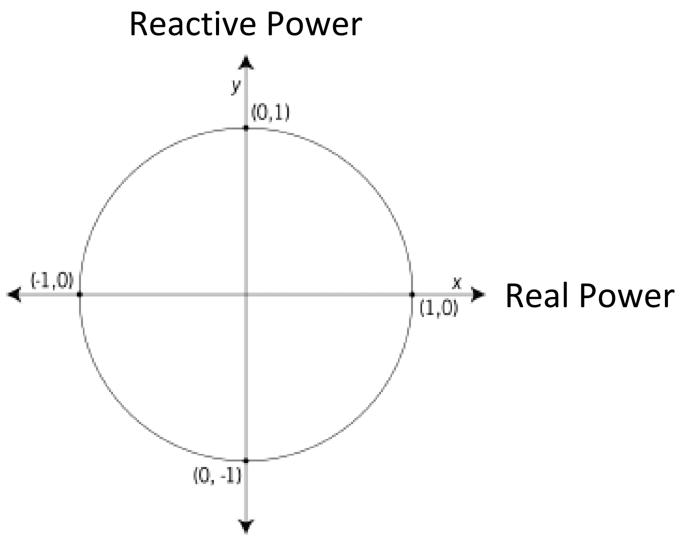


Environment

- **State space:** A list of n coordinate pairs (x, y) inside or on the unit circle
  - $n =$ number of buses
  - $x =$ active power
  - $y =$ reactive power

- **Action space:** $\{0, \dots, n-1\} \times (x, y)$, where (x, y) is inside or on the unit circle
  - e.g. (7, (0.42, -0.24)) corresponds to setting the state of bus 7 to (0.42, -0.24)
  - Only one bus can be modified per step

- **Initial state distribution:** All bus states are initialized to a point in or on the unit circle uniformly at random.



Valid bus states fall inside or on the unit circle

Environment

- **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 voltage and nominal voltage of bus i, respectively.

- **Horizon:** $T = 50$ steps
- **Objective:** Maximize the expected discounted reward

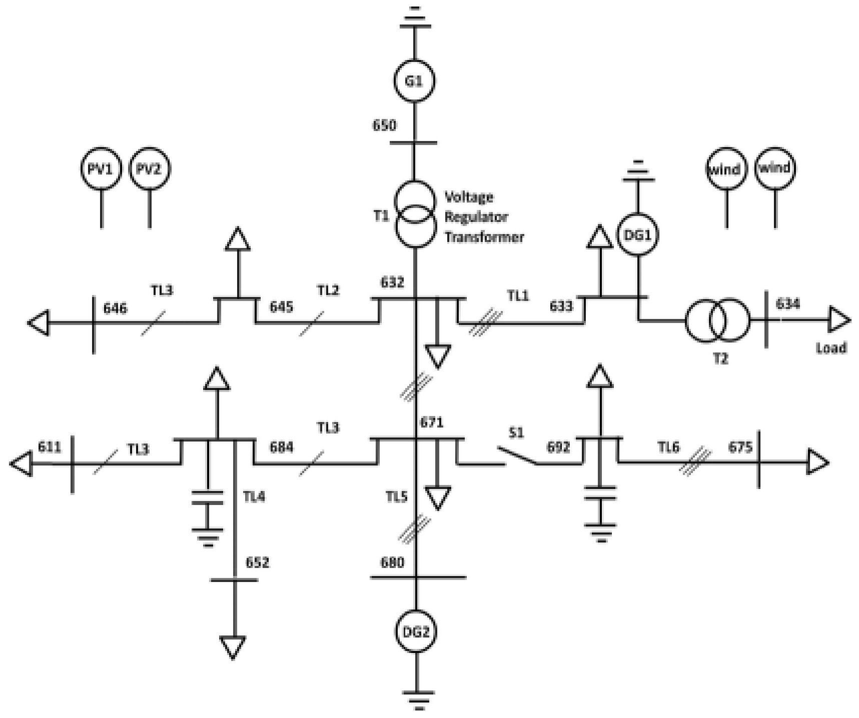
$$J = \mathrm{E}_\pi \left[\sum_{t=1}^{T} \gamma^t r_t \right]$$

where $\gamma \in (0,1)$ is a discounting factor

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

Environment

- There is voltage regulation equipment on the IEEE 13-bus model.
  - Capacitor banks provide reactive power to boost the voltage locally.
  - On-load tap changing transformers (LTCs) adjust the number of windings on the transformer to correct low/high voltages. These are non-linear devices that would make strange discontinuities in the reward surface
- The system was placed in an under-voltage state to drive the need for adjustment.
- We comment out the LTC for now.
- We use OpenDSS to simulate this model.



