Podstawy uczenia maszynowego
Uczenie przez wzmacnianie (Tlareg i strzyga) – raport

Mateusz Kocot

15 maja 2021

Spis treści

1 Sytuacja testowa

Stworzono mapę o rozmiarze 8×8. Zamek znajduje się w prawym górnym rogu. Poza terenem dookoła zamku, mury znajdują się także na kilku innych polach mapy. Mapę oraz przykładowe akcje przedstawiono na rys. 1.

Wiedźmin porusza się i atakuje zgodnie z opisem zadania. W przypadku wejścia na strzygę, umiera. Strzyga została wyposażona w dwa schematy poruszania:

1. Schemat deterministyczny – strzyga porusza się po określonym prostokącie atakując tylko w określonych miejscach. W przypadku natrafienia na Tlarega, strzyga powstrzymuje się z ruchem. Po powrocie z zamku, strzyga trafia na pole losowo wybrane z puli miejsc, na których może się znaleźć.
2. Schemat stochastyczny - strzyga atakuje z prawdopodobieństwem równym 0.3. W każdym ruchu, z prawdopodobieństwem równym 0.5, porusza się w tym samym kierunku co poprzednio. W pozostałych przypadkach, losowany jest nowy kierunek. Jeżeli nowy kierunek wprowadza strzygę na ścianę, strzyga odbija się od niej. Jeżeli i taki ruch nie jest możliwy, strzyga pozostaje w miejscu. Podobnie jak poprzednio, strzyga nie wchodzi na Tlarega.

Rys. 1: Stworzona mapa. Na szaro zaznaczono ściany, na zielono – Tlarega, a na czerwono – strzygę. Rys. po lewej: stan początkowy, rys. na środku: atak strzygi (kolor jasnoczerwony), rys. po prawej: atak Tlarega (kolor jasnozielony), strzyga została uderzona i uciekła do zamku

Atak strzygi powiązany jest z kierunkiem, w którym patrzy (ten, z którym weszła na dane pole). Atakuje ona trzy pola znajdujące się przed sobą. Na przykład, na środkowej mapie na rys. 1, strzyga w poprzedniej turze wykonała ruch w lewo, dlatego teraz atakuje pola po lewej stronie.

Dodatkowe założenia:

- Klątwa rzucana jest po 1000 turach bez trafienia strzygi przez wiedźmina.
- Strzyga ma 5 żyć.
- Strzyga spędza w zamku 5 tur.

2 System kar i nagród

System kar i nagród został stworzony tak, by w miarę możliwości zmuszać wiedźmina do jak najszybszego osiągnięcia celu ostatecznego, czyli zabicia strzygi. W związku z tym, nagrody i kary związane ze śmiercią strzygi lub Tlarega są o wiele rzędów wielkości większe od pozostałych. Poza tym, wiedźmin premiowany jest, gdy zbliży się do strzygi i karany w przeciwnym przypadku. System uwzględnia także nagrodę za uniknięcie ataku oponentki oraz karę za niepotrzebne machanie mieczem i zużywanie eliksirów.

Próbowano wiele kombinacji wartości. Ostatecznie zdecydowano się na następujące:

- śmierć Tlarega po 1000 turach obijania się: $-1.000.000$,
- śmierć Tlarega w walce: -100.000,
- nietrafiony atak Tlarega: -50,
- nietrafiony atak strzygi: 2,
- trafienie strzygi: 10.000,
- ostateczne pokonanie strzygi: 100.000,
- zbliżenie się: 0.2,
- niezbliżenie się: -0.1.

3 Proces uczenia – Sarsa

Wybrano dwa zestawy argumentów:

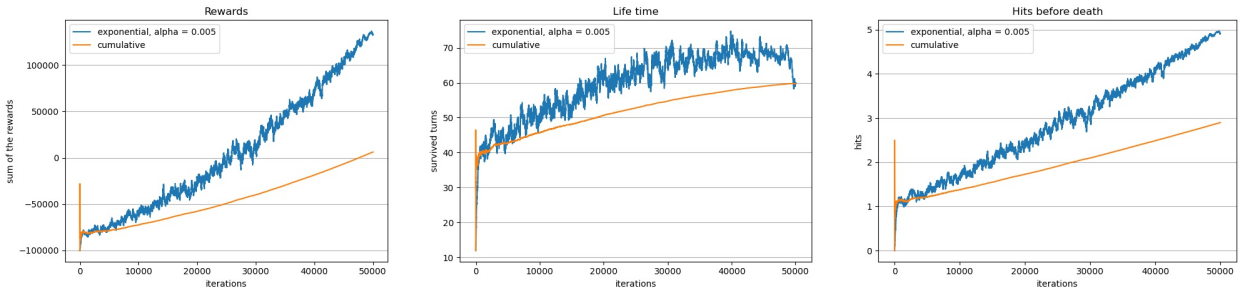
1. $learing\_rate = 0.9, discount\_factor = 0.9$
2. $learing\_rate = 0.1, discount\_factor = 0.1$

$experiment\_rate$ w każdym przypadku maleje liniowo wraz z liczbą iteracji od 0.99 do 0.01.

