Sprawozdanie WSI Lab6 Tobiasz Kownacki

Zadanie:

Proszę zaimplementować algorytm Q-Learning i użyć go do wyznaczenia polityki decyzyjnej dla problemu FrozenLake8x8. W problemie tym celem agenta jest przedostanie się przez zamarznięte jezioro z domu do celu, unikając dziur (zawsze rozpoczynamy epizod z górnego lewego rogu mapy, który ma współrzędne 0). Na początku zbadać wersje bez poślizgu. Dla każdego epizodu, dla minimum 25 niezależnych uruchomień uczenia Q-learning, wyliczyć i zwizualizować na wykresie średnie wartości oryginalnych nagród (tych z gym) w funkcji numeru epizodu. Należy porównać wyniki domyślnego sposobu nagradzania z dwoma własnymi systemami nagród i kar. Propozycje te powinny częściej niż oryginalny system dawać niezerową informację zwrotną agentowi. W drugiej części badań należy włączyć poślizg, należy zwiększyć liczbę epizodów do 10000. Jakie są zmiany w stosunku do wersji bez poślizgu? Jak często udaje się dojść do celu? Jak zaproponowane wcześniej systemy nagród wpływają na wyniki w odniesieniu do wyników podstawowego systemu oceny? Spisać wnioski.

Parametry:

Dobrane parametry do algorytmu Q Learning:

- learning rate (β): 0,25
- discount rate (γ): 0,9
- max liczba kroków w pojedynczej epoce: 200
- epsilon (ε) : zmniejsza się wykładniczo od 1,0 do 0,001

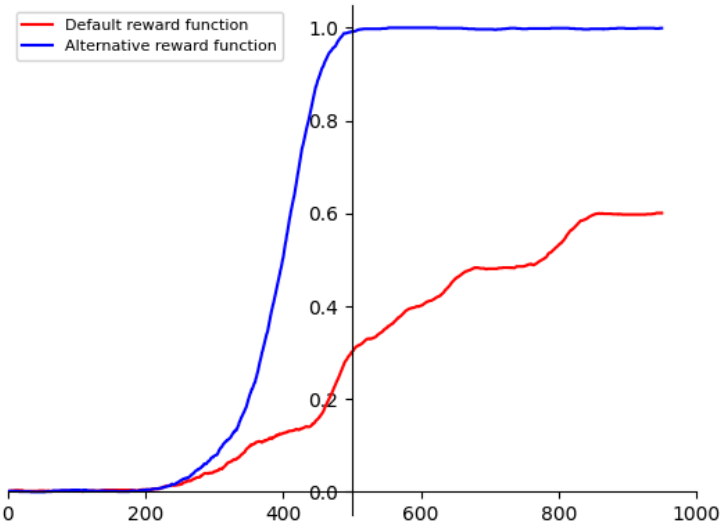
Funkcje oceny:

- domyślna: 1 za dojście do celu, 0 w przeciwnym przypadku
- Alternatywna_1: -1 za wejście w dziurę, 10 za dojście do celu, 0 w przeciwnym przypadku
- Alternatywna_2: -5 za wejście w dziurę, 20 za dojście do celu, -1 za próbę wyjścia poza obszar mapy, 0 w przeciwnym przypadku

1. Wersja bez poślizgu:

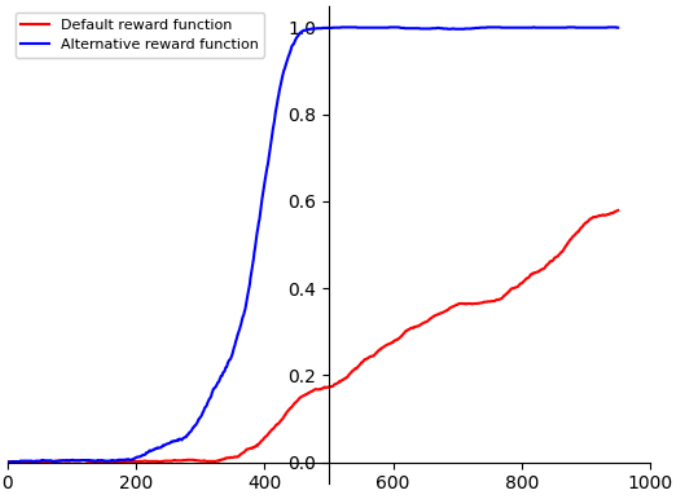
Wykres przedstawia szansę na dojście do celu w zależności od liczby epok na których trenowany był agent. Wyniki są średnią pochodzącą z 25 niezależnych uruchomień algorytmu Q Learning.

a) Porównanie domyślnej funkcji oceny z alternatywną nr1.



