

Minimax Algorithm

Opis i implementacja z przycinaniem alfa-beta w grze kółko i krzyżyk

Wstęp

Algorytm Minimax jest klasyczną metodą podejmowania decyzji w grach dwuosobowych o sumie zerowej, w których gracze wykonują ruchy na przemian, a stan gry jest w pełni obserwowalny. Jego koncepcja jest oparta na twierdzeniu minimax von Neumanna (1928), rozwiniętego później jako algorytm przeszukiwania ¹ i do dziś stanowi fundament wielu systemów grających w gry planszowe, takie jak szachy, warcaby czy właśnie kółko i krzyżyk.

1. Opis algorytmu

1.1 Drzewo gry

Minimax analizuje drzewo gry – strukturę reprezentującą wszystkie możliwe sekwencje ruchów od aktualnego stanu planszy. Węzły drzewa odpowiadają kolejnym pozycjom w grze, a krawędzie – możliwym ruchom. Liście drzewa to stany końcowe (np. mat w szachach, zapełniona plansza w kółko i krzyżyk) lub węzły osiągnięte po wyczerpaniu zadanej maksymalnej głębokości przeszukiwania.

W odróżnieniu od przeglądu wyczerpującego, algorytm Minimax w praktycznych implementacjach nie buduje pełnego drzewa gry – stany potomne są analizowane niezależnie, bez łączenia identycznych stanów wynikających z różnych sekwencji ruchów.

1.2 Założenia

W klasycznym wariacie dla gry dwuosobowej o sumie zerowej zakładamy, że:

1. gracz *MAX* wybiera ruch maksymalizujący wartość pozycji,
2. gracz *MIN* wybiera ruch minimalizujący tę wartość,
3. obaj gracze grają optymalnie.

1.3 Funkcja oceny i heurystyka

Algorytm Minimax wymaga funkcji oceny stanu, która przypisuje każdemu węzłowi drzewa wartość określającą jakość danej pozycji. Dla stanów terminalnych analizowanego drzewa wartość ta jest dokładnie znana:

$$\infty - \text{wygrana } MAX, \quad 0 - \text{remis}, \quad -\infty - \text{wygrana } MIN.$$

Dla pozostałych węzłów, osiągniętych po wyczerpaniu zadanej maksymalnej głębokości przeszukiwania, wartość jest określana heurystyczną funkcją oceny.

¹Wikipedia – Algorytm min-max

Przykładem heurystyki wykorzystanej dla gry w kółko i krzyżyk jest analiza wszystkich możliwych linii wygrywających oraz ocena stopnia ich zajęcia przez obu graczy.

Dla każdej linii o długości n rozpatruje się liczbę pól zajętych przez gracza MAX oraz MIN. Na tej podstawie przypisywana jest wartość punktowa:

- $+n$ – MAX ma $n - 1$ pól w linii, a MIN żadnego (pozycja niemal wygrywająca),
- $-n$ – MIN ma $n - 1$ pól w linii, a MAX żadnego,
- $+1$ – MAX ma $n - 2$ pola w linii, a MIN żadnego,
- -1 – MIN ma $n - 2$ pola w linii, a MAX żadnego,
- 0 – w pozostałych przypadkach (linia zablokowana lub mało istotna).

Dodatkowo heurystyka uwzględnia zjawisko **forka**, czyli sytuację, w której gracz posiada co najmniej dwie linie wygrywające w następnym ruchu. Ponieważ przeciwnik nie może zablokować obu jednocześnie, taka pozycja otrzymuje bardzo wysoką wartość (aproxymującą wygraną), ponieważ w praktyce prowadzi do zwycięstwa w następnym ruchu przy dostatecznej głębokości przeszukiwania.

Heurystyka ta premiuje więc wygrywające struktury. Ostateczna wartość heurystyki jest sumą ocen wszystkich linii z uwzględnieniem powyższego warunku dominującego.

Przykładowe oceny dla planszy 3×3 :

O		O
	X	
X		O

$$h \approx \infty$$

MAX ma wygraną (fork)

O	O	
	X	

$$h = 4 - 2 = 2$$

MAX ma przewagę

X	X	
	O	

$$h = 2 - 4 = -2$$

MIN ma przewagę

O	O	O
X		
X		

$$h = \infty$$

Wygrana gracza MAX

X	X	X
	O	
O		

$$h = -\infty$$

Wygrana gracza MIN

O	X	O
O	X	X
X	O	O

$$h = 0$$

Remis

1.4 Przebieg rekurencji

Algorytm wybiera ruch zapewniający najlepszy wynik, analizując drzewo gry do określonej głębokości i zakładając optymalną grę przeciwnika:

1. dla stanu końcowego lub węzła na maksymalnej głębokości zwracana jest jego ocena,
2. w węźle gracza MAX wybierana jest **maksymalna** wartość spośród dzieci,
3. w węźle gracza MIN wybierana jest **minimalna** wartość spośród dzieci,
4. wartość w korzeniu wskazuje jakość najlepszej decyzji, a odpowiadający jej ruch jest wykonywany.

