PROJEKT

Porównanie uczenia ze wzmocnieniem z programowaniem dynamicznym (value iteration)

Przygotowali:
Maciej Brąszkiewicz 147531
Witold Szerszeń 147512

Poznań 2024

Spis treści

1 Przedstawienie tematu

Zadaniem projektu jest porównanie dwóch różnych metod stabilizacji wahadła w pozycji pionowej z uwzględnieniem działania grawitacji.

Pierwszą z nich jest metoda związana z uczeniem ze wzmocnieniem (reinforcement learing). W tym celu zastosowano algorytm PPO (Proximal Policy Optimization). W algorytmie PPO funkcja kosztu jest specjalnie zmodyfikowana, aby ograniczyć wielkość zmian w polityce agenta. Dzięki temu nowa polityka pozostaje blisko starej, co zapewnia stopniowe i stabilne ulepszanie strategii działania. Kluczowym elementem PPO jest koncepcja "trust region", który ogranicza zbyt duże zmiany w polityce, gwarantując stabilność procesu uczenia i poprawiając efektywność adaptacji agenta do środowiska. Skorzystano z implementacji dostępnej pod linkiem [1]. Konieczna była edycja kodu polegająca na przeniesieniu funckjonalności z biblioteki gym na nowszą wersję - gymnasium.

Drugim sposobem, który został użyty jest podejście korzystające z metody Value Iteration. Polega ona na znajdowaniu optymalnej funkcji kosztu całkowitego poprzez iteracyjne rozwiązywanie równań Bellmana. Wykorzystuje koncepcję programowania dynamicznego do otrzymania funkcji wartości, która przybliża optymalną funkcję wartości, iteracyjnie ją poprawiając, aż do jej zbieżności (lub bliskiego jej wartości). Algorytm Value Iteration zaimplementowano samodzielnie.