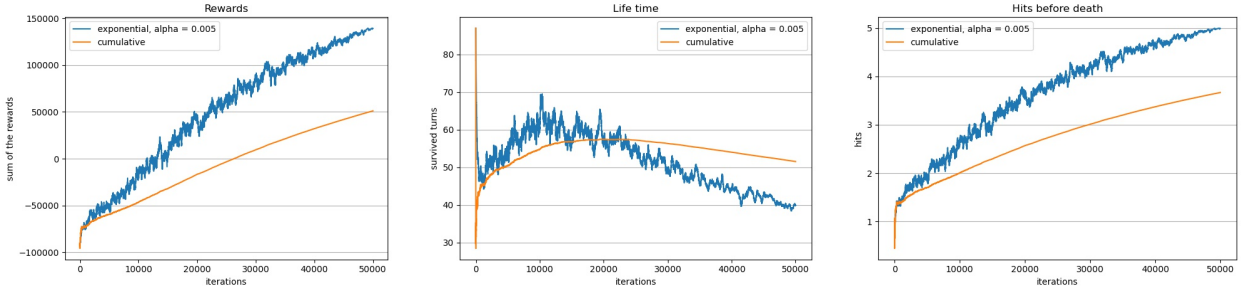
Na wykresach zostaną wykorzystane dwa warianty średniej kroczącej: kumulacyjna (lepiej obrazująca ogólny trend) oraz eksponencjalna z parametrem określającym wagę dodawanych do sumy wartości: $alfa = 0.005$ (lepiej pokazująca aktualny i ostateczny wynik).

3.1 Deterministyczny schemat poruszania strzygi

Na początku eksperyment został wykonany dla deterministycznego schematu poruszania. Wykresy obrazujące przebieg uczenia przedstawiono na rys. 2 (pierwszy zestaw parametrów) i rys. 3 (drugi zestaw parametrów). Czasy wykonania wyniosły ok. 80 s.



Rys. 2: Wykresy obrazujące przebieg uczenia. Pierwszy zestaw parametrów, schemat poruszania strzygi: deterministyczny, algorytm: SARSA

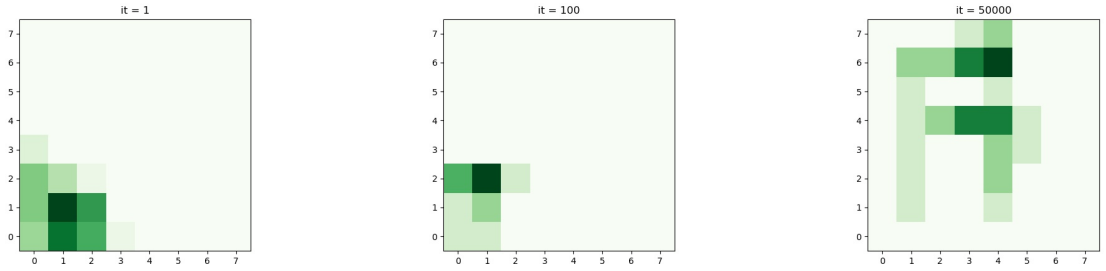


Rys. 3: Wykresy obrazujące przebieg uczenia. Drugi zestaw parametrów, schemat poruszania strzygi: deterministyczny, algorytm: SARSA

W obu przypadkach, po 50.000 iteracji udało się w końcu dojść do sytuacji, w której prawie zawsze Tlareg atakuje 5 razy w trakcie jednej gry, a to oznacza, że tę grę wygrywa. Lepiej SARSA poradziła sobie z drugim zestawem argumentów. W przeciwieństwie do pierwszego

zestawu, udało się skrócić czas życia wiedźmina, co w połączeniu ze zwiększeniem liczby trafień daje szybsze ukończenie gry. Poza tym, dla mniejszej liczby iteracji, wyniki są relatywnie lepsze.