2. Przycinanie alfa-beta

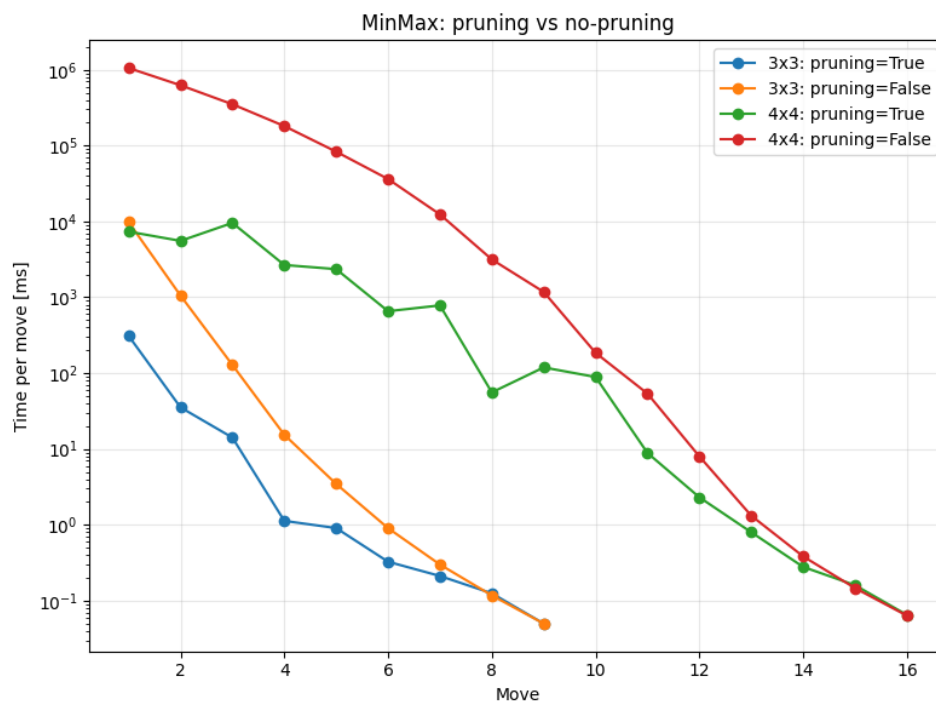
Przycinanie alfa-beta jest optymalizacją algorytmu Minimax, która pozwala pominąć analizę części gałęzi drzewa gry, jeżeli wiadomo, że nie mogą one wpłynąć na ostateczną decyzję. Wynik końcowy (wybrany ruch) pozostaje taki sam jak w klasycznym Minimax, ale liczba ocenianych węzłów jest zwykle znacznie mniejsza.

W trakcie przeszukiwania utrzymywane są dwie granice:

- α – najlepsza (najwyższa) wartość znaleziona dotąd dla gracza *MAX*,
- β – najlepsza (najniższa) wartość znaleziona dotąd dla gracza *MIN*.

Jeżeli w pewnym momencie zachodzi warunek $\alpha \geq \beta$, dalsza eksploracja bieżącej gałęzi jest przerywana, ponieważ optymalnie grający przeciwnik i tak nie dopuści do wyboru tego wariantu.

Poniżej przedstawiono wykres porównujący średni czas wykonania ruchu dla algorytmu Minimax z przycinaniem alfa-beta oraz bez niego, dla plansz 3×3 oraz 4×4 :



Na podstawie wykresu można wyciągnąć następujące wnioski:

- Przycinanie alfa-beta znacząco redukuje czas obliczeń, szczególnie w początkowych fazach gry, gdzie liczba możliwych ruchów jest największa.
- Różnica między wersją z przycinaniem i bez jest znacznie bardziej widoczna dla planszy 4×4 , gdzie drzewo gry jest znacznie większe niż dla planszy 3×3 .
- W miarę postępu gry, czas obliczeń maleje dla obu wariantów, co wynika ze zmniejszającej się liczby dostępnych pól na planszy.
- Dla końcowych ruchów różnice między wariantami zacierają się, ponieważ drzewo gry jest już bardzo płytkie i liczba możliwych stanów jest niewielka.