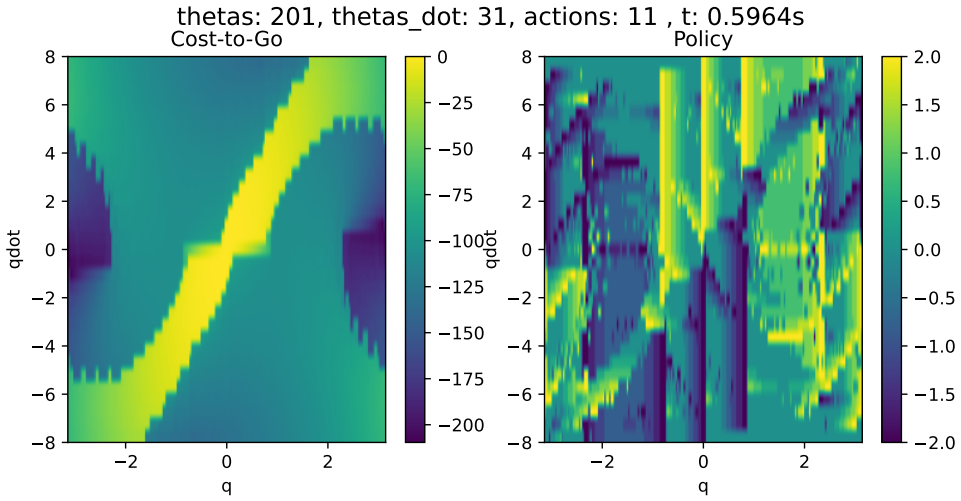
2 Działanie

2.1 Value iteration

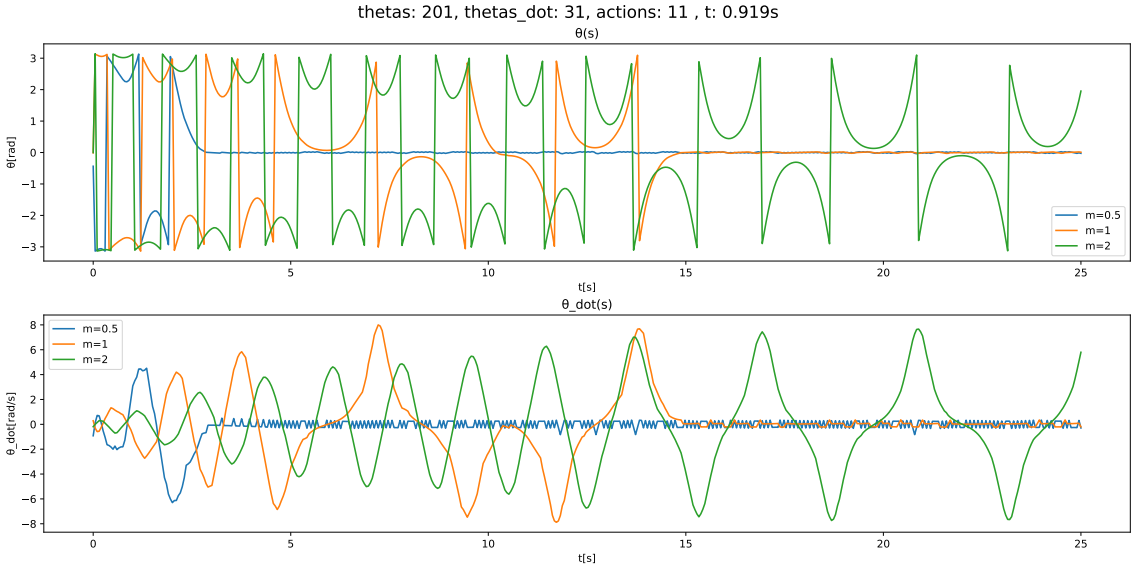
Działanie zaimplementowanych funkcjonalności zostało przetestowane dla różnych parametrów ilości stanów oraz akcji. Funkcję reward zaimplementowano w podany sposób:

$$reward = -(\theta^2 + 0.1 \cdot \dot{\theta}^2 + 0.001 \cdot u^2) \tag{1}$$

Położenie wahadła pionowo w górę odpowiada kątowi $\theta = 0$. Nauka odbywała się dla $m = 1kg$, $g = 9.81m/s^2$, $dt = 0.05s$, $l = 1m$. Parametry zostały dobrane zgodnie z tymi proponowanymi przez bibliotekę gymnasium.

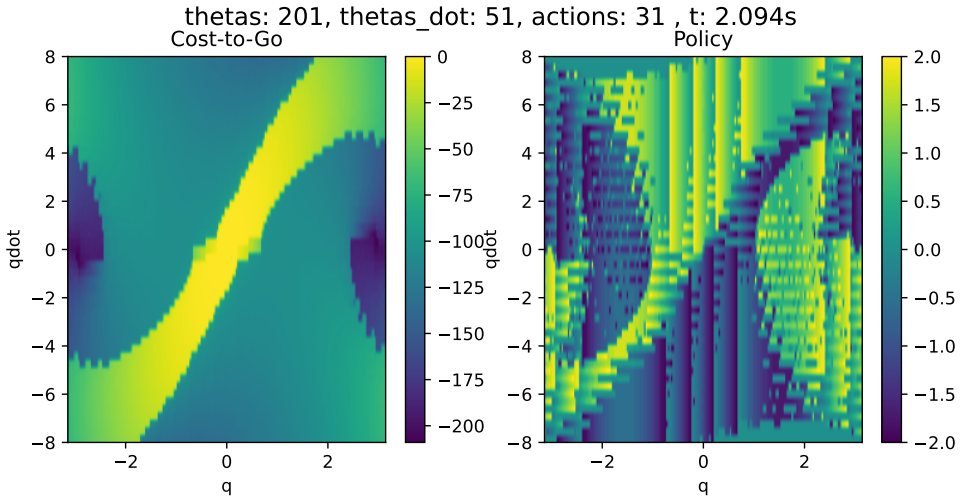


Rysunek 1: Funkcja kosztu i polityka dla danych parametrów.