W obu przypadkach, wiedźmin w początkowych etapach nauki szybko ginął. Później uczył się schematu ruchów. Widać to na histogramach częstotliwości przebywania na różnych polach mapy. Na rys. 4 przedstawiono takie histogramy dla drugiego zestawu parametrów. Te dla pierwszego wyglądają podobnie.



Rys. 4: Histogramy częstotliwości przebywania Tlarega na różnych polach mapy po różnych liczbach iteracji. Drugi zestaw parametrów, schemat poruszania strzygi: deterministyczny, algorytm: SARSA

Zachowanie Tlarega zaobserwowano także z wykorzystaniem animacji. Po przejściu procesu nauczania, wiedźmin nauczył się ruchu strzygi i atakował tak, by samemu nie zostać zaatakowanym.

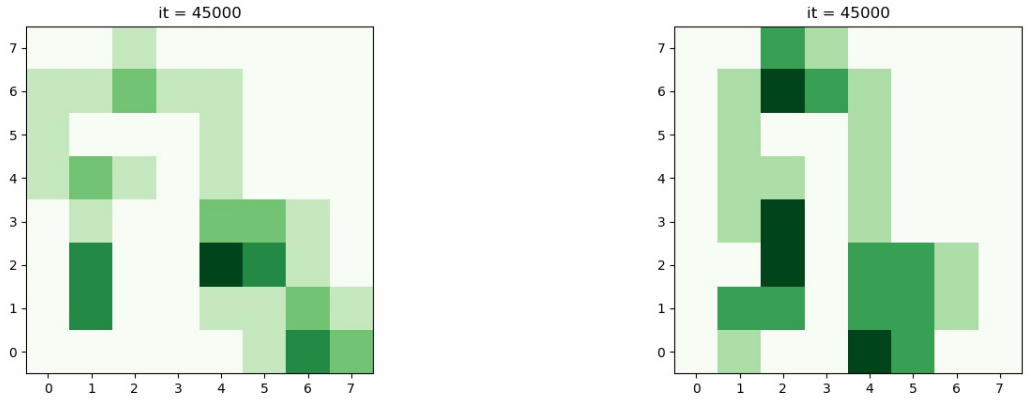
Generalnie, dla obu zestawów parametrów uzyskano podobne zachowanie. Jednakże, porównując ze sobą wiele histogramów częstotliwości przebywania na polach mapy, widać, że w przypadku pierwszego zestawu parametrów, więcej pól jest jasnozielonych (przykład: rys. 5 – zjawisko to nie jest wyraźne), co oznacza, że zgodnie z założeniami, wykorzystanie takiego zestawu zwiększa częstotliwość eksperymentowania i wiedźmin porusza się trochę dalej od stałego schematu ruchów strzygi.

3.2 Stochastyczny schemat poruszania strzygi

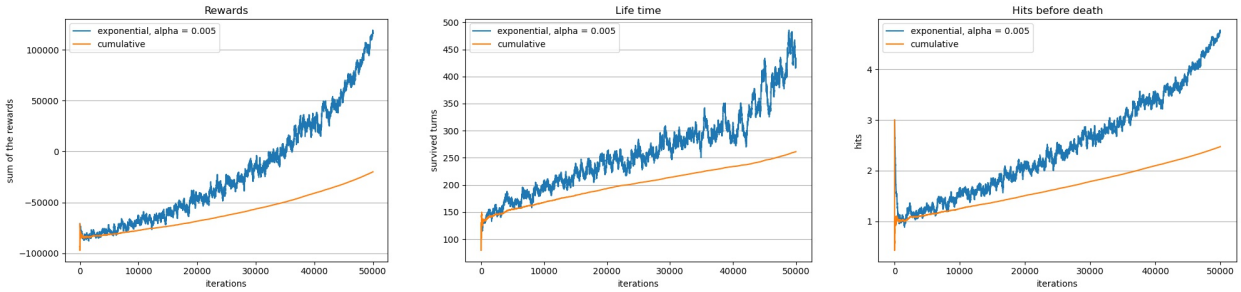
Wykresy dla stochastycznego schematu przedstawiono na rys. 6 (pierwszy zestaw parametrów) i na rys. 7 (drugi zestaw parametrów). Tam razem także wykonano 50.000 iteracji algorytmu, ale ze względu na mniej trywialny schemat poruszania, czasy działania wzrosły odpowiednio do ok. 6 i 7 minut.

