

Sprawozdanie

Wprowadzenie do sztucznej inteligencji - ćwiczenie 3

Prowadzący: dr inż. Rafał Biedrzycki

Wykonała: Aleksandra Majewska (310832)

Grupa: 103

Zadanie

Zaimplementować algorytm min-max z przycinaniem alfa-beta. Algorytm ten należy zastosować do gry w proste warcaby (checkers/draughts). Niech funkcja oceny planszy zwraca różnicę pomiędzy stanem planszy gracza a stanem przeciwnika. Za pion przyznajemy 1 punkt, za damkę 10 p.

Zasady gry (w skrócie: wszyscy ruszają się po 1 polu. Pionki tylko w kierunku wroga, damki w dowolnym) z następującymi modyfikacjami:
bicie nie jest wymagane
dozwolone jest tylko pojedyncze bicie (bez serii).

checkers_stud.py zostało uzupełnione o potrzebne algorytmy i umożliwia grę użytkownik vs. AI. W **checkers_no_vis.py** znajduje się implementacja bez GUI, która umożliwia przeprowadzenie gier AI vs. AI. W pliku **stats.py** znajdują się algorytmy testujące.

Rozgrywkę rozpoczynają białe.

Pytania

- **Czy gracz sterowany przez AI zachowuje się rozsądnie z ludzkiego punktu widzenia? Jeśli nie to co jest nie tak?**
- **Niech komputer gra z komputerem (bez wizualizacji), zmieniamy parametry jednego z oponentów, badamy jak zmiany te wpłyną na liczbę jego wygranych. Należy zbadać wpływ:**
 - **Głębokości drzewa przeszukiwań**
 - **Alternatywnych funkcji oceny stanu, np.:**

- nagrody jak w wersji podstawowej + nagroda za stopień zwartości grupy (jak wszyscy blisko siebie to OK, no chyba, że da się coś zabrać przeciwnikowi);
- za każdy pion na własnej połowie planszy otrzymuje się 5 nagrody, na połowie przeciwnika 7, a za każdą damkę 10;
- za każdy nasz pion otrzymuje się nagrodę w wysokości: $(5 + \text{numer wiersza, na którym stoi pion})$ (im jest bliżej wroga tym lepiej), a za każdą damkę: 10;
-

Gracz vs. AI

Przy głębokości drzewa przeszukiwań do 5 AI zachowuje się jak przeciętny gracz. Przy większych wartościach tego parametru, trudno z nim wygrać. Nietypowe w porównaniu z normalnym graczem jest to, że sztuczna inteligencja przy wszystkich testach rozpoczynała grę takim samym ruchem.

AI vs. AI w zależności od głębokości drzewa przeszukiwań

Zmianie ulegają głębokości drzew przeszukiwań obu graczy. Przyjmują one wartości jak w nagłówku i pierwszej kolumnie tabeli.

```

VICI@LAPTOP-QNEN701:~/WS1/CWS/WS1-1a05$ ./bin/
w - wygrana białych
b - wygrana niebieskich
t - remis
  
```

white/blue	1	2	3	4	5
1	b	t	t	t	t
2	w	t	t	t	t
3	w	t	t	t	t
4	w	t	t	t	t
5	w	w	t	t	t

W większości przypadków jest remis. Występuje delikatna przewaga gracza białego. Wynika ona prawdopodobnie z tego, że rozpoczyna on grę. Ciekawym wyjątkiem jest sytuacja, gdy głębokości obu graczy są równe 1 i wygrywa gracz niebieski.

Alternatywne funkcje oceny stanu

0 - normalna ewaluacja - 10 pkt. za damkę 1 pkt. za pion;

1 - za każdy nasz pion otrzymuje się nagrodę w wysokości: $(5 + \text{numer wiersza, na którym stoi pion})$ (im jest bliżej wroga tym lepiej), a za każdą damkę: 10;

2 - za każdy pion na własnej połowie planszy otrzymuje się 5 nagrody, na połowie przeciwnika 7, a za każdą damkę 10;

Jest to opcja **2** w funkcji **evaluate**.

Przy równym

Głębokość drzewa przeszukiwań to w każdym przypadku 3.

```
w - wygrana białych
b - wygrana niebieskich
t - remis
```

white/blue	0	1	2
0	t	t	t
1	t	t	t
2	w	t	t

Wyniki doświadczenia wskazują, że wykorzystanie algorytmu nr 2 daje białemu przewagę.