



**Wydział Elektroniki
i Technik Informatycznych**

POLITECHNIKA WARSZAWSKA

Wstęp do Sztucznej Inteligencji

Ćwiczenie nr 3:
Dwuosobowe gry deterministyczne

Kaczmarek Robert
293377

Warszawa 2021

Polecenie

Zadanie polega na napisaniu programu lub skryptu, który buduje drzewo gry w wilka i owcę (ang. fox and hounds), a następnie gra sam ze sobą wykorzystując algorytm minimax.

Należy wykazać, że gra nie jest ustawiona.

Użyte narzędzia

Do wykonania zadania został użyty Python w wersji 3.8.2 oraz następujące biblioteki i moduły:

- pygame == 2.1.0
- numpy==1.21.4
- random
- argparse
- enum
- typing

Wyniki

Gra może zakończyć się, gdy gracz grający wilkiem dotrze do linii owiec lub nie będzie miał możliwości ruchu. Inny przypadek zakończenia rozgrywki to brak możliwych ruchów ze strony gracza grającego owcami. Gra nie może zakończyć się remisem.

Dodatnia wartość oceny pozycji oznacza przewagę wilka, a ujemna wartość oceny pozycji oznacza przewagę owiec. Przy oceny pozycji algorytm minimax uwzględnia takie rzeczy jak:

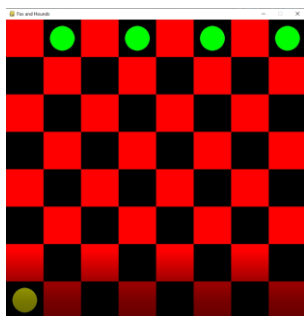
- Wygrana wilka,
- Wygrana owiec,
- Odległość wilka do początkowej linii owiec,
- Odległość wilka od owiec,
- Rozkład kolumnowy owiec na planszy,
- Owce w jednej linii,
- Przejście wilka za owcę,
- Stykanie się owiec z wilkiem.

Wymienione wyżej przypadki otrzymały swoje wagi wyznaczone w sposób eksperymentalny. Przy każdorazowym wywołaniu funkcji minimax zapisywane są do zmiennej wszystkie możliwe ruchy na planszy, które następnie są przetwarzane, aby wybierany ruch był losowy w przypadku zbioru tak samo ocenionych pozycji.

Przykład działania

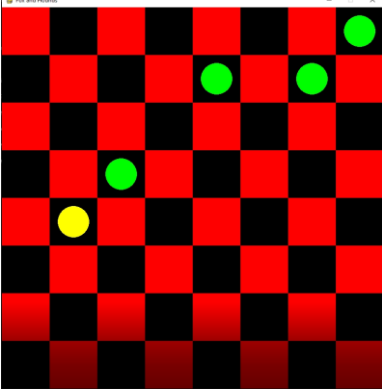
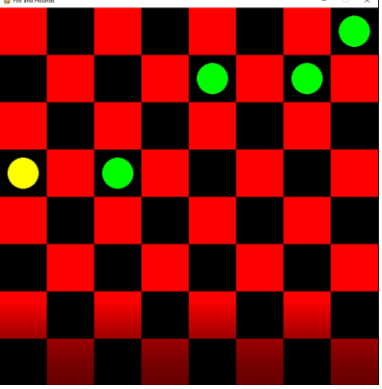
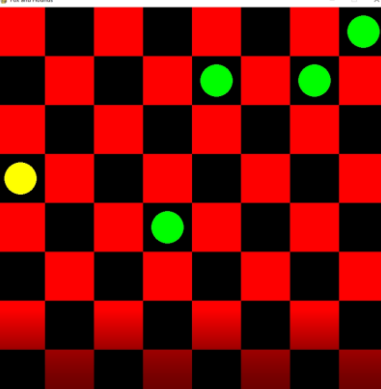
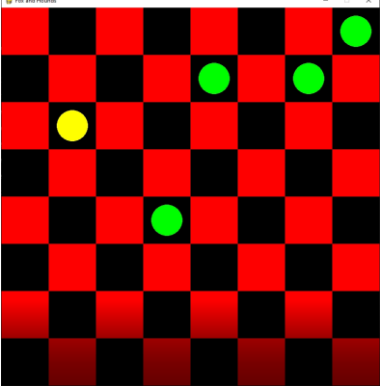
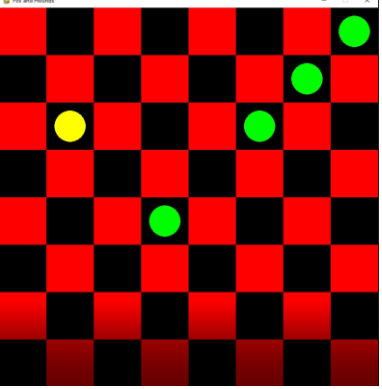
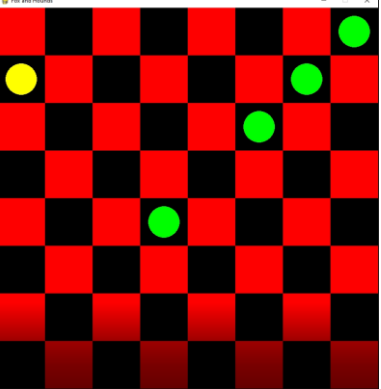
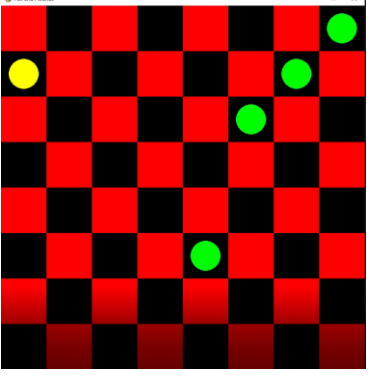
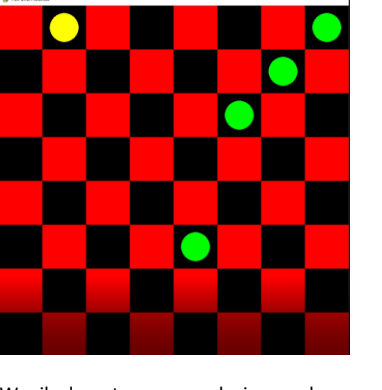
Wywołuję skrypt dla głębokości = 2, startowej kolumny wilka = 0 oraz jednej gry. Wilka będzie ruszał się minimaxem, a owce losowo.

```
python main.py -d 2 -fsc 0 -N 1
```