Domyślna funkcja oceny ma problem z dojściem do celu nawet po 1000 epok. Po ukończeniu tysięcznej epoki, szansa na dojście do celu wynosi średnio 60%. Alternatywna funkcja radzi sobie znakomicie. Już po ukończeniu pięćsetnej epoki praktycznie zawsze nasz agent dochodzi do celu. Dzięki dodaniu kary za wejście w dziurę, algorytm się uczy aby tam nie wchodzić, co skutkuje szybszym nauczeniem się jak dojść do celu.

b) Porównanie domyślnej funkcji oceny z alternatywna nr2.

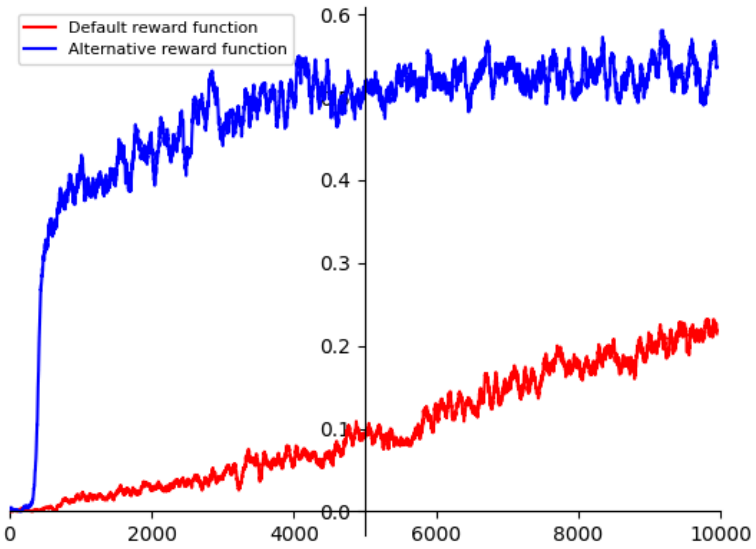


Alternatywna funkcja oceny nr 2 radzi sobie jeszcze lepiej niż nr 1. Dodanie kary za próbę wyjścia z mapy, powoduje unikanie dziur i ścian, co jeszcze bardziej przyśpiesza naukę naszego agenta

2. Wersja z poślizgiem

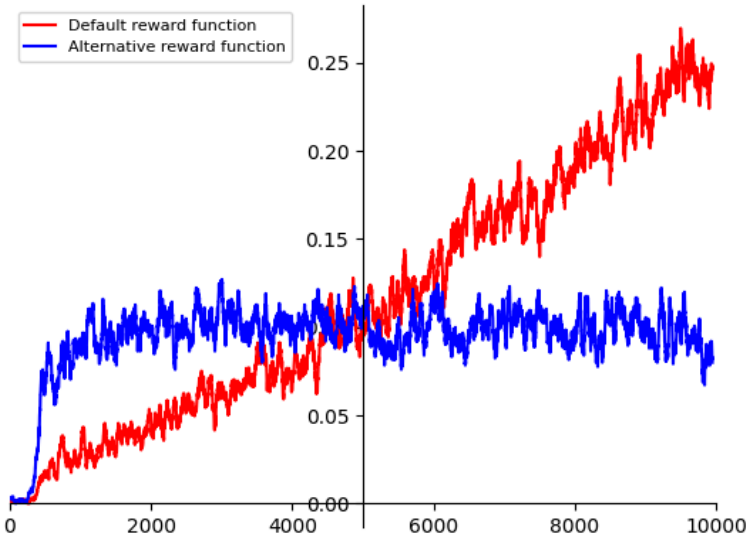
Wykres przedstawia szansę na dojście do celu w zależności od liczby epok na których trenowany był agent. Wyniki są średnią pochodzącą z 25 niezależnych uruchomień algorytmu Q Learning.

a) Porównanie domyślnej funkcji oceny z alternatywną nr 1.