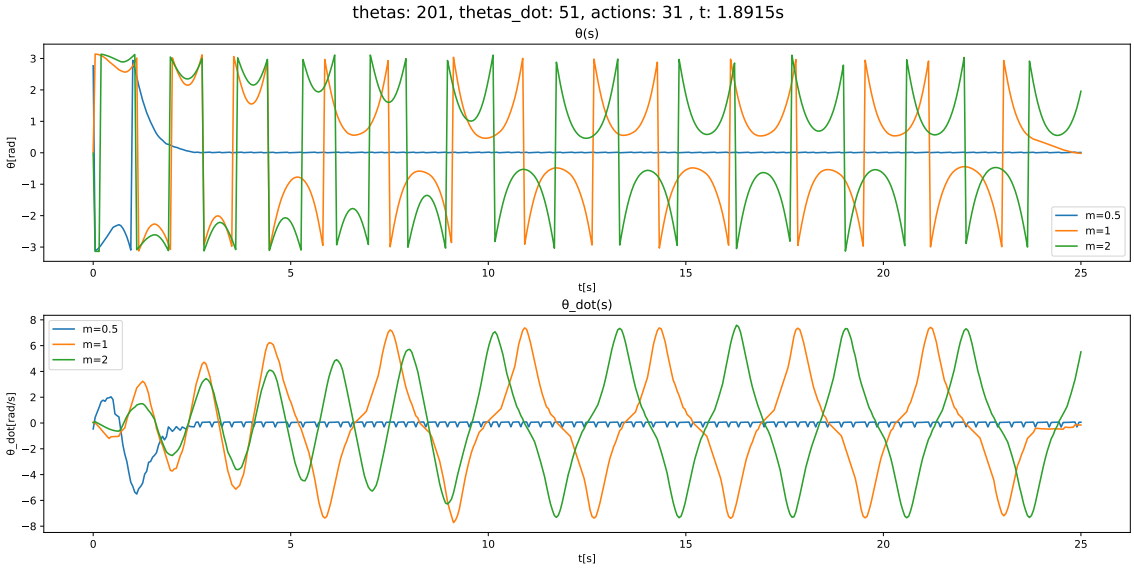
Rysunek 2: Przebiegi położenia i prędkości kątowej dla różnych mas wahadła.

Jak można zauważyć na wykresach 2, dla podanych parametrów, wahadło stabilizuje się dla mas równych 0.5kg oraz 1kg.



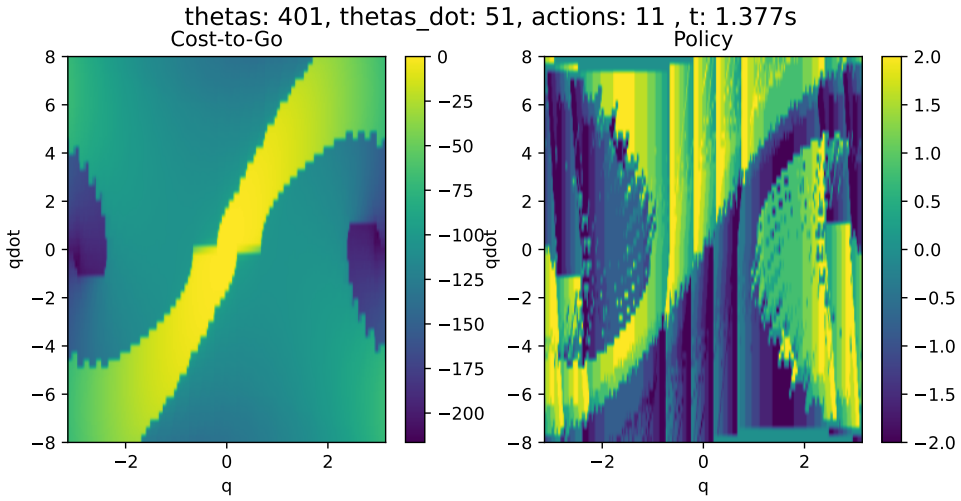
Rysunek 3: Funkcja kosztu i polityka dla danych parametrów.

Na wykresie 3 można zauważyć, że zwiększając liczbę dostępnych akcji, funkcja kosztu uległa "wygładzeniu".

