

# Wprowadzenie do sztucznej inteligencji | cw. 3

Damian D'Souza

W mojej implementacji algorytmu minimax, aby zachowanie komputera było bardziej różnorodne, zastosowałem losowy wybór ruchu spośród tych o takiej samej ocenie.

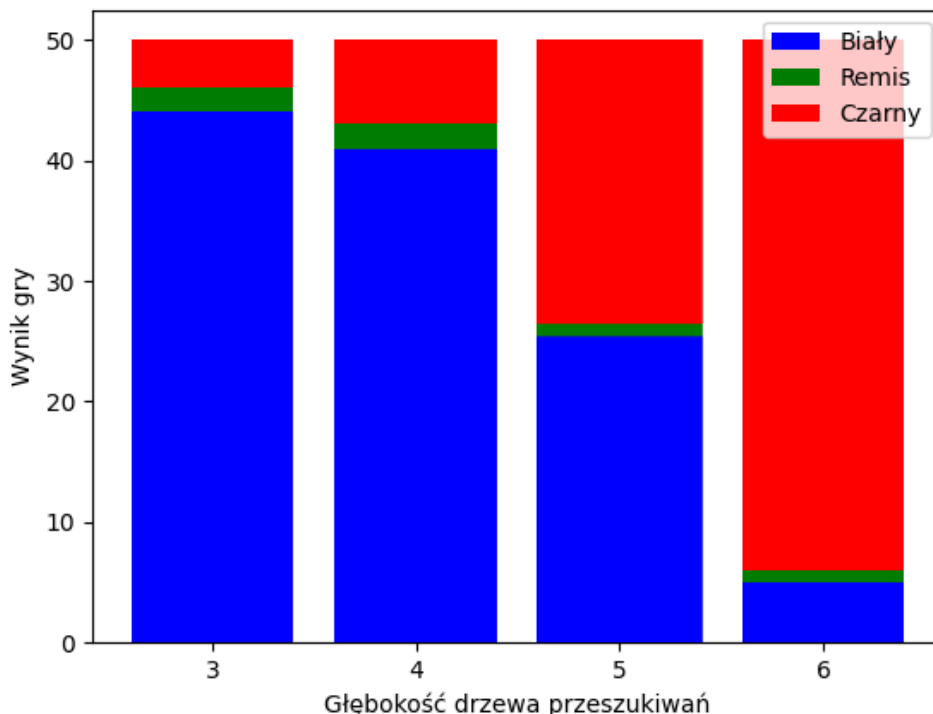
Czy gracz sterowany przez AI zachowuje się rozsądnie z ludzkiego punktu widzenia? Jeśli nie to co jest nie tak?

Gracz sterowany przez komputer nie zawsze zachowuje się rozsądnie – w niektórych sytuacjach, mimo że nic mu nie grozi, nie przesuwają pionka, który mógłby stać się damką, ani nie bije wrogiego pionka. Zastosowanie losowości w wyborze ruchu miejscami pomaga, ponieważ często ruchy rozsądne są oceniane równie wysoko, jak te mniej rozsądne.

## Wpływ czynników na działanie algorytmu

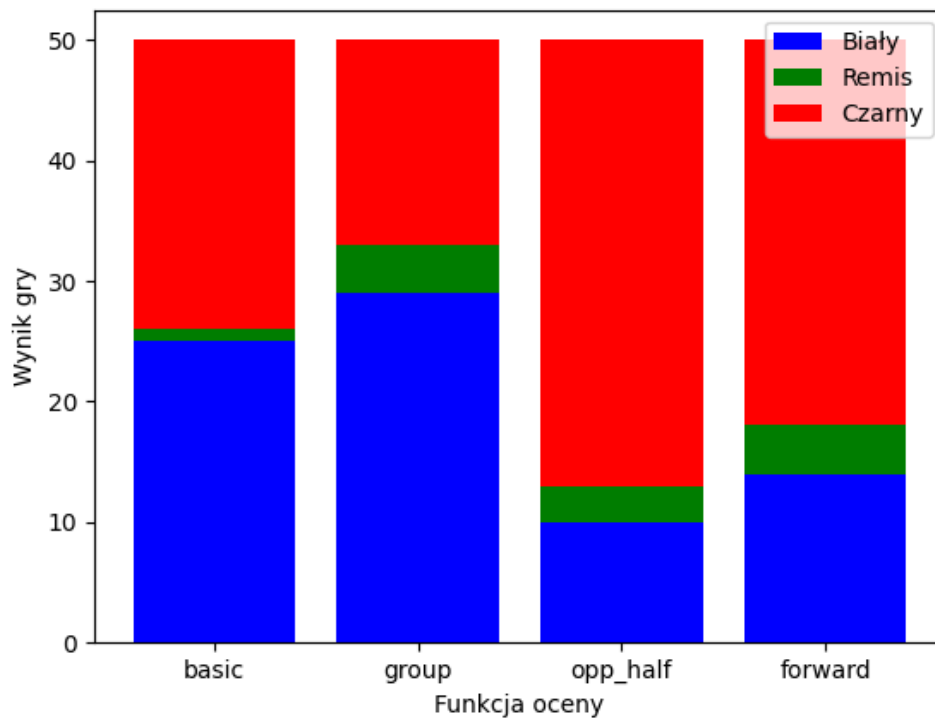
W badaniach zmieniane były parametry gracza czarnego

