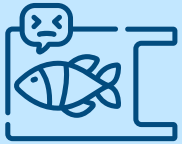
Aquarium: środowisko MARL w PettingZoo

Wykonał Bartłomiej Tarcholik

Uczenie przez wzmacnianie kontra nadzorowane



Uczenie przez wzmacnianie	Uczenie nadzorowane
Nauka na własnych doświadczeniach (metoda prób i błędów) ocenianych nagrodą od środowiska	Nauka na podanym zestawie danych i ich prawidłowych wyników
Decyzja opiera się na aktualnym stanie i poprzednich doświadczeniach, nieraz zależnych od siebie	Decyzja oparta na nauczonej bazie danych
Decyzje zależne od siebie i prowadzące jedna do drugiej	Decyzje niezależne od siebie nawzajem

Gym, Gymnasium, PettingZoo

- Są to ustandaryzowane środowiska bazujące na OpenAI Gym, które symulują najczęściej różnego rodzaju gry lub środowiska
- W przypadku Gym oraz Gymnasium występuje zawsze 1 agent, PettingZoo jest środowiskiel MARL – Multi-Agent Reinforcement Learning
- Oferują podobne do siebie metody interakcji ze środowiskiem przez agentów oraz zwracają podobnego typu nagrody
- W przypadku PettingZoo jest możliwość zdefiniowania środowiska zarówno równoległego, jak i sekwencyjnego

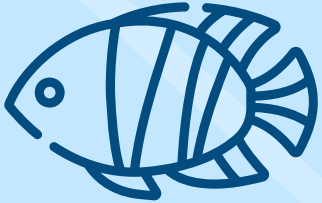
Założenia projektu

Założenia przy rozpoczęciu pracy:

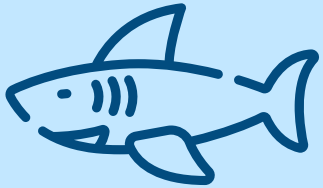
1. Środowisko oparte o mapę
2. Przynajmniej jeden typ agenta, który wchodzi w interakcję ze środowiskiem
3. Przestrzeń obserwacji agenta oparta o jego wzrok
4. Nagrody oparte o pożywienie oraz śmierć
5. Stworzenie sieci DQN rozwiązującej problem

Co udało się zrealizować:

1. Środowisko na mapie akwarium 32x32
2. Dwa typy agentów (rekin, ryba) mające podobny sposób działania, lecz inne pożywienie
3. Przestrzeń obserwacji agenta oparta o widzenie kątowe
4. Nagrody są aktualnie problemem
5. Próby stworzenia sieci zostały zablokowane przez problemy z uczeniem



Polityka nagród



Ryba:

- Wysoka nagroda za zjedzenie pożywienia
- Kara wprost proporcjonalna do straconego pożywienia
- Żywi się roślinami, które pojawiają się na planszy
- Większy koszt szybkiego ruchu
- Kara za śmierć lub bycie ugryzionym

Rekin:

- Wysoka nagroda za zjedzenie ryby
- Kara wprost proporcjonalna do straconego pożywienia
- Żywi się innymi rybami (nie rekinami)
- Mniejszy koszt szybkiego ruchu
- Kara za śmierć

Przestrzeń akcji i obserwacji

Dostępne akcje:

- Spoczynek (-1 food)
- Ruch o 1 kratkę (-2 food)
- Szybki ruch o 2 kratki (-5/6 food)
- Żywienie (ryba roślinami, rekin rybami)

W sumie przestrzeń akcji składała się z 10 możliwych akcji

Przestrzeń obserwacji:

1. Pierwsza próba – przekazanie mapy (2D array) widocznej z punktu widzenia agenta gdzie liczba oznaczała co jest na polu, lub -1 oznaczające bycie pola poza wzrokiem agenta
2. Aktualna próba – jaki obiekt jest widoczny pod konkretnym kątem (1D array)
3. Przy obu opcjach: pozycja X oraz Y agenta oraz aktualna zawartość żołądka

