Sztuczna inteligencja i inżynieria wiedzy laboratorium

Sprawozdanie 3

*Algorytmy rozwiązywania gier*

Arkadiusz Marcinowski

228160

W8, Informatyka

1. **Wstęp**

Celem zadanie było zapoznanie się z wyszukiwania strategii gry w jej drzewie stanów. Cel obejmował zapoznanie się z podstawowymi pojęciami związanymi z teorią gier (gra, strategia, drzewo gry, stan, przestrzeń stanów). Zapoznanie się z podstawowymi algorytmami oraz przebadanie ich działania dla gry w Stratego: min-max oraz alfa-beta.

1. **Zasady gry Stratego**

Stratego to gra turowa, ruchy wykonywane są naprzemiennie przez dwóch graczy.

Zdobywanie punktów:

* Zaznaczony cały wiersz
* Zaznaczona cała kolumna
* Zaznaczona dowolna przekątna o długości większej niż 1

Punktów odpowiadającą ilości zaznaczonych w jednym z powyższych przypadków prostych otrzymuje gracz, który jako ostatni wstawił pionek zamykający prostą.

Zamykając więcej niż jedną prostą w jednym ruchu, punkty się sumują, np. zamykając kolumnę i wiersz jednym pionkiem, na planszy 5x5, gracz otrzymuje 5+5 punktów.

1. **Algorytm MinMax, AlphaBeta**
   1. **Nieparzysta liczba pól gry ‘49’**

Algorytmy zostały porównane na planszy wielkości 7x7. Zmieniana była jedynie głębokość przeszukiwania dla algorytmów oraz algorytm który zaczynał rozgrywkę.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pierwszy ruch: | TAK | |  | |
| Głębokość | minMax | | alphaBeta | |
| czas [ms] | pkt. | czas [ms] | pkt. |
| 1 | 0 | 102 | 0 | 90 |
| 2 | 20 | 97 | 0 | 95 |
| 3 | 116 | 104 | 40 | 88 |
| 4 | 2325 | 105 | 466 | 87 |
| 5 | 80978 | 118 | 4041 | 74 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pierwszy ruch: |  | | TAK | |
| Głębokość | minMax | | alphaBeta | |
| czas [ms] | pkt. | czas [ms] | pkt. |
| 1 | 0 | 90 | 0 | 102 |
| 2 | 6 | 95 | 4 | 97 |
| 3 | 101 | 88 | 81 | 104 |
| 4 | 2056 | 87 | 560 | 105 |
| 5 | 72995 | 74 | 4551 | 118 |

Przedstawione powyżej zestawiania pokazują, że algorytm AlphaBeta jest o wiele bardziej wydajny niż algorytm MinMax. Czas jego działania jest zdecydowanie mniejszy. Oba zestawienia są swoimi odbiciami w zależności od zaczynającego grę algorytmu. Potwierdza to jedynie, że algorytm AlphaBeta działa jak MinMax, z pewną heurystyką ograniczającą czas jego działania.

Należy również zaznaczyć, że zaoszczędzony czas algorytmu AlphaBeta może być zrównoważony przez głębsza przeszukiwanie, co zwiększa efektywność działania algorytmu.

**Przewaga algorytmu zaczynającego rozgrywkę jest jednak tak znacząca z uwagi na nieparzystą ilość pól gry.**

* 1. **Parzysta liczba pól gry ‘36’**

Algorytmy zostały porównane na planszy wielkości 6x6. Zmieniana była jedynie głębokość przeszukiwania dla algorytmów oraz algorytm który zaczynał rozgrywkę.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pierwszy ruch: | TAK | |  | |
| Głębokość | minMax | | alphaBeta | |
| czas [ms] | pkt. | czas [ms] | pkt. |
| 1 | 1 | 65 | 1 | 75 |
| 2 | 5 | 61 | 5 | 79 |
| 3 | 48 | 74 | 30 | 66 |
| 4 | 701 | 68 | 162 | 72 |
| 5 | 13229 | 62 | 870 | 78 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| SIZE = 12 |  |  |  |  |
| Pierwszy ruch: |  | | TAK | |
| Głębokość | minMax | | alphaBeta | |
| czas [ms] | pkt. | czas [ms] | pkt. |
| 1 | 1 | 75 | 1 | 65 |
| 2 | 5 | 79 | 5 | 61 |
| 3 | 42 | 66 | 27 | 74 |
| 4 | 527 | 72 | 166 | 68 |
| 5 | 10959 | 78 | 1041 | 62 |