3. Ocena gracza Minimax

W tej sekcji porównano zachowanie gracza Minimax z dwoma innymi typami graczy: **losowym** oraz **ludzkim**. Ocena ma charakter jakościowy i opiera się na analizie przebiegu partii oraz scenariuszy testowych (wygranie, blokowanie, remis).

3.1 Porównanie z graczem losowym

Gracz losowy wybiera dowolne wolne pole bez analizy konsekwencji. W efekcie:

- Minimax bardzo dobrze wykorzystuje błędy przeciwnika i często przechodzi do szybkiej wygranej,
- praktycznie nie oddaje pozycji wygrywających, jeśli są one widoczne na analizowanej głębokości,
- nawet gdy nie może wygrać natychmiast, dąży do pozycji bezpiecznej (co najmniej remisowej).

Typowy schemat gry przeciw losowemu przeciwnikowi wygląda następująco: Minimax buduje przewagę pozycyjną (zajęcie centrum lub rogu), a następnie zamyka linię albo tworzy jednocześnie dwa zagrożenia, których przeciwnik nie potrafi poprawnie zablokować.

3.2 Porównanie z graczem ludzkim

W starciu z człowiekiem zachowanie Minimax jest bardziej interesujące:

- przeciw osobie grającej niedokładnie Minimax jest zwykle skuteczniejszy, ponieważ natychmiast karze błędy taktyczne,
- przeciw osobie grającej poprawnie strategicznie wynik często kończy się remisem,
- Minimax nie „ryzykuje” ruchów spekulacyjnych: wybiera wariant o najwyższej gwarantowanej wartości, więc jego styl jest stabilny i przewidywalny.

W praktyce człowiek ma przewagę głównie wtedy, gdy algorytm działa z małą głębokością przeszukiwania. Przy większej głębokości Minimax eliminuje większość pułapek i taktycznych przeoczeń.

3.3 Minimax kontra Minimax

W rozgrywce dwóch graczy Minimax (ta sama głębokość, te same reguły oceny) wynik jest deterministyczny i zazwyczaj remisowy. Potwierdzają to testy automatyczne dla planszy 3×3 oraz 4×4 , gdzie przy grze „idealnej kontra idealna” żaden z graczy nie uzyskuje trwałej przewagi.

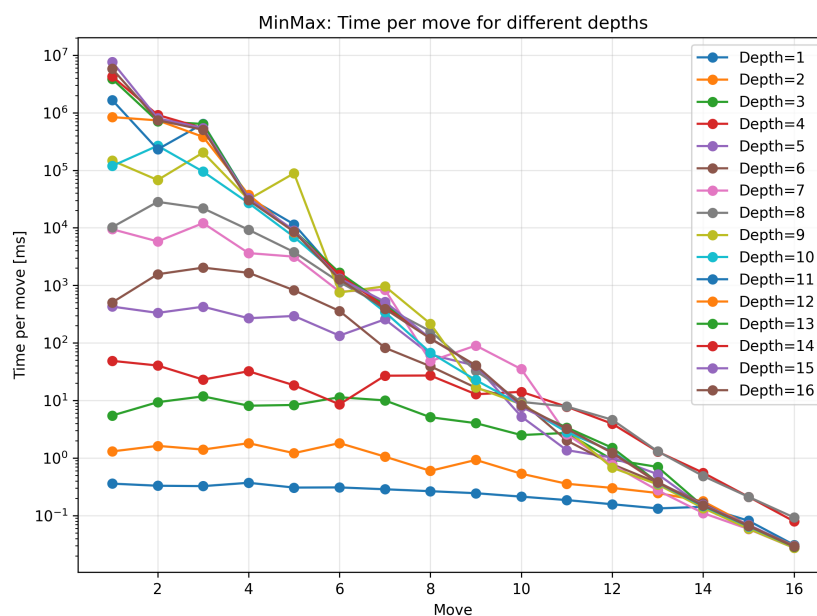
Taki rezultat jest zgodny z teorią gier dla kółka i krzyżyka: przy optymalnej grze obu stron pozycja początkowa jest pozycją remisową.

4. Wpływ głębokości przeszukiwania

Głębokość przeszukiwania jest jednym z kluczowych parametrów algorytmu Minimax, który bezpośrednio wpływa zarówno na jakość podejmowanych decyzji, jak i czas ich obliczania. Zwiększanie głębokości pozwala uwzględnić dalsze konsekwencje ruchów, jednak wiąże się z rosnącym kosztem obliczeniowym.