Previous Work

- A different subgroup trained a Deep Q-Network (DQN) agent to stabilize modified version of the IEEE 13-bus system with coarsely discretized bus states.
  - The only allowable bus states were the ‘corners’ of the unit circle:
    (0,0), (1,0), (0, 1), (-1,0), (0,-1)

- We use the DQN agent’s performance in this discrete environment as a baseline

- The state space for the continuous environment may have a different optimal configuration than the discrete environment. Since the optimal configuration in the discrete environment is a valid state in the continuous environment, we expect our agent to find a configuration that is at least as good as the configuration DQN finds in the discrete environment.

Hybrid* Action Space

- Action space contains both discrete and continuous components.

- Algorithms specifically designed for hybrid action spaces such as Multi-Pass DQN (MP-DQN) are slow and will not scale to larger systems with large discrete action components.

- Instead, we generalize policy gradient methods to hybrid action spaces using a simpler approach [https://arxiv.org/pdf/1511.04143.pdf]:
  - For discrete action: Use n output weights followed by a softmax activation and then sample from the resulting distribution.
  - For continuous action: output two continuous values – a mean and variance – for each action, and then sample from the Gaussian distribution parametrized by the outputted mean and variance.

- We apply this to Proximal Policy Optimization (PPO), Deep Deterministic Policy Gradient (DDPG), and Soft Actor-Critic (SAC). We focus on SAC.

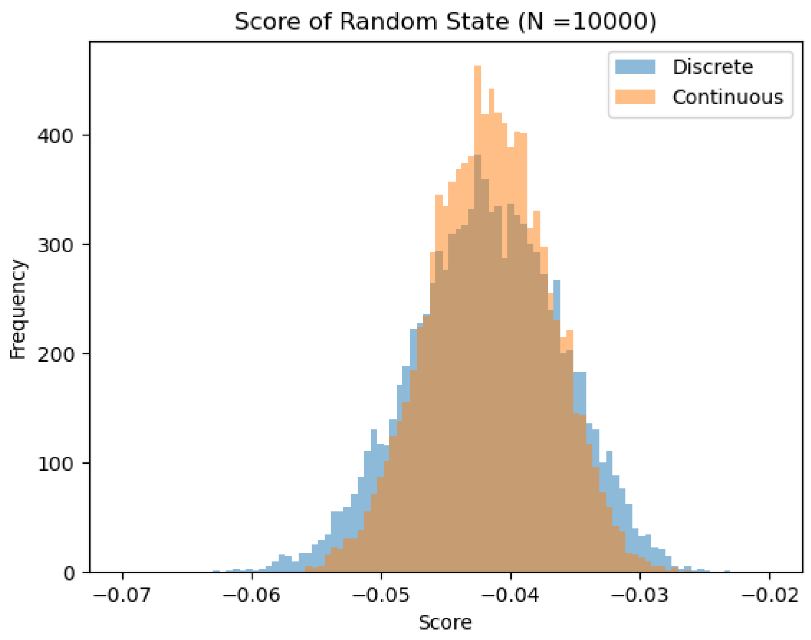
*In the RL literature, this is typically referred to as a 'parametrized action space,' where each discrete action choice is parameterized by continuous actions. We prefer the term 'hybrid' because in our setup, the discrete action is not explicitly parametrized by the continuous actions.

