Graph Convolutional Reinforcement Learning for Active Wake Control in Windfarms

Jefferson Yeh (c.yeh-4@student.tudelft.nl) • Supervisor: Greg Neustroev • Responsible Professor: Mathijs de Weerdt

TUDelft

1. Background

Windfarm and Turbine Wake Loss

Windfarms play an increasingly critical role for renewable wind energy production. But their efficiency suffer from wake induced power losses.

- Wakes (Figure 1) are a region of high turbulence and low wind speed created when wind passes through a turbine.
- Wake causes downstream turbines to suffer efficiency losses.

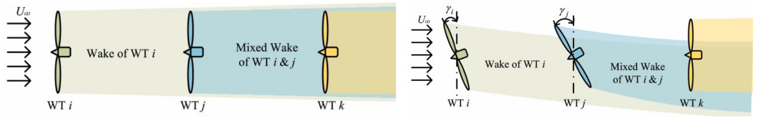


Figure 1: Turbine wake interaction, image adapted from Dong [1]

Active Wake Control (AWC)

Wake losses can be reduced by yawing upstream turbines to steer the wake away from downstream turbines. Energy production gains in downstream turbines can outweigh the losses in the upstream turbine. AWC is the active yawing of turbines (Figure 1b). Existing methods include:

- Model-based classical control → over reliant on accurate system model.
- Singe Agent Reinforcement Learning (SARL) → windfarm size scalability issues due to combinatorial explosion.

Multi-Agent Reinforcement Learning (MARL)

- One turbine per agent → no combinatorial explosion.
- Fully cooperative, shared reward (windfarm output).

Paper Focus: *Graph Convolutional Reinforcement Learning (DGN) [2]*

- Agent graph representation, promote collaboration between agent and neighbors → Windfarm and turbines naturally topological .
- Windfarm representation have many modelling approaches and information encapsulation.

2. Research Questions

How can DGN be efficiently applied to the Active Wake Control of windfarms?

1. How does the learning rate influence the performance of DGN when applied to AWC of windfarms?
2. How does the performance of DGN scale with increasing windfarm sizes?
3. How can the modelling of directionality in windfarm topological features be exploited by DGN to improve the windfarm performance?

3. Methodology

DGN Architecture (Figure 2)

- Encoder Layer encodes agent state into features.
- Convolutional Layers learn latent features in the interactions between encoded agent features.
- Q Network learns Q function for agent.
- Each agent has own set of above layers.

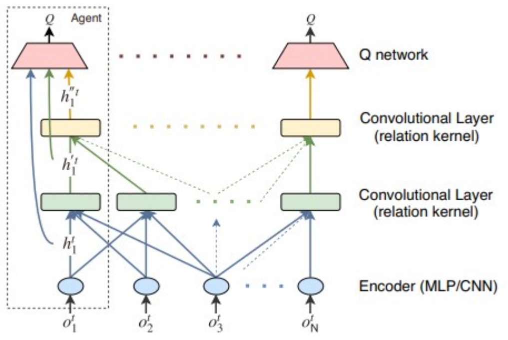


Figure 2: DGN architecture, image from Jiang [2]

Windfarm Graph Representation

Using edge directionality to model inter-agent information flow. Turbines sufficiently spaced at 10 rotor diameter lengths apart.

Undirected (DGN):

- Bidirectional communication
- All information shared

Directed Upstream (DGN-U):

- Upstream communication
- Agent learns about turbines it affects.

Directed Downstream (DGN-D):

- Downstream communication
- Agent learns about turbines that affect it.

