Area Coverage with Unmanned Aerial Vehicles Using Reinforcement Learning

Charles Zhang, Elisabeth Landgren, Frank Zhang, Aaron Gould, and Prof. Esra Kadioglu

Math, Statistics, and Computer Science - Macalester College – St. Paul, MN

Introduction

- Area coverage is a common task that can be used for things such as surveying powerlines, inspecting crops, and disinfecting large areas such as stadiums
- Deploying Unmanned Aerial Vehicles (UAVs), or drones, with cameras to complete these tasks can reduce cost and time significantly
- Human UAV pilots vary in ability to cover the area effectively and can be expensive to hire
- Mathematical pathfinding calculations are cumbersome and cannot effectively handle multiple drones
- Reinforcement learning can be used to train agents to find coverage paths

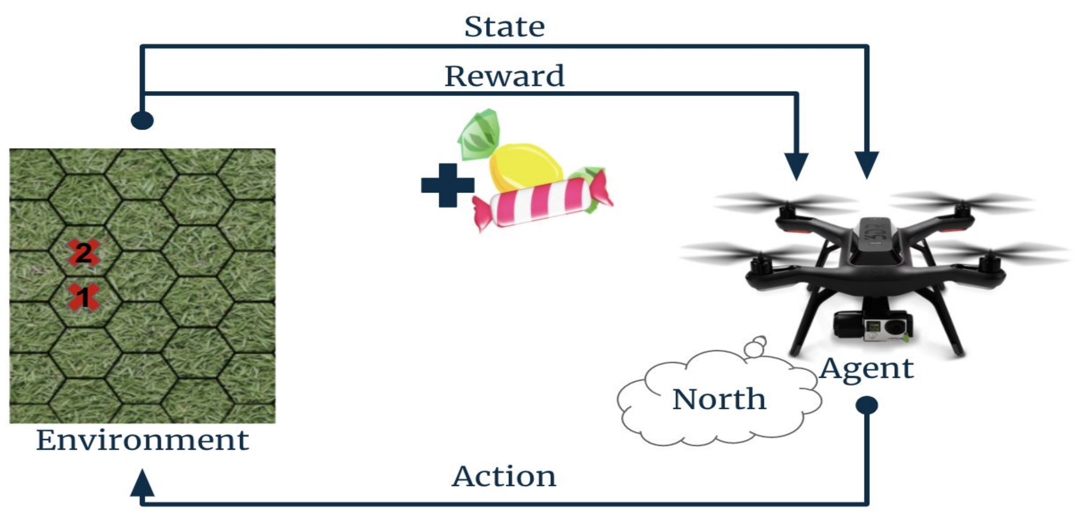


Illustration of the agent environment relationship in reinforcement learning

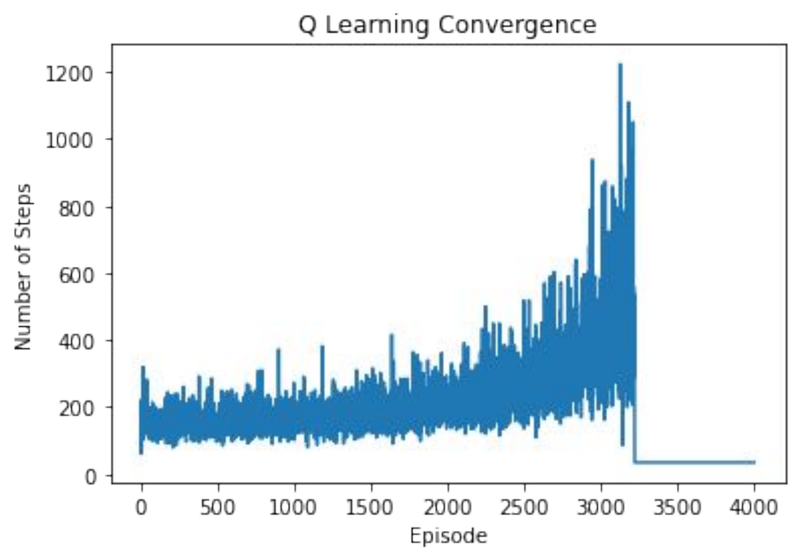
- Agents learn by interacting with the environment and receiving feedback about the current state and chosen action.
- Sign and size of rewards given impact the probability of future actions

Toolbox

Python 3: Flexible coding language with many helpful reinforcement learning toolkits
OpenAI gym: A python library that is used for comparing different reinforcement learning algorithms. This is helpful for creating a standardized custom environment
Stable Baselines: A python toolkit that allows usage of tested agents for a large range of reinforcement algorithms using the gym interface

Single Agent Coverage

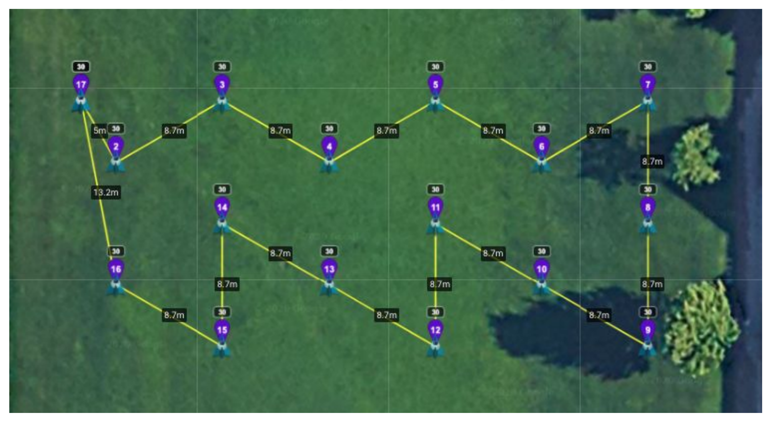
- Using a hexagonal tessellation and a hamiltonian path we were able to decrease the number of waypoints in the path
- By assigning a Q-value to each cell, the agent was able to learn using tabular Q-learning



Sample learning rate for a single agent, allowing revisits on a hexagonal tessellation

Flight

- We created an algorithm that uses the path the agent learned and translates it to GPS coordinates that a drone could follow. Here we see an example of the path a drone could take on Shaw field using the fly litchi website.