Experiments

- Train 10 different agents using SAC/DQN and compare their performance
  - SAC operate in continuous environment
  - DQN operate in discrete environment

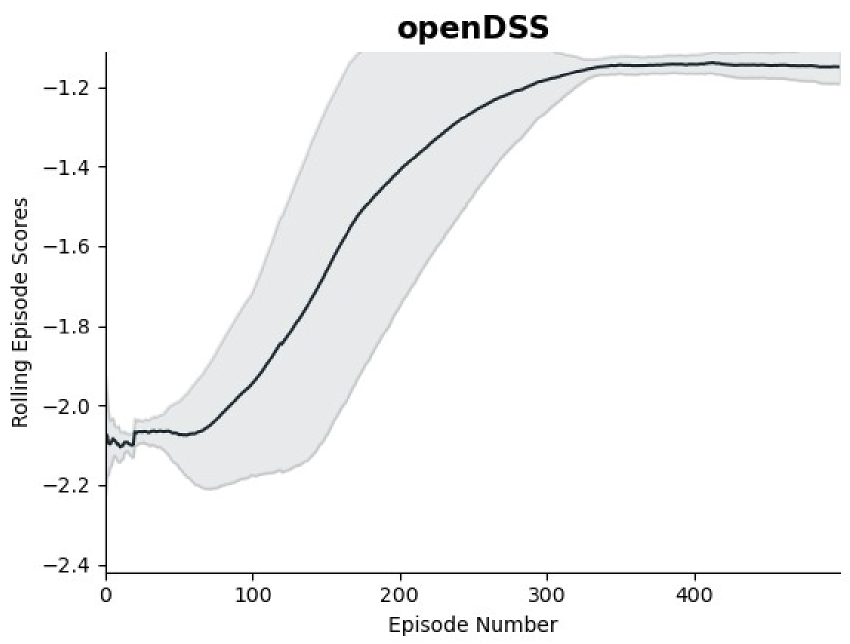
- Cannot compare scores directly
  - Starting in a more favorable state will generally result in a higher episode score
  - SAC agent has access to more states than the DQN agent

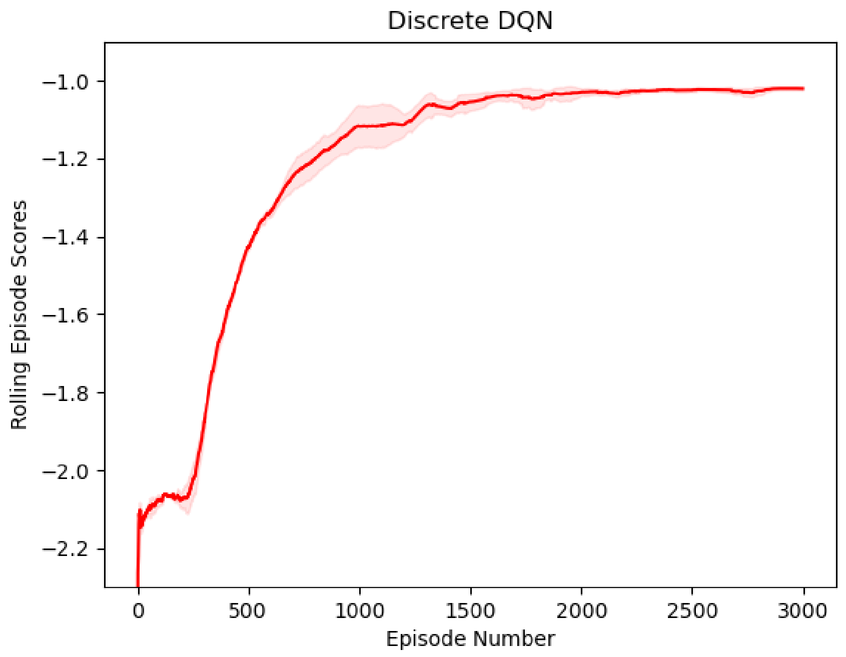
- Assess performance at test time by looking at the score of the final state of each episode.



The score of random state in the continuous environment has less variance than a random state in the discrete environment.