Włączenie poślizgu diametralnie zmniejsza szanse dojścia agenta do celu, dla obu funkcji nagrody, pomimo zwiększenia zakresu osi X wykresu do 10 000. Domyślna funkcja oceny przez cały wykres notuje powolny i stabilny wzrost. Po nauce agenta na 10 000 epokach jego średnia skuteczność w dojściu do celu wynosi zaledwie 20%. Alternatywna funkcja oceny nr1 radzi sobie o wiele lepiej. Na początku rośnie wykładniczo i w okolicy 500 epoki, skuteczność dobija do około 0,3. Potem notuje powolny wzrost. Po nauce agenta na 10 000 epokach jego średnia skuteczność w dojściu do celu wynosi około 50% Spadek skuteczności zapewne spowodowany jest tym, że w wersji z poślizgiem istnieje tylko ⅓ szans na pójście w wybranym kierunku. W miejscu obok celu jest wiele dziur, co skutkuje tym, że agent nie celowo do nich wpada.

b) porównanie domyślnej funkcji oceny z alternatywną nr 2.



W przypadku funkcji alternatywnej nr 2 wyniki są bardzo zaskakujące. Okazuje się, że przy włączonym poślizgu funkcja nagrody radzi sobie o wiele gorzej niż funkcja alternatywna nr 1, a nawet gorzej niż funkcja domyślna. Skuteczność alternatywnej funkcji nr 2 wynosi zaledwie około 10 % po przetrenowaniu agenta na 10 000 epokach. Krzywa funkcji alternatywnej nr 2 ma kształt alternatywnej nr 1. tylko odpowiednio pomniejszonej.

c) Porównanie funkcji alternatywnej nr 1 i 2.

Tak niespotykane na pierwszy rzut oka wyniki zachęciły mnie do przeprowadzenia dodatkowych badań i sprawdzeniu jaki status ma agent po zakończeniu epoki. Wyniki są średnią z 25 niezależnych uruchomień.

funkcja/stan agenta	Dojście do celu	Wejście w dziurę	pozostały stan
Alternatywna nr 1.	47,81%	20,72%	31,79%
Alternatywna nr 2	9,59%	36,14%	54,25%

Dodanie kary za wejście w ścianę w funkcji alternatywnej nr 2 w wersji z poślizgiem powoduje, że nasz agent o wiele częściej wpada w dziurę i błąka się po mapie.

3. Wnioski:

Wszystkie alternatywne funkcje nagrody wypadają lepiej w przypadku wersji bez poślizgu. Dawanie częściej niezerowej oceny agentowi poskutkowało lepszymi wynikami. Nowe funkcje oceny pozwalają uzyskać skuteczność prawie 100% po nauce agenta na zaledwie 500 epokach. Funkcja domyślna radzi sobie gorzej, najpewniej potrzebuje większej liczby epok do trenowania. Dla problemu Frozen Lake w wersji bez poślizgu najlepiej wybrać funkcje alternatywną nr 2.

W przypadku wersji z poślizgiem, algorytm alternatywny nr 1 radzi sobie całkiem dobrze osiągając skuteczność około 50% po nauce na zaledwie 4 000 epok. Funkcja nagrody domyślna osiąga skuteczność około 20%. Zapewne potrzebuje większej liczby epok do trenowania, aby dojść do skuteczności funkcji alternatywnej nr 1. Funkcja alternatywna nr 2, ma bardzo niską skuteczność sięgającą około 10% po nauce agenta na 10 000 epokach. Spowodowane jest to dodaniem kary za próbę wyjścia za mapę. W wersji z poślizgiem ta funkcja nagrody radzi sobie najgorzej. Dla problemu Frozen Lake w wersji z poślizgiem najlepiej wybrać funkcje alternatywną nr 1.