

WSI21Z Piotr Szmurło

Ćw 3. (7 pkt), data oddania: do 22.11.2021 - Dwuosobowe gry deterministyczne

Zaimplementować grę w kółko i krzyżyk na planszy 3x3 z użyciem algorytmu Minimax oraz przycinania alfa-beta.

Przyjęta funkcja heurystyczna oceniająca sytuację w grze z wykładu: suma liczby punktów za pola zgodnie z macierzą:

$$\begin{bmatrix} 3 & 2 & 3 \\ 2 & 4 & 2 \\ 3 & 2 & 3 \end{bmatrix}$$

Gracz max zaczyna.

D _{max}	D _{min}	alfa-beta (max)	alfa-beta (min)	Zwycięzca	Przeszukane stany (max)	Przeszukane stany (min)
9	9	Nie	nie	Remis	557512	60708
9	9	Nie	Tak	Remis	557512	4796
9	3	Tak	Nie	Remis	37816	594
3	9	Nie	Tak	Min	964	4593
1	0	Nie	Tak	Max	48	18
3	3	Nie	Nie	Min	964	593
2	9	Tak	Tak	Remis	190	4537
3	9	Tak	Tak	Min	548	4593

Wnioski i obserwacje:

W niektórych przypadkach komputer o głębokości D = 2 remisuje podczas gdy dla D = 3 przegrywa, np. dwa ostatnie wiersze tabeli. Wynika to z przyjętej heurystyki.

Dla 1000 gier:

D_{max} = 2; D_{min} = -1 (losowo): 40 remisów, 0 wygranych gracza min (losowego)

D_{max} = 3; D_{min} = -1 (losowo): 26 remisów, 3 wygranych gracza min (losowego)

Komputer o małej głębokości przeszukiwania jest w stanie przegrać z losowo grającym komputerem. (Heurystyka nie zawsze wskaże dobrą decyzję)

Przycinanie alfa-beta nie zmienia decyzji komputera, a znacznie zmniejsza ilość przeszukiwanych stanów.

Algorytm jest optymalny, gdy przeciwnik gra optymalnie.

Czy zaczynając zawsze tak samo (i z tymi samymi ustawieniami) przebieg rozgrywki jest deterministyczny?

Tak, rozgrywka przebiega tak samo w przypadku tych samych ustawień; minimax nie jest algorytmem uczącym się.

Czy można wygrać z komputerem? Jeżeli tak to kiedy?

Można wygrać z komputerem, jeśli głębokość przeszukiwania nie jest duża.

Przy optymalnej grze w kółko i krzyżyk wynikiem zawsze jest remis. Dla $D=9$ (maks. liczba ruchów) gracz nigdy nie przegra, bo przeanalizował każde możliwe rozwiązanie – już dla pierwszego ruchu obliczona wypłata wskazuje na remis (obojętnie które pole wybierze), zakładając, że przeciwnik gra optymalnie. Algorytm w takim wypadku ignoruje funkcję heurystyczną i wybiera pierwsze wolne pole spośród tych o najlepszej wypłacie, co może nie być optymalne w przypadku, gdy przeciwnik nie zagra optymalnie. W takim przypadku może doprowadzić do remisu zamiast do zwycięstwa. Usprawnieniem mogłaby być heurystyka wybierająca ruch spośród tych, które są sobie równe co do wypłaty.

Dla dużych głębokości przeszukiwania gra zawsze kończy się remisem.

Czy jeżeli wiemy, że przegramy/zremisujemy (nie wygramy) to czy taka gra może sprawiać przyjemność ;)?

Raczej nie, chyba że przegrywamy z napisanym przez siebie programem 😊