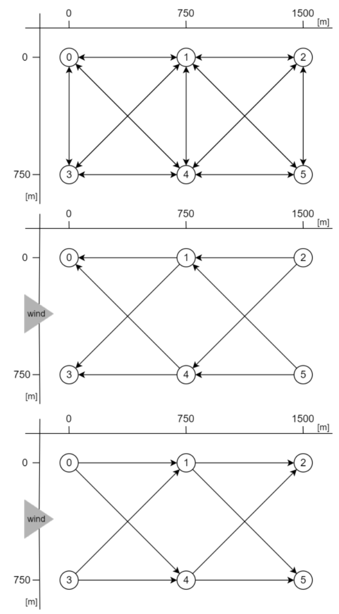


Figure 3: Undirected, upstream directed and downstream directed windfarm

Baselines

Use to benchmark DGN performance

FLORIS [3]

- Numerical gradient based solver.
- Near optimal solution.

Static Yaw

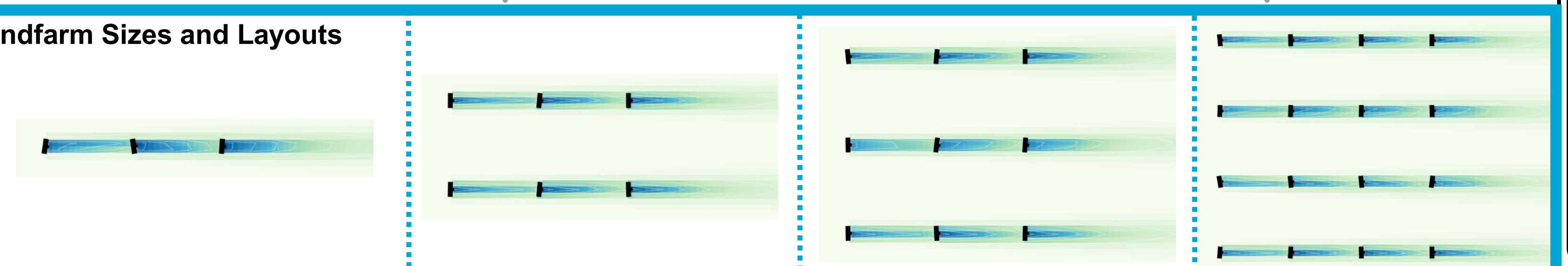
- All turbines face wind.
- No AWC

Deep Q-Network (DQN) [4]

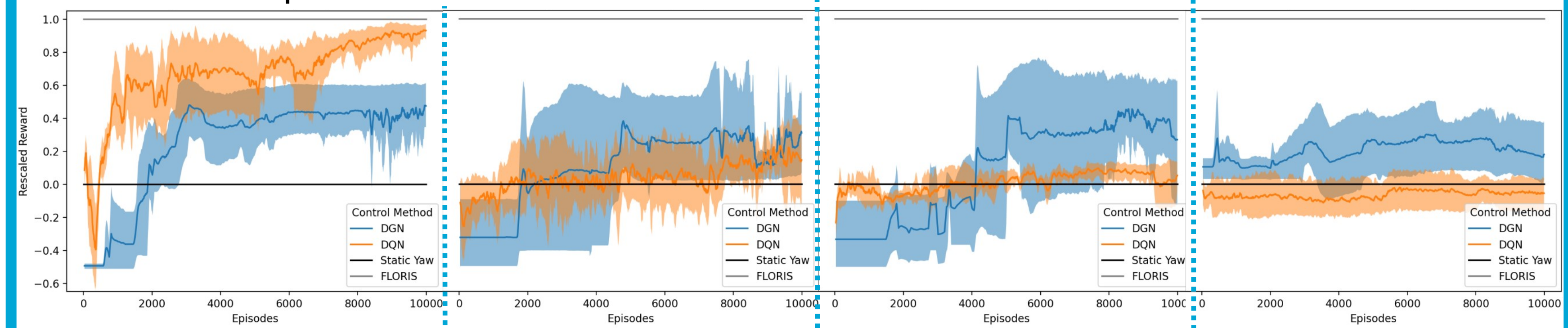
- Q-network based SARL algorithm.