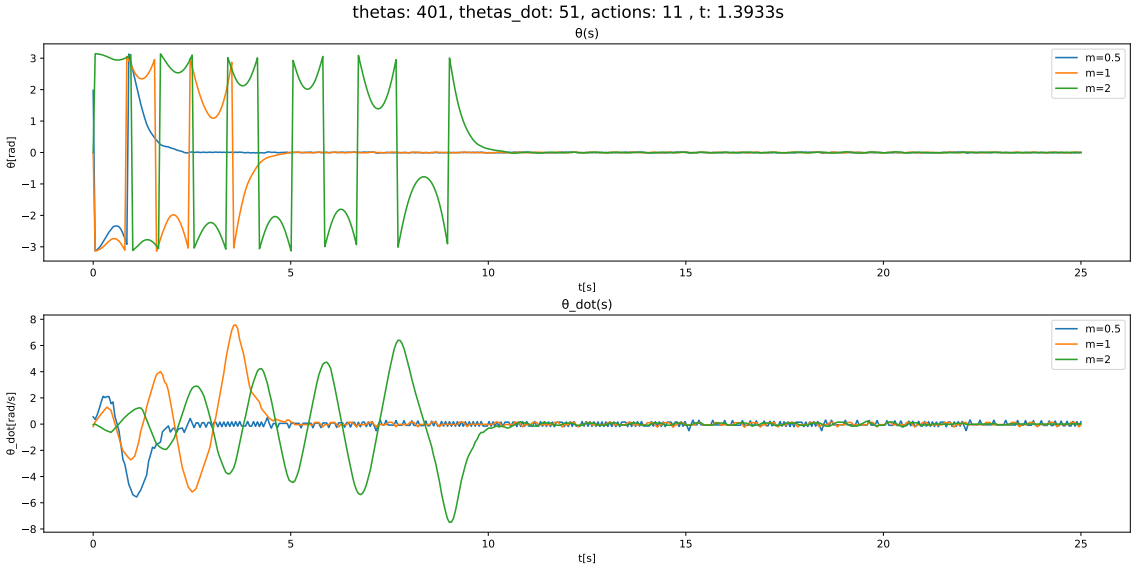


Rysunek 4: Przebiegi położenia i prędkości kątowej dla różnych mas wahadła.

Na wykresach 4 można zaobserwować, że zwiększanie ilości możliwych stanów, nie zawsze poprawia rezultaty. W podanym oknie czasowym wahadło nie ustabilizowało się dla $m = 1kg$, jednak w ostatnich sekundach można zaobserwować stabilizację prędkości, oraz przerwanie okresowości kąta.

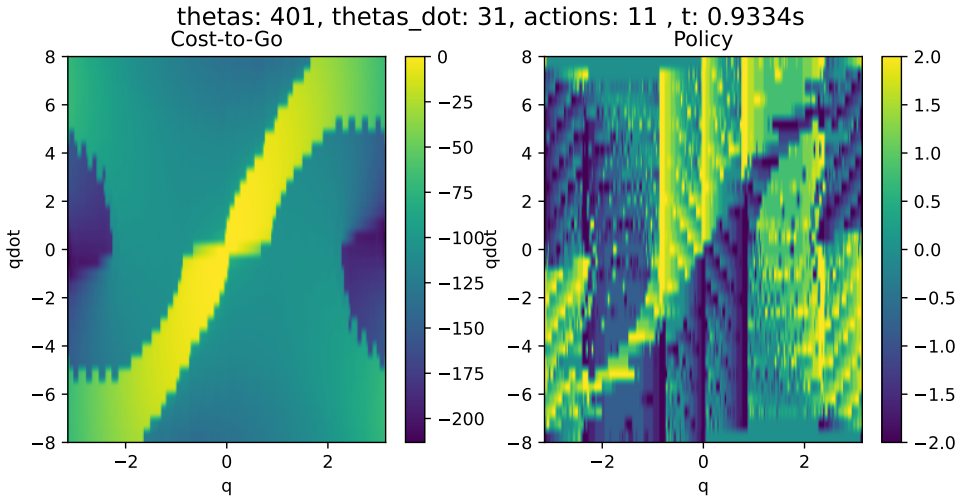


Rysunek 5: Funkcja kosztu i polityka dla danych parametrów.