Kształty wykresów są podobne w porównaniu do tych z poprzedniego punktu. Wzrosła natomiast długość życia wiedźmina, co jest zrozumiałe – trudniej teraz przewidzieć ruchy strzygi.

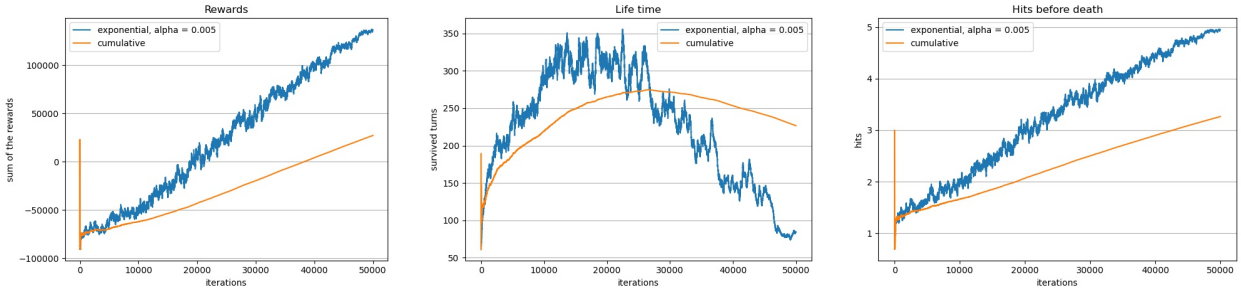
Histogramy nie są już skupione wokół stałego schematu poruszania strzygi (przykład: rys. 8). Jest to zrozumiałe, strzyga może już poruszać się po całej mapie, a każde pole jest dobre, żeby ją zaatakować. Jedynie na początku, gdy wiedźmin szybko ginie, porusza się on głównie w okolicy swojej pozycji startowej.



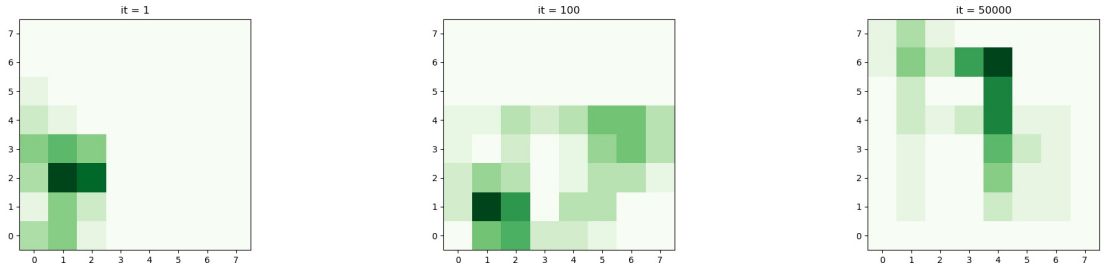
Rys. 5: Histogramy częstotliwości przebywania Tlarega na różnych polach mapy po różnych liczbach iteracji. Po lewej stronie: pierwszy zestaw parametrów, po prawej: drugi zestaw. Schemat poruszania strzygi: deterministyczny, algorytm: SARSA



Rys. 6: Wykresy obrazujące przebieg uczenia. Pierwszy zestaw parametrów, schemat poruszania strzygi: stochastyczny, algorytm: SARSA



Rys. 7: Wykresy obrazujące przebieg uczenia. Drugi zestaw parametrów, schemat poruszania strzygi: stochastyczny, algorytm: SARSA

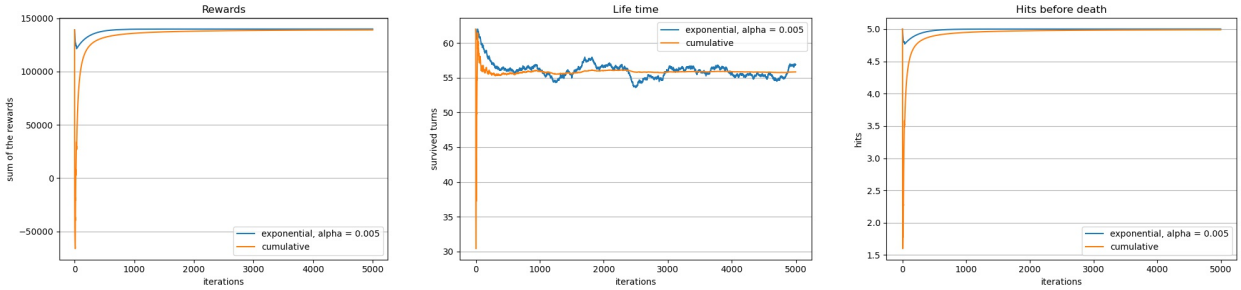


Rys. 8: Histogramy częstotliwości przebywania Tlarega na różnych polach mapy po różnych liczbach iteracji. Drugi zestaw parametrów, schemat poruszania strzygi: stochastyczny, algorytm: SARSA