Rys. 1. Pozycja początkowa.

<p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (7, 0) to (6, 1) with value of 0.11104868913857678</p>	<p>Losowy ruch owiec.</p>	<p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (6, 1) to (5, 0) with value of 0.06342105263157895</p>
<p>Losowy ruch owiec.</p>	<p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (5, 0) to (4, 1) with value of 0.1643617021276596</p>	<p>Losowy ruch owiec.</p>
<p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (4, 1) to (3, 0) with value of 0.2244047619047619</p>	<p>Losowy ruch owiec.</p>	<p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (3, 0) to (4, 1) with value of 0.26479591836734695</p>

 <p>Losowy ruch owiec.</p>	 <p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (4, 1) to (3, 0) with value of 0.3744047619047619</p>	 <p>Losowy ruch owiec.</p>
 <p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (3, 0) to (2, 1) with value of 0.6871794871794872</p>	 <p>Losowy ruch owiec.</p>	 <p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (2, 1) to (1, 0) with value of 0.8419354838709678</p>
 <p>Losowy ruch owiec.</p>	 <p>Wynik algorytmu poprzedzający ruch: Best move Fox is from (1, 0) to (0, 1) with value of 10</p>	<p><i>Gra kończy się zwycięstwem wilka!</i></p>

Badania zostały przeprowadzone dla trzech scenariuszy:

- Gracz będący wilkiem wykonuje losowe ruchy, a gracz będący owcą rusza się korzystając z algorytmu minimax.
- Gracz będący owcą wykonuje losowe ruchy, a gracz będący wilkiem rusza się korzystając z algorytmu minima.
- Obaj gracze używają algorytmu minimax.

Po wstępnych testach okazało się, że dla głębokości 1 i 2 funkcja oceniająca pozycję jest zbyt dokładna i praktycznie powtarza się cały czas ta sama gra, ponieważ nie występuje przypadek dwóch lub więcej najlepszych ruchów o równej wartości, więc element losowości nie oddziaływał na wynik. W związku z tym przeprowadzone zostały dodatkowe testy dla oceny pozycji, która tylko i wyłącznie zwraca 10 w przypadku wygranej wilka i -10 w przypadku wygranej owiec.

Zostało przyjęte założenie, że zawsze pierwszy rusza się wilk. Z uwagi na rosnącą wykładniczo złożoność obliczeniową pomiary zostały ograniczone do głębokości równej 5. Powyżej głębokości 6 obiekty na planszy nie przesuwają się płynnie i zbyt dużo czasu zajęłaby analiza 100 gier.

Tabela 1. Obaj gracze używają minmaxa, ale przy ocenie planszy uwzględniony jest jedynie warunek wygranej.

Liczba gier	Głębokość	Startowa kolumna wilka	Winrate wilka[%]	Winrate owiec[%]	Średnia liczba ruchów na grę
100	1	0	83	17	45,28
100	1	2	80	20	42,03
100	2	0	100	0	45,65
100	2	2	98	2	45,74
100	3	0	94,5	5,5	44,38
100	3	2	93	7	43,75
100	4	0	98	2	42,61
100	4	2	99	1	44,12
100	5	0	90	10	36,36
100	6	2	92	8	38,38

Na podstawie wyników zawartych w Tabeli 1. można stwierdzić, że przy prostej ocenie pozycji, sprawdzającej jedynie wystąpienie wygranej, zdecydowanym faworytem jest wilk. Nie powinno to być zaskoczeniem. Wydaje się, że wilkowi łatwiej jest dojść do linii owiec niż owcom złapać wilka. Warto zwrócić uwagę na średnią liczbę ruchów na grę. Wartość ta jest sumą liczby wykonanych ruchów obu graczy. Podczas gry, w której zaczyna gracz będący wilkiem, maksymalnie może zostać wykonanych $(4 \cdot 7 + 1) \cdot 2 = 100$ ruchów, co oznacza, że gry były stosunkowo długie.