### 1. Wpływ głębokości drzewa przeszukiwań



Jak widać na wykresie, zmniejszenie głębokości znacząco pogarsza wyniki gracza czarnego. Przy takich samych głębokościach obaj gracze osiągają podobne wyniki, natomiast po zwiększeniu głębokości do 6 drugi gracz wygrywa zdecydowaną większość gier. Jest to jednak osiągnięte kosztem znacznie dłuższego czasu obliczeń dla każdego ruchu.

## 2. Wpływ funkcji oceny stanu

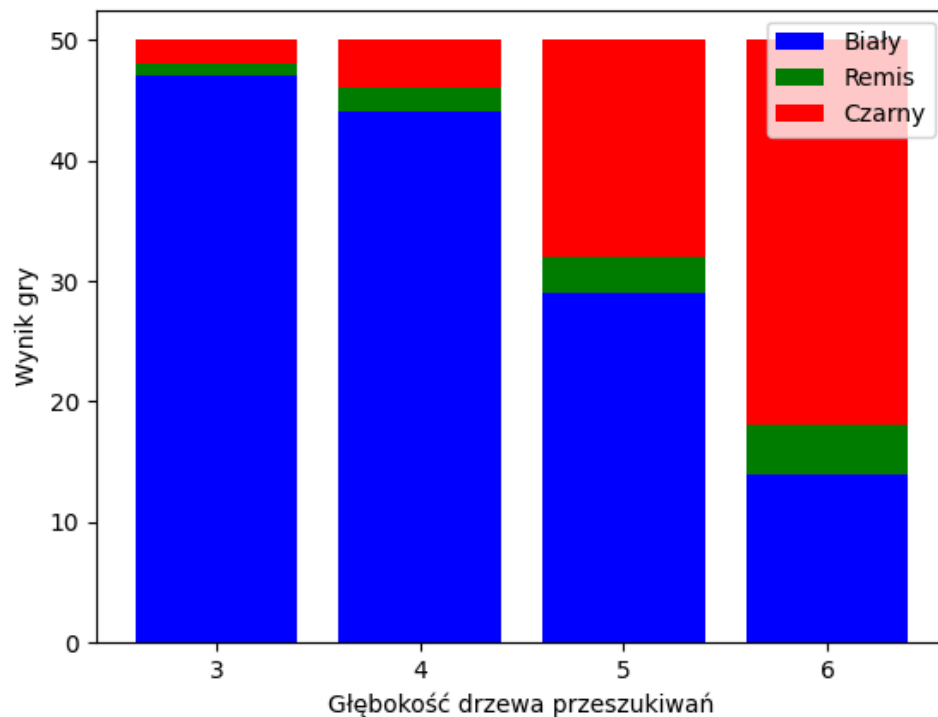


Zastosowanie funkcji faworyzującej zwartość grupy pogorszyło wyniki gracza czarnego oraz zwiększyło liczbę remisów. Może to wynikać z ograniczonej agresji ze strony czarnego, którego funkcja ewaluacyjna bardziej nagradzała trzymanie pionków razem niż bicie wrogich pionków.

Funkcje opp\_half i forward znacząco poprawiły wyniki gracza czarnego. Funkcja opp\_half okazała się skuteczniejsza, prawdopodobnie dlatego, że w przypadku funkcji forward pionki pod koniec gry miały tendencję do trzymania się krawędzi planszy. W niektórych sytuacjach prowadziło to do powtarzania ruchów, co mogło zakończyć się remisem lub wygraną białych, jeśli mieli więcej damek lub pionków.