4. Results

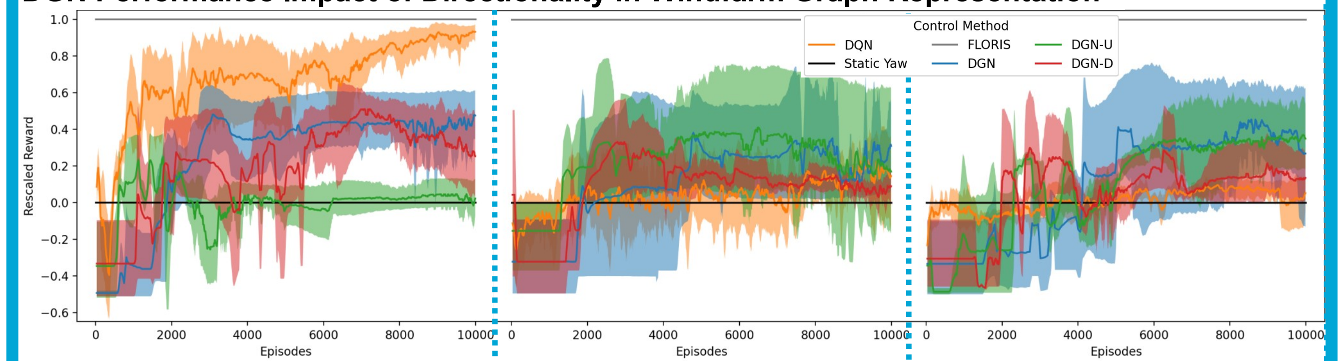
Windfarm Sizes and Layouts



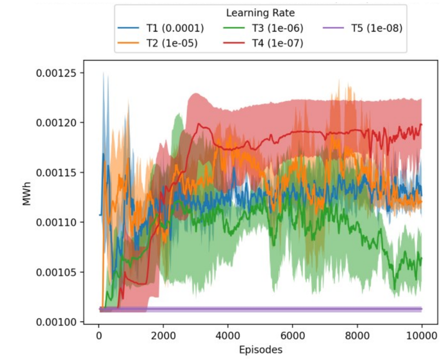
DGN Performance Impact of Windfarm Size



DGN Performance Impact of Directionality in Windfarm Graph Representation



Performance Impact of Learning Rate



5. Conclusions and Future Works

Learning Rate (LR) Experiments:

- Large LR over and undershoot gradient preventing DGN from converging.
- Decreasing LR improves learning stability.
- Small LR converge to the immediate minimum, preventing exploration.

Windfarm Size Experiments:

- DGN performance stable as size increase.
- DQN performance degrades as size increase.
- MARL advantageous over SARL for AWC

Graph Directionality Experiment:

- Upstream better than Downstream for large windfarms and vice versa. Further experimentation required to investigate contradiction.
- Directed and Undirected yield comparable performance at each tested size. Additional information transfer in DGN not more useful.

DGN is applicable for AWC to learn useful policies, especially at scaled windfarms.

- Varied action space exploration methodology and explore DGN hyperparameters.
- Explore real windfarm layouts and stochastic wind processes.
- Further investigate directionality modelling and dynamic updating of graphs.

References

[1] H. Dong, J. Zhang, and X. Zhao, "Intelligent wind farm control via deep reinforcement learning and high-fidelity simulations," Applied Energy, vol. 292, p. 116928, 6 2021.

[2] J. Jiang, C. Dun, T. Huang, and Z. Lu, "Graph convolutional reinforcement learning," 8th International Conference on Learning Representations, ICLR 2020, 2020.

[3] P. A. Fleming, A. P. J. Stanley, C. J. Bay, J. King, E. Simley, B. M. Doekemeijer, and R. Mudafort, "Serial-refine method for fast wake-steering yaw optimization," Journal of Physics: Conference Series, vol. 2265, p. 032109, 5 2022.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," arXiv, 2013.