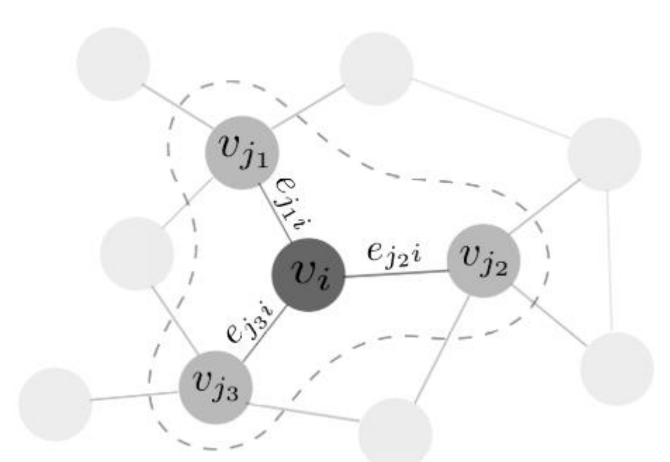
Control and Coordination of Multi-Drone Systems Using Graph Neural Networks

Parth Khopkar, Computer Science
Mentor: Heni Ben Amor, Assistant Professor
School of Computing, Informatics and Decision Systems Engineering

Motivation

Multi-robot systems show great promise in performing complex tasks in areas ranging from search and rescue to interplanetary exploration. Yet controlling and coordinating the behaviors of these robots effectively is an open research problem. We investigate techniques to control a multi-drone system where the drones learn to act in a physics-based simulator using demonstrations from artificially generated motion data that simulate flocking behavior in biological swarms. The Graph Neural Network (GNN) controller[1] used for the drones is able to learn an efficient representation of low-level interactions in the system, boosting prediction abilities.

Graph Neural Networks



$$e_{ji}^t = \phi^e(v_j^t, v_i^t)$$
$$\bar{e}_i^t = \sum_{j \in \mathcal{N}_i} e_{ji}^t$$
$$h_i^t = \phi^v(x_i^t, \bar{e}_i^t)$$
$$y_i^t = \psi(h_i^t)$$

Fig. 1 Operations in Graph Neural Networks (GNNs)

Graph Neural Networks[2] are a class of function approximators that operate on graph-based data structures while explicitly modeling the interaction dynamics of the entities present in the graphs. In the context of multi-agent systems, graphs can be used to model the connections between the agents and other entities present in the environment.