Bardzo ciekawe wyniki prezentują się w Tabeli 2., zawierającej wyniki badań przeprowadzonych dla obu graczy korzystających z minimaxa. Jak wcześniej już zostało wspomniane gry dla głębokości 1 i 2 w stu procentach wskazują na szybką wygraną wilka. Dla większej głębokości, gry stają się dłuższe i czasami to owce częściej wygrywają.

Tabela 2. Przypadek gdy obaj gracze używają minimaxa, a ocena pozycji uwzględnia wszystkie rzeczy.

Liczba gier	Głębokość	Startowa kolumna wilka	Winrate wilka[%]	Winrate owiec[%]	Średnia liczba ruchów na grę
100	1	0	100	0	13
100	1	2	100	0	13
100	1	4	100	0	13
100	1	6	100	0	13
100	2	0	100	0	21
100	2	2	100	0	29
100	2	4	100	0	13
100	2	6	100	0	13
100	3	0	76	24	42,11
100	3	2	81	19	43,27
100	3	4	100	0	26,72
100	3	6	100	0	25,96
100	4	0	30	70	43,93
100	4	2	39	61	43,39
100	4	4	20	80	42,54
100	4	6	19	81	42,41
100	5	0	46	54	42,17
100	5	2	35	65	42,63
100	5	4	74	26	35,57
100	5	6	60	40	38,1
100	6	0	35	65	45,85

Gdy gracz będący wilkiem korzysta z minimaxa, to jedynie dla głębokości 1 pojawiają się możliwe wygrane owiec. Czasami losowe ruchy mogą akurat być tymi najlepszymi.

Tabela 3. Przypadek gdy owce ruszają losowo, a wilk korzysta z minimaxa.

Liczba gier	Głębokość	Startowa kolumna wilka	Winrate wilka[%]	Winrate owiec[%]	Średnia liczba ruchów na grę
100	1	0	93	7	17,81
100	1	2	95	5	19,63
100	2	0	100	0	18,72
100	2	2	100	0	19,57
100	3	0	100	0	14,92
100	3	2	100	0	16,12
100	4	0	100	0	16,32
100	4	2	100	0	16
100	5	0	100	0	18,68
100	5	2	100	0	17,72

Tabela 4. Przypadek gdy wilk rusza losowo, owce korzystają z minimaxa..

Liczba gier	Głębokość	Startowa kolumna wilka	Winrate wilka[%]	Winrate owiec[%]	Średnia liczba ruchów na grę
100	1	0	56	44	43,81
100	1	2	53	47	45,52
100	2	0	39	61	46,95
100	2	2	52	48	46,77
100	3	0	22	78	41,8
100	3	2	25	75	40,93
100	4	0	6	94	42,03
100	4	2	7	93	42,28
100	5	0	4	96	40,19
100	5	2	2	98	41,17

W przypadku gdy wilk rusza losowo, a owce korzystają z minimaxa, kolosalna przewaga po stronie owiec zaczyna się przy głębokości równej 4 i 5. Dla głębokości równej 3 owce przegrywają co czwartą grę. Dla głębokości 1 i 2 uzyskane wyniki wskazują na prawie równą szansę na wygraną obu graczy.

W żadnej z czterech tabel nie została zauważona zależność wygranej od startowego pola z paroma wyjątkami, ale to raczej kwestia optymalizacji funkcji oceny pozycji.

Wnioski

Na podstawie przeprowadzonych badań można uznać, że gra nie jest ustawiona. Każdy z graczy ma szansę wygrać (jeśli gramy z ograniczoną głębokością, doliczenie do samego końca prawdopodobnie wykazałoby wygraną owiec).

Oczywiście można znacznie ulepszyć ocenę pozycji w zaimplementowanym algorytmie minimax przeprowadzając więcej eksperymentów, aby poprawić wagi ocenianych przypadków lub poprzez dodanie nowych ocen np. punktowanie znanych „patternów”. Innym możliwym usprawnieniem mogłoby być zastosowanie przycinania alfa-beta lub jego ulepszenie „Principal Variation Search”.

Rozbieżność w wynikach może świadczyć o nieoptymalizowanej funkcji oceny pozycji. Dobranie odpowiednich parametrów przy ocenie pozycji jest swego rodzaju sztuką, gdyż gra powinna być tak zoptymalizowana, aby wilk był syty i owca cała 😊.