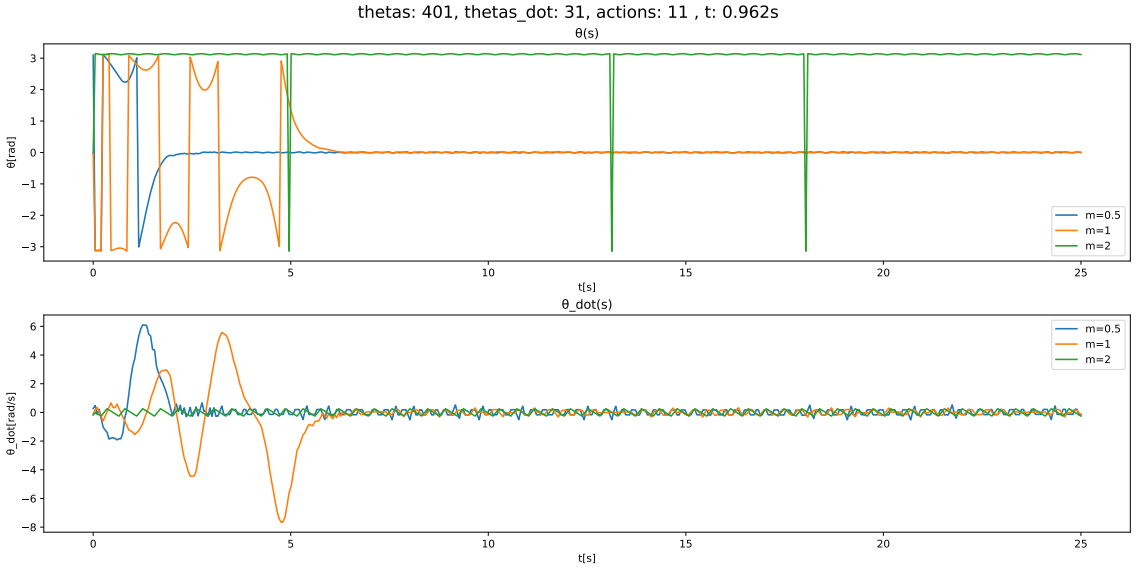


Rysunek 6: Przebiegi położenia i prędkości kątowej dla różnych mas wahadła.

Na wykresach 6 można zauważyć, że dla podanych parametrów wahadło stabilizuje się dla wszystkich z podanych mas. Znaczy to, że Value Iteration, dla odpowiednio dobrze zdyskretyzowanych przestrzeni, jest w stanie generalizować rozwiązanie.

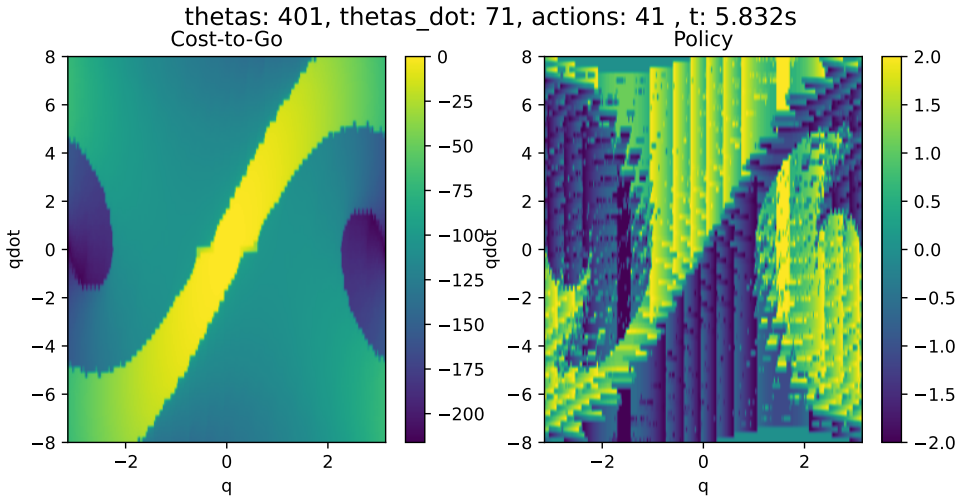


Rysunek 7: Funkcja kosztu i polityka dla danych parametrów.