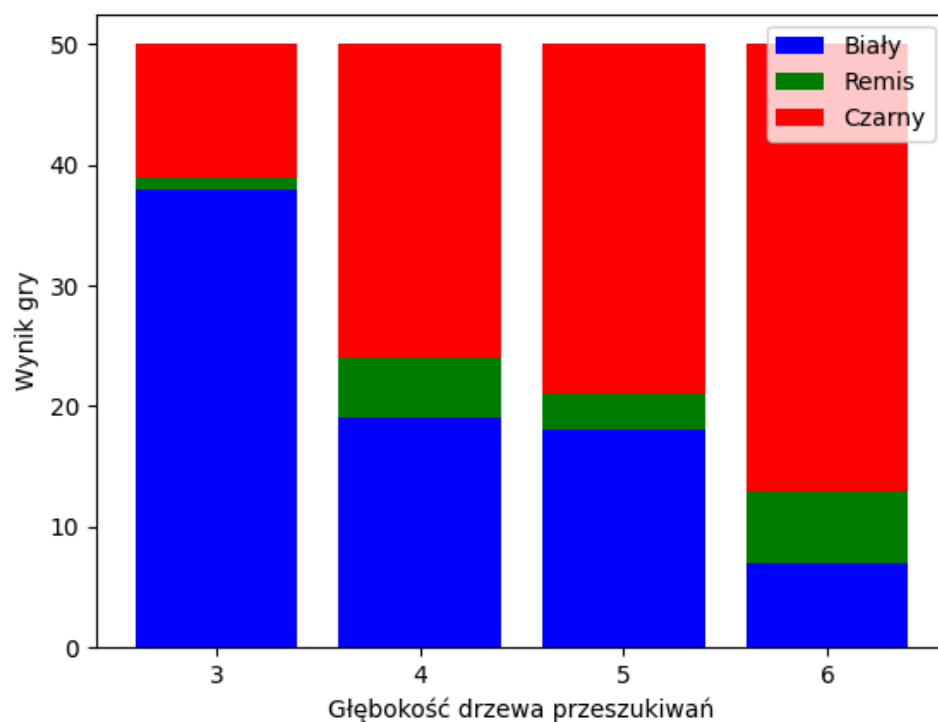
Na koniec aby znaleźć najlepszą funkcję oceny wykonałem symulacje gier z różnymi połączeniami głębokości i funkcji.

#### Funkcja group



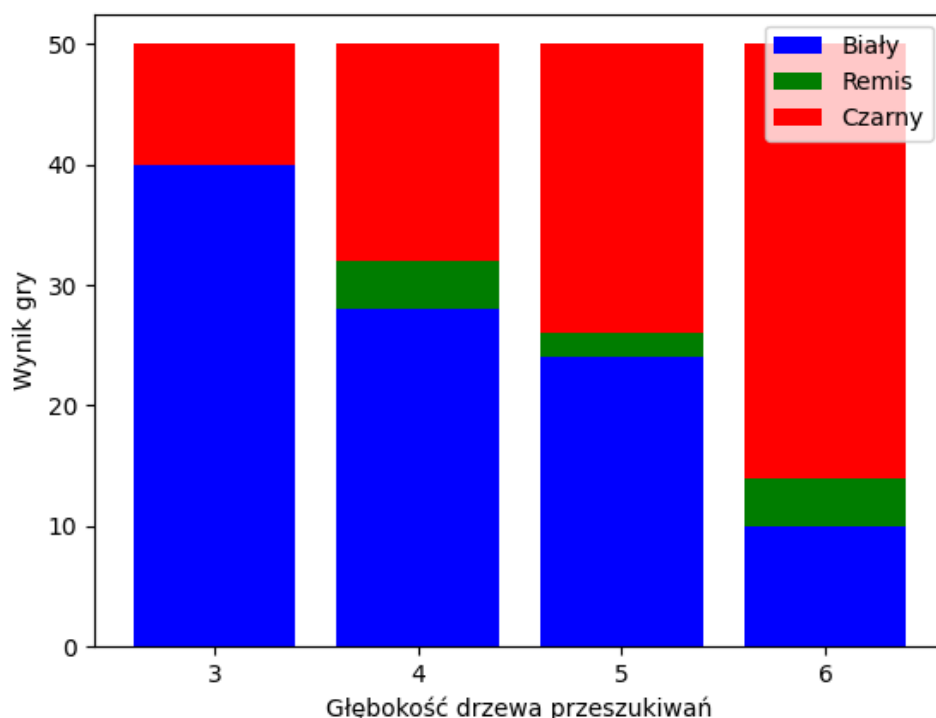
W przypadku danej funkcji również zauważalny jest trend wzrostu jakości wraz ze zwiększeniem głębokości, jednak wyniki są widocznie gorsze niż dla podstawowej funkcji oceny.

#### Funkcja opp\_half



Po zastosowaniu tej funkcji zaobserwowano znaczącą poprawę – nawet przy mniejszej głębokości wyniki są porównywalne do wyników innych funkcji przy głębokości równej 5.

### Funkcja forward



Wyniki są bardzo zbliżone do tych uzyskanych dla funkcji `opp_half` – gracz czarny przegrał tylko nieznacznie więcej gier. Najbardziej zauważalną różnicą jest spadek jakości gry przy głębokości równej 4.

### Wniosek

Algorytm minimax wykazuje dużą zależność od doboru funkcji oceny oraz parametrów, takich jak głębokość przeszukiwania. Z przeprowadzonych badań wynika, że najlepsza okazała się funkcja `opp_half`, która zapewniła najlepsze wyniki spośród testowanych funkcji. Jej istotną zaletą jest możliwość zmniejszenia głębokości przeszukiwania bez znaczącej utraty jakości gry, co pozwala na bardziej efektywne wykorzystanie zasobów komputera. Dobór odpowiednich parametrów i funkcji oceny jest kluczowy dla optymalizacji działania algorytmu.