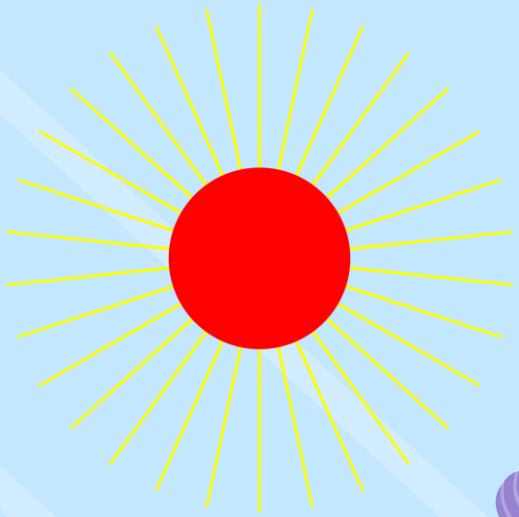
Obserwacje kątowe

Obserwacje produkują wektor n-elementowy, w którym co każde 3 stopnie jest przekazywany obiekt, który widzi agent.

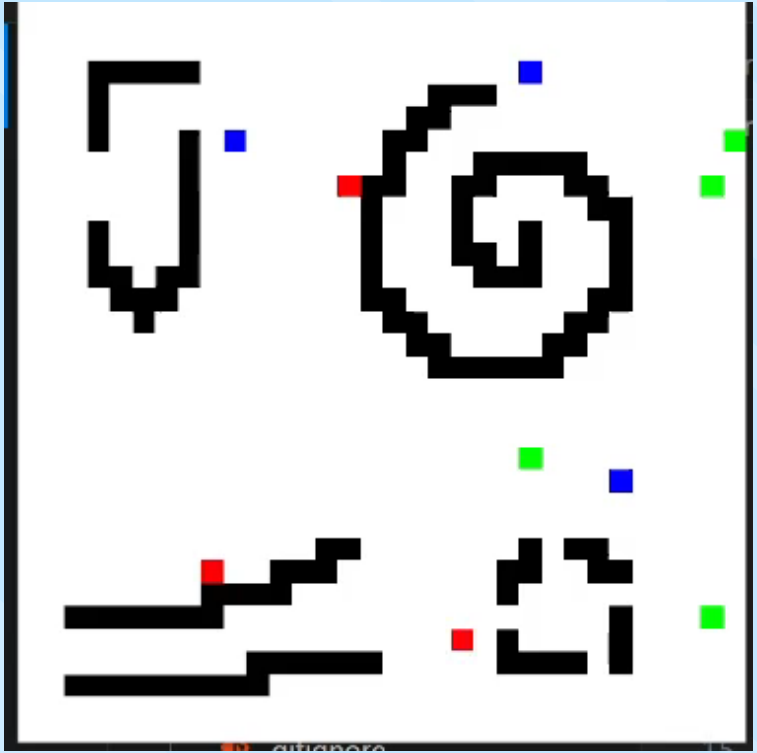
Efektem tych obserwacji jest symulacja wzroku agenta. Im więcej pól pod rząd zajmuje jakiś obiekt, tym bliżej znajduje się on agenta.

Jest to swego rodzaju widok 2D otoczenia, z którego agent może wywnioskować pozycje innych agentów i obiektów lub ścian akwarium.