Methodology

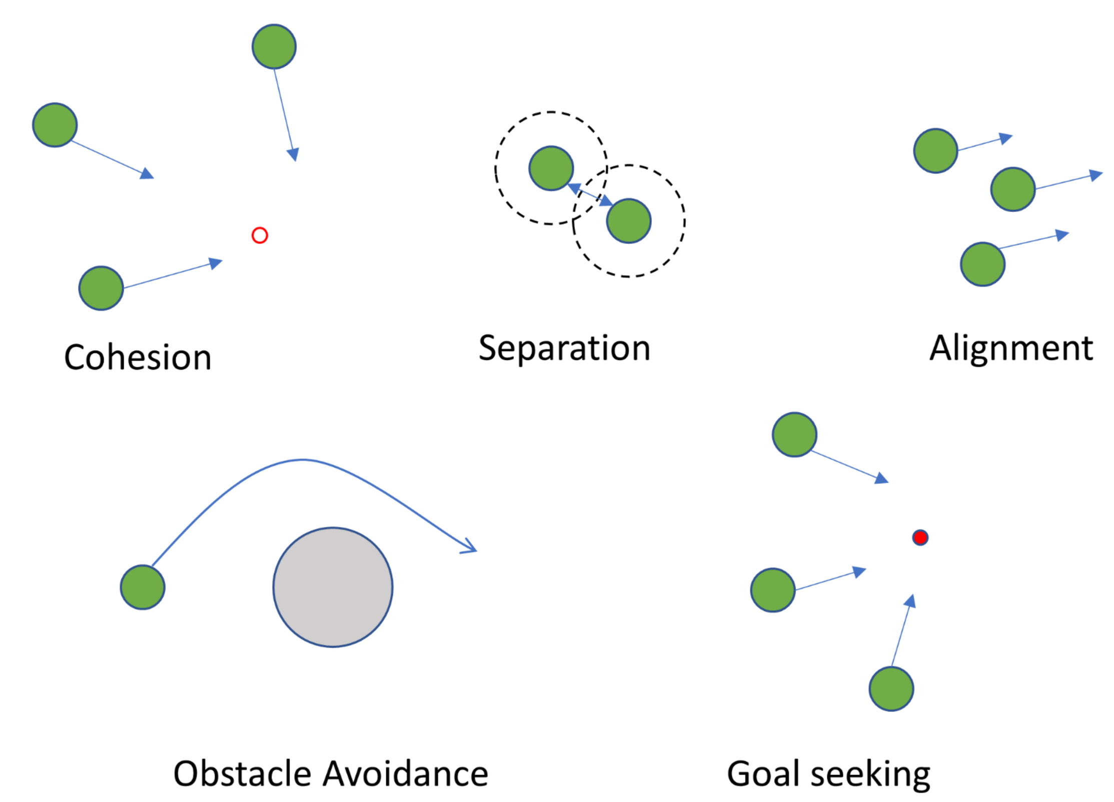


Fig. 2 Different behaviors in the Boids model that Determine motion patterns of the boids.

Imitation Learning using Boids model

The Boids model[3] simulates the aggregate motion of a flock of animals where the behavior of each member of the flock is the result of interactions with their neighbors. The final behavior of the system can be described as emergent, which means that the behavior of the entire system depends on the relationships each member of the system has with each other rather than their individual properties. We use demonstrations generated using this model to train our Graph Neural Network controller whose output is provided to a PID controller that controls the drones in a PyBullet physics-based simulator..

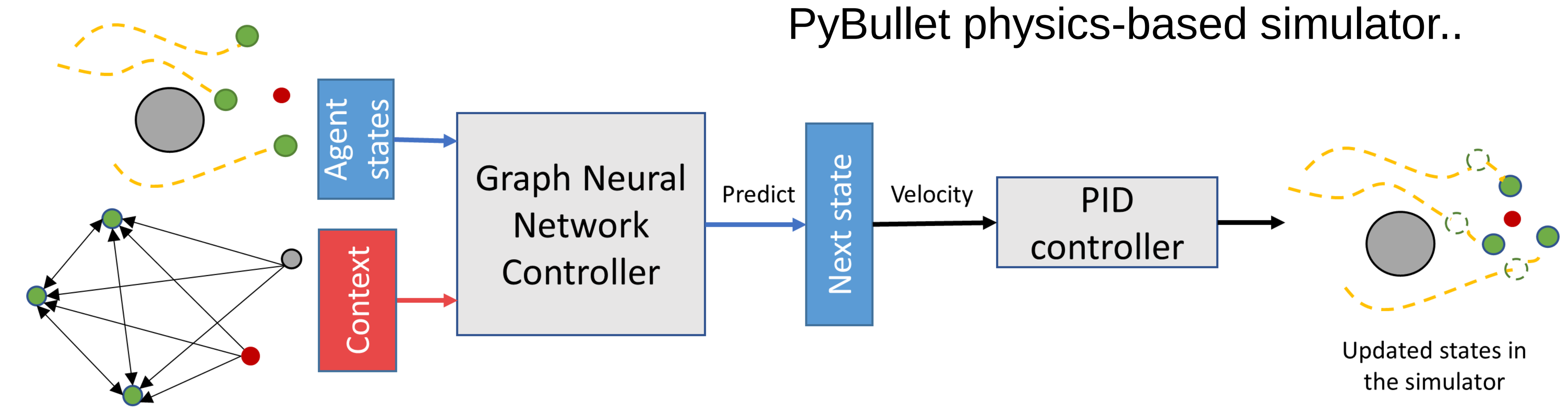
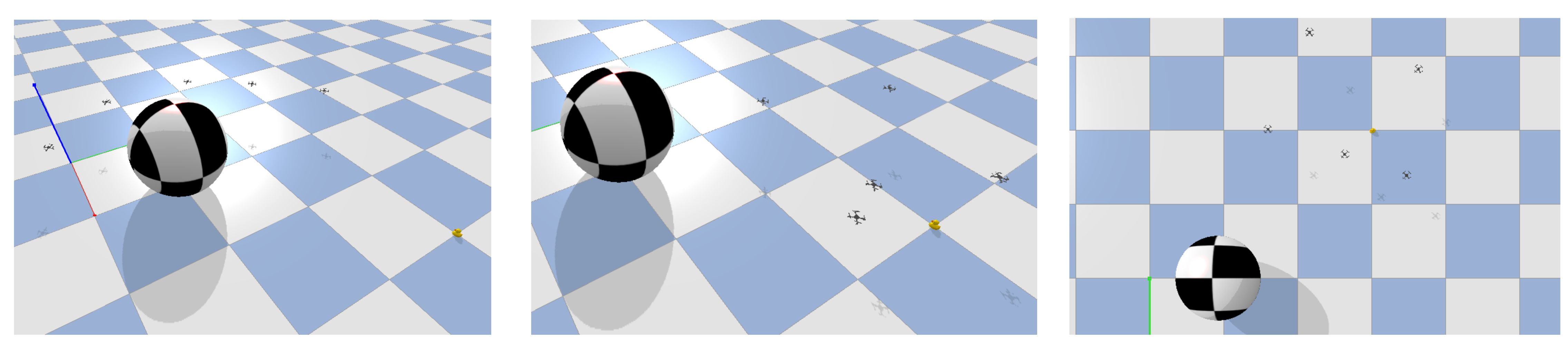


