

# WSI – sprawozdanie

Pryimak Andrii-Stepan 336173

## Algorytm Min-Max

### Wstęp

Celem zadania było zaimplementowanie algorytmu Min-Max z przycinaniem alfa-beta, który ma zostać zastosowany do gry w warcaby. Zostało to wykonane na podstawie określonych zasad gry oraz funkcji oceny stanu planszy, która ma uwzględniać różne aspekty rozgrywki, takie jak liczba pionów, damki oraz położenie na planszy.

### Pytanie

**Czy gracz sterowany przez AI zachowuje się rozsądnie z ludzkiego punktu widzenia?  
Jeśli nie, to co jest nie tak?**

Algorytm sterujący grą nie zawsze zachowuje się rozsądnie z ludzkiego punktu widzenia, mimo że jego głównym celem jest maksymalizacja funkcji oceny. Problem polega na tym, że funkcja oceny nie zawsze dokładnie odzwierciedla to, co gracz ludzki uznaje za "dobrą" pozycję na planszy.

AI stara się maksymalizować wynik funkcji oceny w każdym kroku, ale sama funkcja oceny opiera się na prostych zasadach, takich jak liczba pionków, ich pozycja na planszy, obecność damki czy jej położenie. Algorytm dąży do szybkiego zdobywania punktów, ponieważ funkcja oceny premiuje takie ruchy.

Często takie podejście prowadzi do wyboru "obejść", które niekoniecznie sprzyjają wygranej w dłuższym okresie. Przykładem może być sytuacja, w której algorytm decyduje się na szybkie zdobycie punktów za pionki na połowie przeciwnika, ignorując przy tym potencjalne zagrożenia w przyszłych turach. Choć z punktu widzenia funkcji oceny takie ruchy mogą wydawać się korzystne, w rzeczywistości mogą prowadzić do narażenia się na łatwe pułapki lub osłabienie swojej pozycji.

Też problemem często jest zakończenia partii ponieważ kiedy na planszy zostaje mniej pająków a głębokość nie jest wystarczająca dla obliczenia jakiegoś polepszenia pozycji to algorytm strzela bo dla każdego ruchu funkcja zwraca jednakowe wartościowanie.

## Analiza wyników

### Wpływ głębokości drzewa przeszukiwań.

Głębokość białych = 5

Głębokość czarnych	Wygrane białych	Remisy	Przegrane
1	5	0	0
2	5	0	0
3	4	0	1
4	3	0	2
5	2	0	3
6	0	1	4
7	0	0	5

### Inne funkcje oceny

Pionowo algorytm białych

Notacja: Wygrana białych/ remis / wygrana czarnych

	basic_ev	group_prize	push_to_opp_half	push_forward
basic_ev	2/0/3	1/1/3	1/0/4	1/0/4
group_prize	2/0/3	1/1/3	2/0/3	2/1/2
push_to_opp_half	4/0/1	0/0/5	2/1/2	1/1/3
push_forward	3/0/2	1/0/4	2/0/3	2/0/2

Funkcja group\_prize preferuje naprzykład się trzymać razem a na końcu zwykle wygrywa często ponieważ zostawia pionki w pierwszej linijce tablicy i oponent nie może wygrać i partia się kończy bo oponent nie ma ruchów

Push\_to\_op\_half lub push\_forward preferuje bicie do przodu nap bicie do tyły i zwykle nie biję do tyły kiedy ma możliwość ponieważ otrzyma -2 punkty za zmniejszenia zmiennej row

Basik\_env dziga ogólnie dobrze ale często lub nie bije i myśli że zbije potem lub jeśli ma dużo bić to czasem jeśli funkcja oceny dla dużo pozycji jednakowa to można powiedzieć że strzela bo dla basic funkcja oceny nie bardzo się zmienia w porównaniu do innych

## Podsumowanie

Kluczowym elementem sukcesu algorytmu jest odpowiedni dobór funkcji oceny oraz głębokości przeszukiwania. Jeśli funkcja oceny nie odzwierciedla dobrze rzeczywistości i jej powiązania z wygraną, algorytm zamiast dążyć do zwycięstwa, maksymalizuje wartość funkcji oceny, co często prowadzi do suboptymalnych decyzji.

Przykładem takiego zachowania może być sytuacja, w której algorytm preferuje ruchy prowadzące do szybkiego zdobycia damki, nawet kosztem utraty kilku pionków. Dla człowieka taka strategia może wydawać się zbyt ryzykowna i nieprzemyślana, ponieważ w dłuższej perspektywie może prowadzić do osłabienia pozycji i przegranej.

Dla funkcji oceny `push_forward` lub `push_opponent_half` jest nadmiernie agresywne, powoduje, że algorytm forsuje ruchy do przodu, ignorując fakt, że raz przesunięty pionek nie może się wycofać. Takie "bezargumentowe pushowanie" pionków jest często błędem strategicznym.

Moim zdaniem funkcja oceny przyznająca 10 punktów za damkę jest zbyt przesadzona, co prowadzi do nierównowagi w decyzjach AI. Dodatkowo algorytm z funkcją oceny premiuje agresywne ruchy traci w długoterminowej perspektywie, gdyż nie uwzględnia konieczności budowania solidnej pozycji na planszy. W efekcie, program gra agresywnie, co często kończy się niepotrzebnymi stratami pionków i osłabieniem pozycji na planszy.