Experiments: Performance



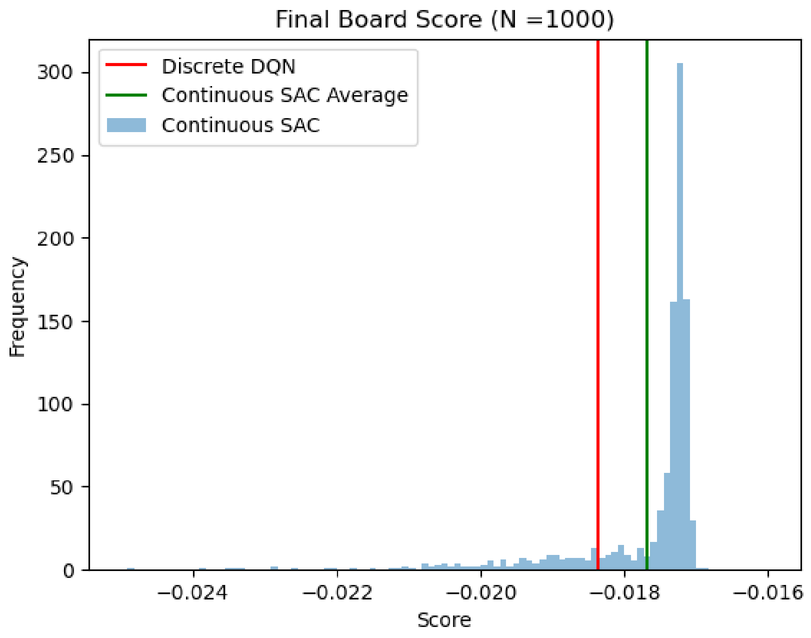


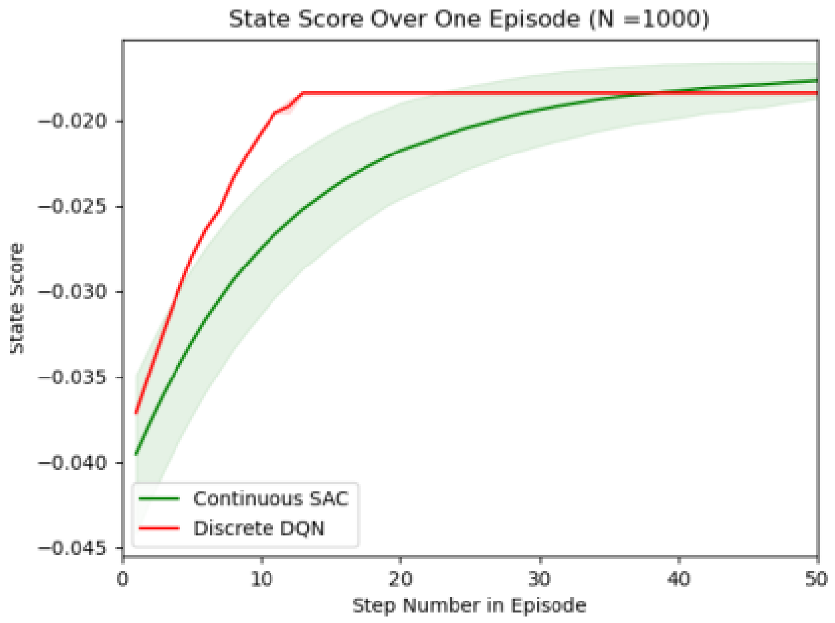
- Solid lines indicate average score over 10 runs
- Shaded belt indicates +/- one stdev