Single agent path for hexagonal tessellation shown on Shaw field

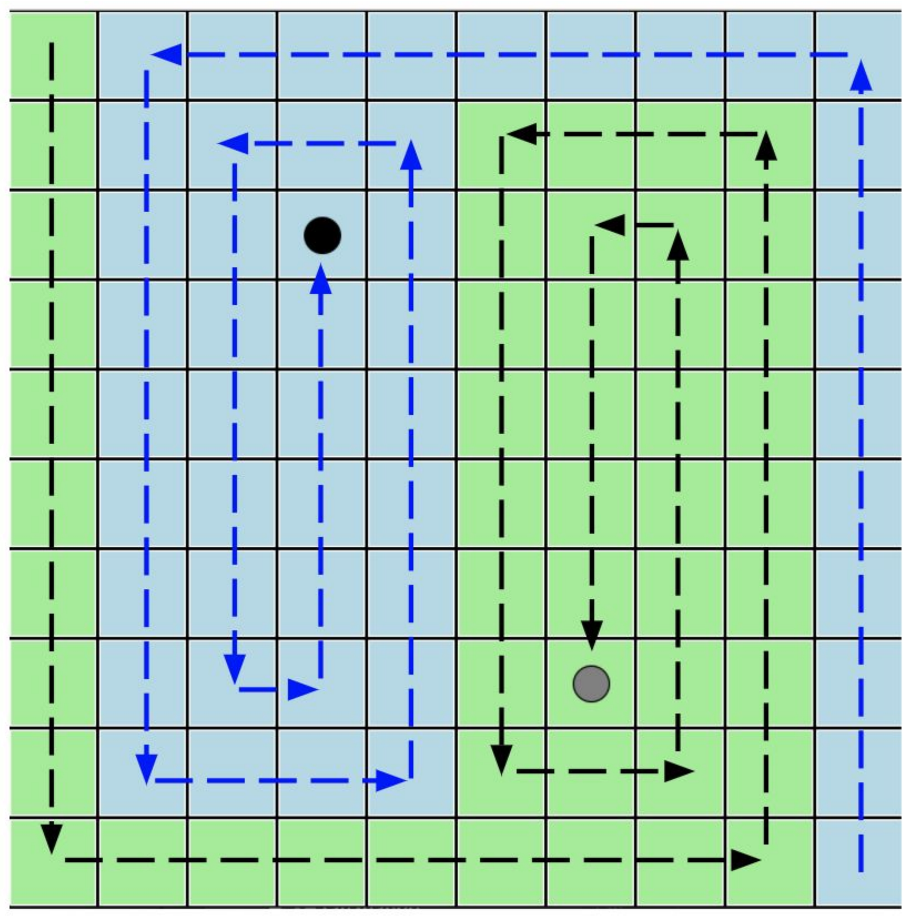
Future Work

Environment size is a limiting factor for the Reinforcement learning we applied. Working on understanding and analyzing why this is a limiting factor and making an in depth comparison of the mathematical path finding approach compared to the reinforcement learning approach would be helpful in understanding the use cases for each of these approaches better.

Multi Agent Coverage

We implemented a custom gym environment for coverage path planning for multiple drones. We imagine the multi-drones as a single agent such as they move together at every step:

- **Observation Space:** A 1D array of the coordinate of all agent;
- **Action Space:** At each step the action is a permutation of all agent's possible move, so each agent makes a move per step, which guarantees all drones cover the same amount of field;
- **Reward:** Give a small positive reward if all agents visit a new grid at a step, and give a huge reward and end the episode if the field is covered. If an agent visits a visited field, terminate the algorithm and gives no reward at the step.
- **Result:** We tested out a 10x10 environment for 2 drones using Actor Critic using Kronecker-Factored Trust Region (ACKTR) provided by stable baseline. The algorithms can find coverage paths after about 25,000 episode training.



Dual agent coverage path on a 10x10 square tessellation

References

1.Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning : an Introduction* (Second, Ser. Adaptive computation and machine learning). MIT Press.
2.Lapan, M. (2020). *Deep reinforcement learning hands-on : apply modern rl methods to practical problems of chatbots, robotics, discrete optimization, web automation, and more* (Second). Packt Publishing.
3.Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J. & Zaremba, W. (2016). OpenAI Gym (cite arxiv:1606.01540)
4.Hill, Ashley and Raffin, Antonin and Ernestus, Maximilian and Gleave, Adam and Kanervisto, Anssi and Traore, Rene and Dhariwal, Prafulla and Hesse, Christopher and Klimov, Oleg and Nichol, Alex and Plappert, Matthias and Radford, Alec and Schulman, John and Sidor, Szymon and Wu, Yuhuai (2018). Stable Baselines. GitHub repository. https://github.com/hill-a/stable-baselines

Our Code and Animations

Scan here to view our github repository and see some animations of the path coverage

Acknowledgements

Special thanks to Clare Boothe Luce Foundation and Macalester College CSR for funding of this research