4 Proces uczenia – Q-Learning

Przed wykonaniem eksperymentu myślałem, że zastosowanie Q-Learningu poprawi otrzymywane wyniki. Wydaje się, że zastosowanie tego algorytmu w tym przypadku jest bardziej wskazane, nie ma tu w końcu żadnych ryzykownych miejsc jak przepaść, obok której lepiej nie chodzić. Zwłaszcza w przypadku deterministycznym zmiana ta wydaje się bardzo dobra. Wiedząc dokładnie co się stanie, nie trzeba eksperymentować.

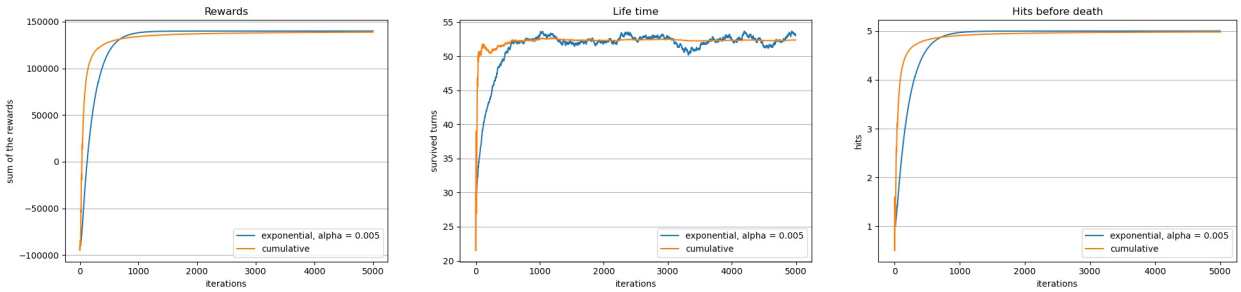
Nie spodziewałem się natomiast aż tak dużej poprawy. Po zmianie algorytmu na Q-Learning, już dla kilku tysięcy iteracji znajdowane jest rozwiązanie, które zapewnia zabicie strzygi praktycznie za każdym razem. Dzięki temu, czas działania znacząco zmalał. Co więcej, wyniki są podobne w przypadku schematu stochastycznego. Przykłady dla schematu deterministycznego przedstawiono na rys. 9 (pierwszy zestaw parametrów) i rys. 10 (drugi zestaw parametrów)



Rys. 9: Wykresy obrazujące przebieg uczenia. Pierwszy zestaw parametrów, schemat poruszania strzygi: deterministyczny, algorytm: Q-Learning

W ciągu 2000 iteracji udało się uzyskać podobne wyniki jak w ciągu 50.000 iteracji z wykorzystaniem algorytmu SARSA.

Na tych przykładach można także zaobserwować znaczenie parametrów uczenia ze wzmocnieniem. Z pierwszym zestawem, wykresy wzrastają szybko i gwałtownie. Drugi zestaw premiuje stabilność i jest to też widoczne na wykresach – algorytm potrzebuje wówczas więcej iteracji by osiągnąć określone wyniki. Efekt ten nie jest aż tak widoczny w przypadku schematu sto-



Rys. 10: Wykresy obrazujące przebieg uczenia. Drugi zestaw parametrów, schemat poruszania strzygi: deterministyczny, algorytm: Q-Learning

chastycznego, gdzie odpowiednie wykresy stworzone z wykorzystaniem pierwszego i drugiego zestawu parametrów są niemal identyczne.

5 Wnioski

Całkiem efektownie udało się wytrenować Tlarega, który teraz bez problemu potrafi pokonać strzygę. O ile w przypadku algorytmu SARSA, trzeba było poczekać kilka, a najlepiej kilkanaście minut, by odpowiednio wytrenować wiedźmina, to Q-Learning robił to samo w ciągu kilkunastu sekund. W tym przypadku, dobór parametrów nie okazał się być bardzo istotny, natomiast przy większym i bardziej rozbudowanym problemie, najprawdopodobniej byłaby to sprawa kluczowa.