Czas a głębokość

Dla planszy 4×4 rozegrano partie Minimax kontra Minimax z przycinaniem alfa-beta, gdzie obaj gracze korzystali z tej samej głębokości z zakresu $1 \dots 16$. Dla każdego ruchu mierzono czas jego obliczenia i zaprezentowano na wykresie poniżej.

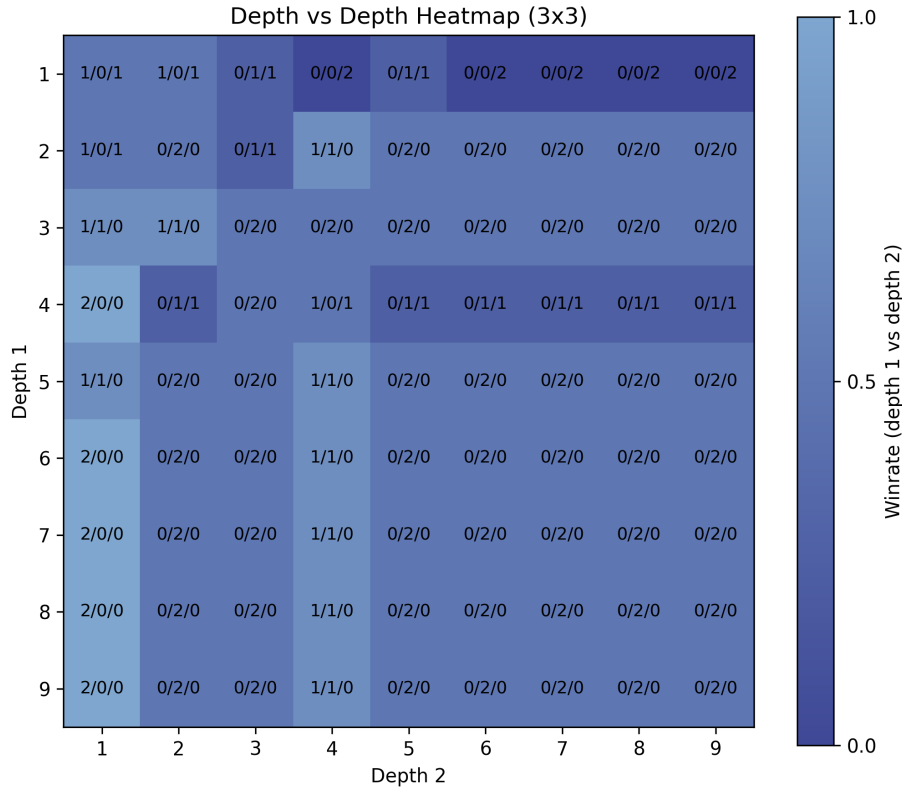


Wnioski:

- Czas wykonania ruchu rośnie wykładniczo z głębokością – na początku partii różnica między $d=1$ a $d=16$ wynosi około od ~ 1 ms do ~ 2.8 h na ruch.
- Dla każdej głębokości czas monotonicznie maleje wraz z numerem ruchu. Jest to bezpośredni skutek kurczenia się przestrzeni poszukiwań: każdy wykonany ruch zmniejsza liczbę wolnych pól, co redukuje analizowane drzewo gry i skraca czas rekurencji.
- Pod koniec partii (ruchy 13–16) krzywe dla wszystkich głębokości zbiegają się do podobnych wartości rzędu 10^{-1} ms – przy niemal pełnej planszy głębokość przestaje mieć znaczenie, bo drzewo gry jest trywialne.
- Widoczne lokalne wzrosty czasu (np. ruch 4–5 dla wysokich głębokości) wynikają prawdopodobnie z tego, że w tych pozycjach drzewo jest jeszcze silnie rozgałęzione, a przycinanie alfa-beta jest mniej skuteczne niż w pozycjach bardziej zdeterminowanych.

Głębokość a jakość gry

Dla planszy 3×3 zestawiono wszystkie pary głębokości z zakresu $1 \dots 9$. Dla każdej pary rozegrano dwie partie (ze zmianą kolejności ruchu), a wyniki zapisano w formacie *wygrane/remisy/porażki*.