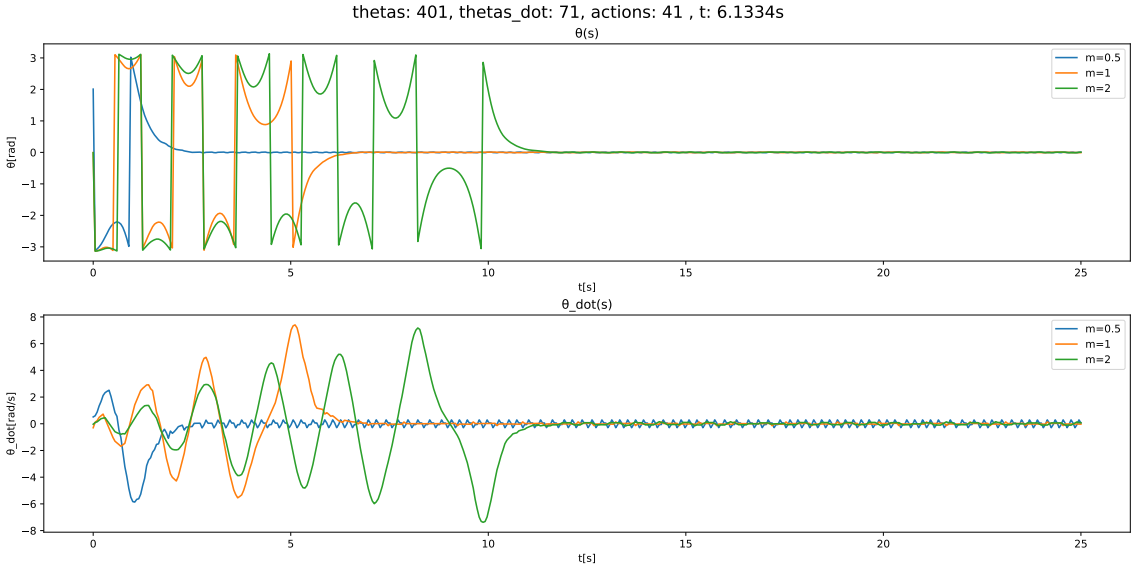


Rysunek 8: Przebiegi położenia i prędkości kątowej dla różnych mas wahadła.

Na wykresach 8 widać, że wahadło stabilizuje się dla każdej z podanych mas, jednak dla masy $m = 2kg$ wahadło stabilizuje się "na dole". Odchylenia w dół wynikają ze zmiany wartości kąta z π na $-\pi$.



Rysunek 9: Funkcja kosztu i polityka dla danych parametrów.



Rysunek 10: Przebiegi położenia i prędkości kątowej dla różnych mas wahadła.

Na wykresach 8 widać, że wahadło stabilizuje się dla każdej z podanych mas.

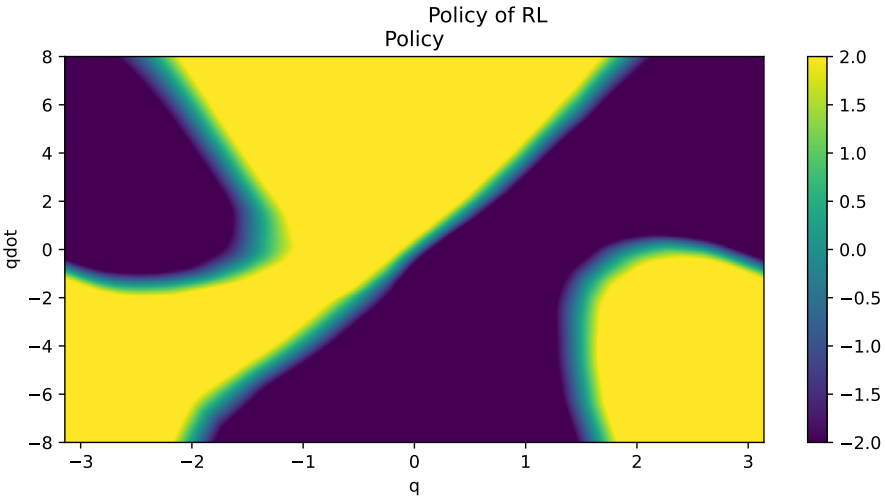
θ	$\dot{\theta}$	**number of actions**	**ITSE**
201	31	11	214.17
201	31	31	555.98
201	31	41	603.95
201	51	11	44.44
201	51	31	181.95
201	51	41	178.92
201	71	11	33.36
201	71	31	73.6
201	71	41	53.82
401	31	11	58.73
401	31	31	63.47
401	31	41	60.64
401	51	11	24.63
401	51	31	58.01
401	51	41	37.88
401	71	11	41.84
401	71	31	51.21
401	71	41	48.65

Tabela 1: Zależność ITSE od ilości możliwych stanów.

W tabeli 1 przedstawiono zależność pomiędzy ilością możliwych stanów, a wskaźnikiem jakości ITSE (Integral Time Squared Error).