Fig. 3 System Architecture of the controller



(a) (b) (c)

Fig. 4 The drones flying in the PyBullet based simulator. (a) The drones maneuver to avoid obstacles (b) The drones reach the goal and hover near it (c) Top view of the drones hovering near the goal

Results

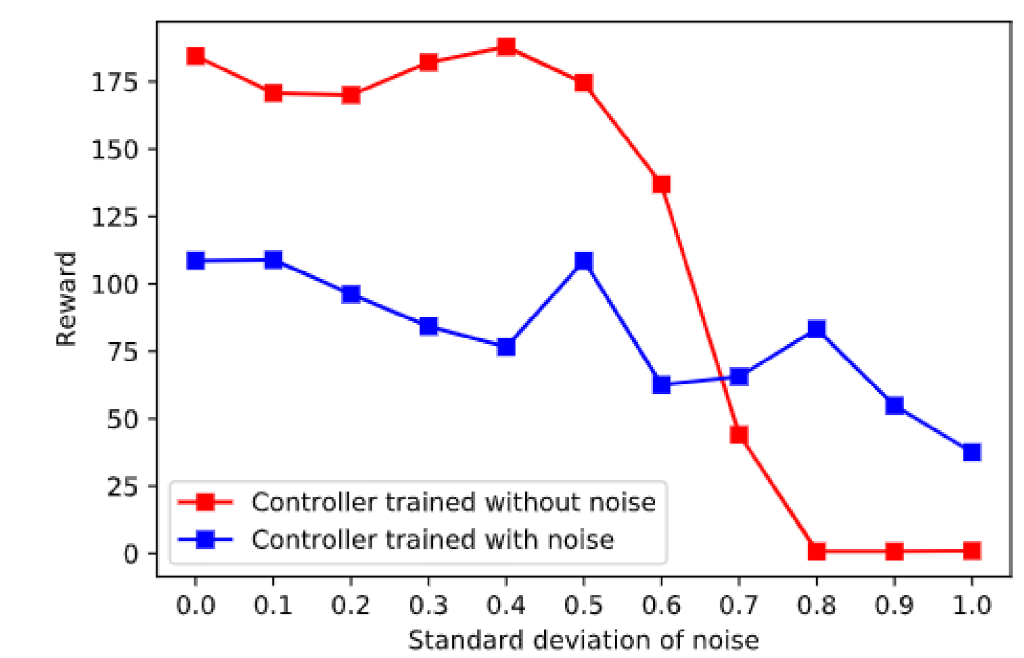


Fig. 5 Results on robustness of the controller

In order to test the robustness of the controller, it was tested on noisy observations. We found that training the controller on noisy data can prevent collisions even at higher noise and allow for a more graceful performance degradation.

Conclusion and Future Work

We showed that GNNs trained on expert demonstrations can be used to leverage the problem structure to effectively control and coordinate multiple drones in a Physics based simulator. In the future we would like to implement the controller on real drone swarms to test the effectiveness of the proposed method for tasks such as search and rescue.

References

[1] Zhou, Siyu, et al. "Clone Swarms: Learning to Predict and Control Multi-Robot Systems by Imitation." *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2019*. Institute of Electrical and Electronics Engineers Inc., 2019.
[2] Battaglia, Peter W., et al. "Relational inductive biases, deep learning, and graph networks." arXiv preprint arXiv:1806.01261 (2018).
[3] Craig W. Reynolds. 1987. Flocks, herds and schools: A distributed behavioral model. SIGGRAPH Comput. Graph. 21, 4 (July 1987), 25–34.