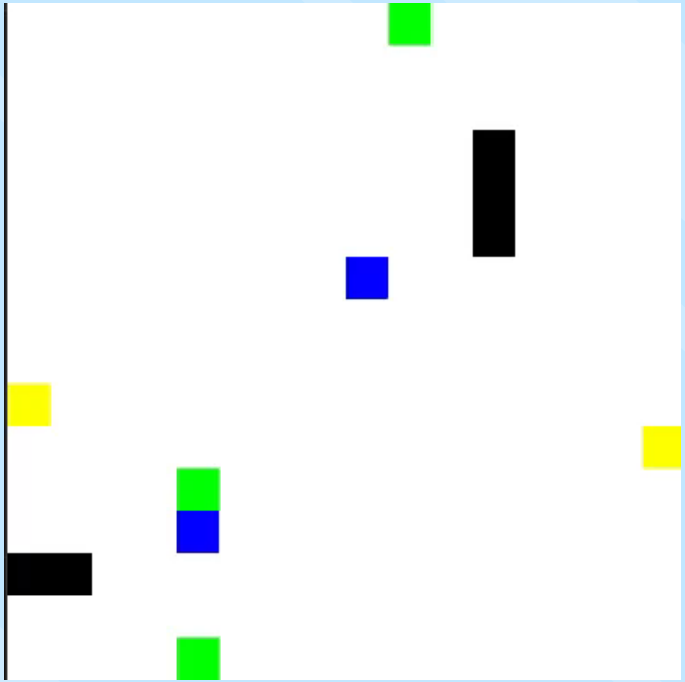
0°	3°	6°	9°	•••	348°	351°	354°	357°



Dawna mapa



Nowa mapa



Deep Q Learning

Algorytm Q-uczenia tworzy tabelkę Q, która przypisuje każdemu stanowi najlepszą akcję do podjęcia. Podejście to działa bardzo dobrze dla małych środowisk, lecz w przypadku większych ilości możliwych stanów, bardzo szybko przestaje mieć sens.

Rozwiązaniem jest Q-uczenie głębokie, które zamiast tworzyć tabelkę, przybliża funkcję, która działa w podobny sposób, czyli w zależności od stanu produkuje najlepszą możliwą akcję.

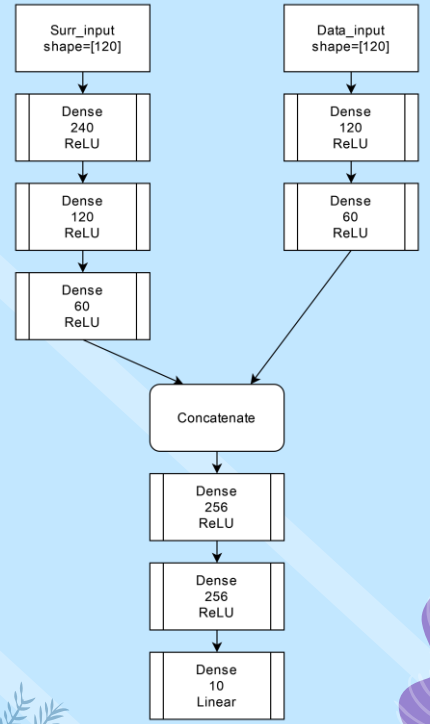
Algorytm posiada model, który na początku obserwuje i zapisuje działania losowe, a potem sam jest na nich uczony i ulepsza swoje działanie.

Model sieci

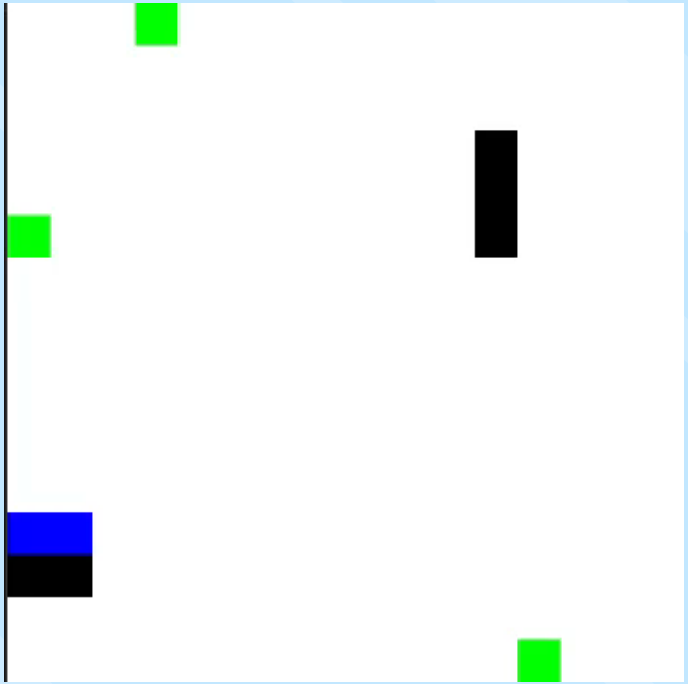
Sieć jest posiada dwa wejścia:

- n-elementowy wektor obserwacji kątowych
- 3-elementowy wektor parametrów agenta, padding do n elementów przy użyciu zer

Dane są przetwarzane i na wyjściu zwracane jest prawdopodobieństwo wykonania każdej z 10 akcji

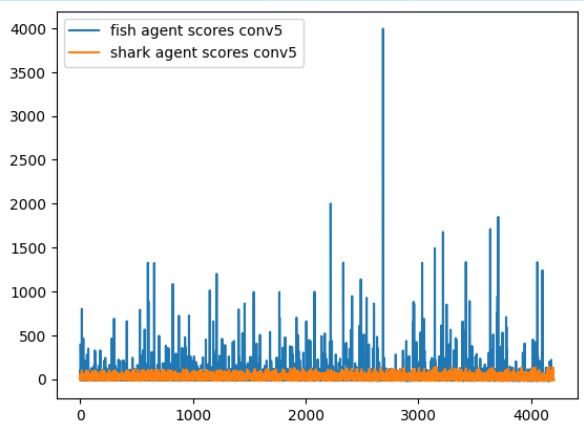


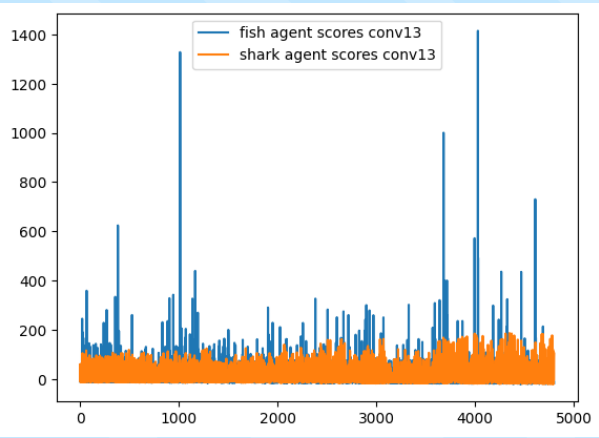
Wytrenowane modele



Wykres nagród

Starsze wykresy nagród





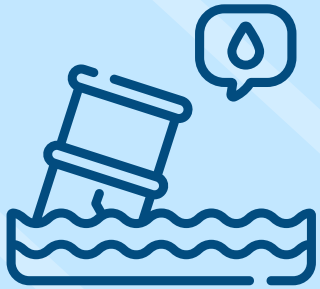
Problem z uczeniem

DQN ze swojej natury ma preferencje, to znaczy często wybiera jeden konkretny typ ruchu i go powtarza aż do skutku.
Może być to spowodowane przez:

- Niewłaściwe nagrody, zwłaszcza dla rekinów
- Zbyt dużą przestrzeń obserwacji
- Zbyt luźne zezwolenia na podejmowanie akcji

Plany na poprawę:

- Zmiana z DQN na A2C lub PPO
- Uproszczenie przestrzeni obserwacji
- Zaimplementowanie maski akcji



Wnioski z projektu

- Największym problemem w stworzeniu środowiska była słaba dokumentacja PettingZoo, środowiska sekwencyjnego AECEnv oraz braku przykładów z których możnaby się wspomóc
- Mimo tych trudności środowisko działa prawidłowo i zadowalająco
- Największym aktualnym problemem są nagrody i ich niezbyt przewidywalne zachowanie
- Stworzenie sieci zajęło długo czasu ze względu na nietypową przestrzeń obserwacji
- Żadna z próbowanych bibliotek RL nie oferowała dobrego gotowego rozwiązania dla agentów
- W celu lepszych efektów konieczne jest wprowadzenie zmian w środowisku