- Note the different axis scales

Experiments: Quality of Final Configurations

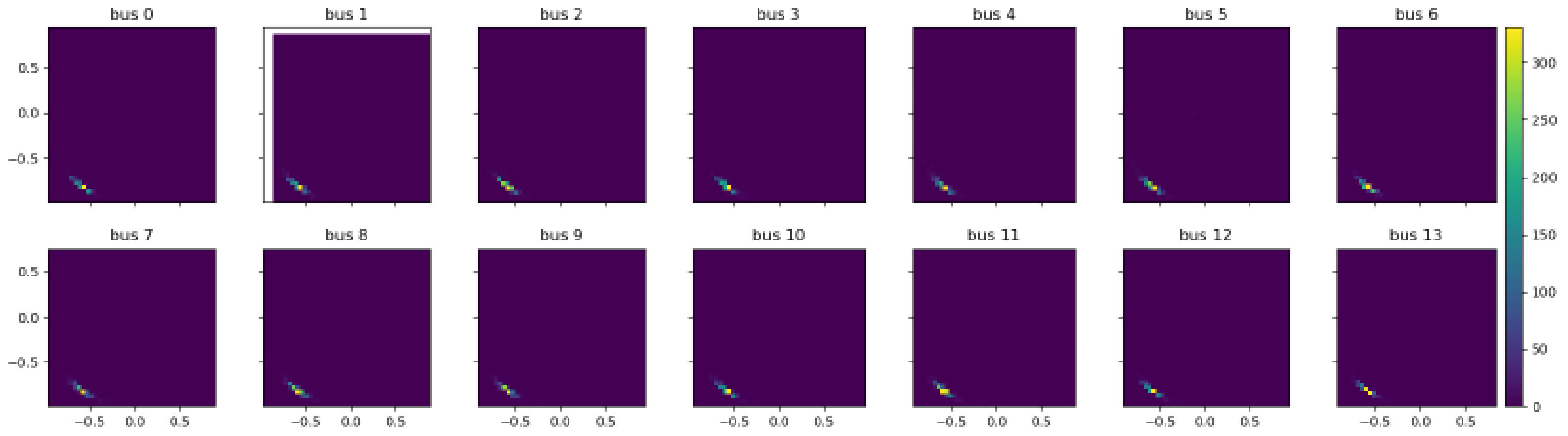
- SAC converges to a better final state, but takes more steps to reach it. Hence, SAC converges to a larger final score than DQN



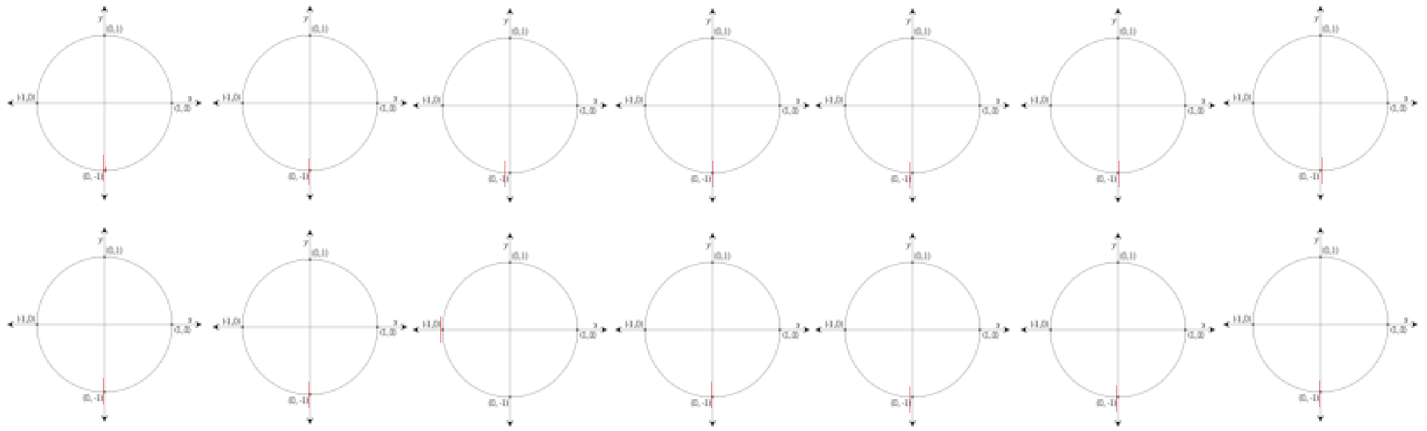


Experiments: Final Configurations

Heat map of final state configurations for SAC in continuous environment



Final state configurations for DQN in discrete environment



Horizonal Axis:
Real Power

Vertical Axis:
Reactive Power

Conclusions

- DRL agents can learn to stabilize a more realistic/complex power system environment with a continuous state space and hybrid action space.

- Another 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