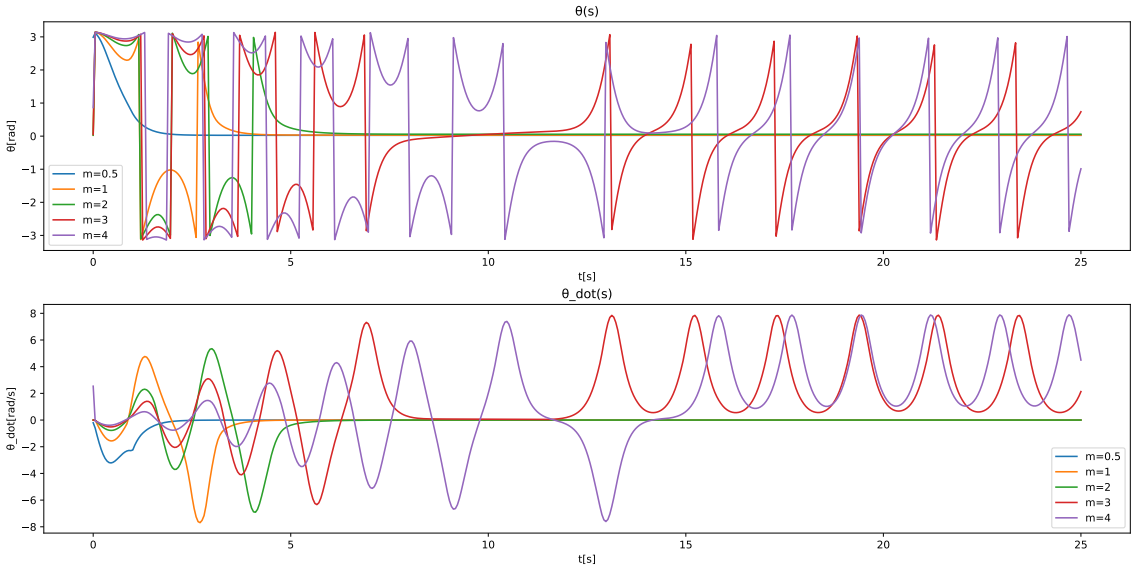
2.2 RL (PPO)

Do realizacji procesu nauki przy użyciu algorytmu PPO, wykorzystano 3-wartstwową sieć neuronową: warstwa wejściowa miała 3 węzły($cos(\theta)$, $sin(\theta)$, $\dot{\theta}$), ukryta 64 węzły, a wyjściowa 1 węzeł (wymuszenie). Nagroda wykorzystana do nauki sieci była taka sama jak w przypadku Value Iteration (równanie 1). Z tego powodu wykresy są w pewnym stopniu podobne. Jednak dzięki zastosowaniu sieci neuronowej nie ma konieczności dyskretyzacji stanu oraz możliwych akcji. W rezultacie wykres jest "gładki", co przedstawia rysunek 11.



Rysunek 11: Policy dla sieci neuronowej nauczonej przez RL.

Jeżeli chodzi o uzyskane przebiegi, to dla $m = 1kg$ (dla której był uczony) wahadło stabilizuje się. Jednak NN nie jest w stanie generalizować rozwiązania dla dużej grupy mas. Jak można zaobserwować, stabilizacje otrzymano dla co najwyżej $m = 2kg$. Mniejsze masy nie stanowią problemu, gdyż nie wymagają etapu rozbujania wahadła, tym samym szybciej zbiegają do pożądanych wartości.



Rysunek 12: Przebiegi położenia i prędkości kątowej dla różnych mas wahadła.

3 Podsumowanie

Dla testowanego zadania, RL w postaci PPO niekoniecznie działa lepiej niż DP. Czas trwania nauki NN był znacznie dłuższy niż optymalizacji funkcji kosztu dla Value Iteration. Dużą zaletą użycia NN jest brak konieczności dyskretyzacji stanu oraz możliwych akcji, co skutkuje naturalniejszymi rezultatami, jednak uzyskane wyniki mogą nie być prawdziwe w całej przestrzeni. Dla obu metod udało się uzyskać obiecujące rezultaty.

5. Jak chcecie w prosty sposób pokazać, że dyskretyzacja ma znaczenie to warto zaprezentować to w inny sposób, np. jako wykres gdzie na jednej osi będą jakoś określone parametry dyskretyzacja a na drugiej jakość stabilizacji najlepiej skondensowana do jednej liczby.

Literatura

[1] `https://github.com/ericyangyu/PPO-for-Beginners.git`