Wnioski:

- Wraz ze wzrostem głębokości rośnie jakość gry: płytkie strategie przegrywają, ponieważ nie dostrzegają odpowiedzi przeciwnika ani nie potrafią planować sekwencji kilku ruchów do przodu.
- Dla głębokości $d \geq 5$ rozgrywki w większości przypadków dążą do remisów, co jest zgodne z teorią gry w kółko i krzyżyk – przy optymalnej grze obu stron pozycja startowa jest remisowa.
- Wynik może zależeć od kolejności ruchu, szczególnie dla małych głębokości. Wynika to z ograniczonego horyzontu analizy: gracz rozpoczynający uzyskuje inicjatywę, której przeciwnik przy płytkim przeszukiwaniu nie jest w stanie w pełni zneutralizować.
- Zaobserwowano anomalię dla $d = 4$: głębokość ta osiąga gorsze wyniki zarówno od $d = 2$, jak i od $d \geq 5$, mimo że głębokości sąsiednie ($d = 3$ oraz $d = 5$) radzą sobie lepiej. Najbardziej prawdopodobnym wyjaśnieniem jest efekt parzystości głębokości – przy parzystym d ostatni poziom przeszukiwania przypada na ruch przeciwnika, przez co funkcja oceny jest wywoływana w stanie „z jego perspektywy”. W połączeniu z aproksymacyjnym charakterem heurystyki (opartym na wykrywaniu lokalnych zagrożeń, takich jak fork) może to prowadzić do błędnych preferencji ruchów w pozycjach, w których taktyczne odpowiedzi znajdują się tuż za horyzontem przeszukiwania. Ta hipoteza wymagałaby dalszej analizy np. przesłedzenia kilku gier.

Podsumowanie

W pracy przedstawiono implementację algorytmu Minimax z przycinaniem alfa-beta dla gry kółko i krzyżyk na planszach 3×3 oraz 4×4 . Przeprowadzona analiza jakościowa oraz czasowa potwierdziła zgodność działania programu z teorią gier: przy odpowiednio dużej głębokości przeszukiwania gracz Minimax podejmuje decyzje prowadzące co najmniej do remisu, natomiast przeciw słabszemu przeciwnikowi skutecznie wykorzystuje jego błędy.

Najważniejsze wnioski:

- Przycinanie alfa-beta znacząco redukuje czas obliczeń w porównaniu do klasycznego Minimax, szczególnie we wczesnych fazach gry oraz dla większych plansz.
- Zwiększanie głębokości przeszukiwania poprawia jakość gry jedynie do pewnego poziomu; po osiągnięciu strategii zbliżonej do optymalnej dalsze zwiększanie głębokości wpływa głównie na wzrost kosztu obliczeniowego, bez istotnej poprawy jakości decyzji.
- Czas podejmowania decyzji jest największy w początkowej fazie gry i maleje wraz z jej postępem, co wynika z systematycznego zmniejszania się przestrzeni stanów.
- Jakość wyników zależy nie tylko od głębokości przeszukiwania, ale również od jakości heurystyki oraz kolejności rozpatrywania ruchów, co bezpośrednio wpływa na skuteczność przycinania alfa-beta.

Ograniczenia rozwiązania:

- Dla większych plansz złożoność obliczeniowa pozostaje wysoka mimo zastosowania przycinania alfa-beta, co sprawia, że pełne przeszukiwanie szybko staje się niepraktyczne.
- Zastosowana heurystyka może prowadzić do efektu horyzontu oraz lokalnych błędów oceny, szczególnie przy małych i średnich głębokościach przeszukiwania.
- Implementacja nie wykorzystuje dodatkowych technik optymalizacyjnych, takich jak tablice transpozycji czy bazy otwarcie lub zamknięcie.
- W eksperymentach zaobserwowano anomalię dla głębokości $d = 4$, wynikającą najprawdopodobniej z interakcji parzystości głębokości, ograniczonego horyzontu przeszukiwania oraz aproksymacyjnego charakteru funkcji oceny.

Podsumowując, algorytm Minimax z przycinaniem alfa-beta stanowi skuteczną i przewidywalną metodę dla gier deterministycznych o pełnej informacji. Jego praktyczna efektywność zależy jednak od kompromisu pomiędzy głębokością przeszukiwania, jakością heurystyki oraz dostępnym czasem obliczeń. W przyszłości rozwiązanie może zostać rozszerzone o dodatkowe techniki optymalizacyjne, takie jak tablice transpozycji, zaawansowane porządkowanie ruchów czy iteracyjne pogłębianie przeszukiwania, co pozwoliłoby znacząco zwiększyć efektywność algorytmu dla większych przestrzeni stanów. Dodatkowo możliwe jest zastosowanie metod uczenia maszynowego do ulepszenia funkcji oceny pozycji, co mogłoby poprawić jakość podejmowanych decyzji w sytuacjach trudnych do uchwycenia przez prostą heurystykę.