Przedstawione powyżej zestawiania pokazują, że algorytm AlphaBeta jest o wiele bardziej wydajny niż algorytm MinMax. Czas jego działania jest zdecydowanie mniejszy. Oba zestawienia są swoimi odbiciami w zależności od zaczynającego grę algorytmu. Potwierdza to jedynie, że algorytm AlphaBeta działa jak MinMax, z pewną heurystyką ograniczającą czas jego działania.

Należy również zaznaczyć, że zaoszczędzony czas algorytmu AlphaBeta może być zrównoważony przez głębsza przeszukiwanie, co zwiększa efektywność działania algorytmu.

**Większe szanse na wygraną ma algorytm zaczynający rozgrywkę jako drugi, z uwagi na to, że drzewo rozgrywki jest już ograniczone przez ruch przeciwnika.**

1. **MinMax, AlphaBeta, heurystyki kolejności wyszukiwania**

Algorytmy zostały porównane na planszy wielkości 7x7 oraz 6x6. Zmieniana była jedynie głębokość przeszukiwania dla algorytmów oraz algorytm który zaczynał rozgrywkę.

**4.1. Proponowane heurystyki**

**Heurystyka Narożna** sortuje zestaw możliwych ruchów w taki sposób, by najpierw sprawdzane były pola które leżą najdalej od środka planszy.

**Heurystyka Środka** sortuje zestaw możliwych ruchów w taki sposób, by najpierw sprawdzane były pola które leżą najbliżej środka planszy.

**4.2. Nieparzysta liczba pól gry ‘49’**

**Przewaga algorytmu zaczynającego rozgrywkę jest jednak tak znacząca z uwagi na nieparzystą ilość pól gry.**

**4.2.1. Heurystyki przeciwko czystym algorytmom**

W tym przypadku nie występuje powtarzalność punktów, zróżnicowana jedynie tym, który gracz zaczyna. Zaczynający gracz wciąż w większości przypadków wygrywa.

Porównując z wynikami z *punktu 3:*

* Heurystyka Narożna zwiększa szansę na wygraną.
* Heurystyka Środkowa zmniejsza szansę na wygraną.
* Algorytmy oparte na heurystykach działają dłużej z uwagi na złożoność obliczeniową sortowania możliwych ruchów

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pierwszy ruch: | TAK | |  | |  | Pierwszy ruch: |  | | TAK | | |
| Heurystyka: | Narożna | |  | |  | Heurystyka: | Narożna | |  | | |
| Głębokość | minMax | | alphaBeta | |  | Głębokość | minMax | | alphaBeta | | |
| czas [ms] | pkt. | czas [ms] | pkt. |  | czas [ms] | pkt. | czas [ms] | pkt. | |
| 1 | 52 | 110 | 4 | 82 |  | 1 | 62 | 98 | 0 | 94 | |
| 2 | 67 | 112 | 10 | 80 |  | 2 | 68 | 78 | 5 | 114 | |
| 3 | 190 | 111 | 55 | 81 |  | 3 | 212 | 82 | 91 | 110 | |
| 4 | 2609 | 120 | 557 | 72 |  | 4 | 2270 | 77 | 629 | 115 | |
| 5 | 86956 | 120 | 3294 | 72 |  | 5 | 80126 | 77 | 3576 | 115 | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
| Pierwszy ruch: | TAK | |  | |  | Pierwszy ruch: |  | | TAK | | |
| Heurystyka: | Środkowa | |  | |  | Heurystyka: | Środkowa | |  | | |
| Głębokość | minMax | | alphaBeta | |  | Głębokość | minMax | | alphaBeta | | |
| czas [ms] | pkt. | czas [ms] | pkt. |  | czas [ms] | pkt. | czas [ms] | pkt. | |
| 1 | 52 | 108 | 0 | 84 |  | 1 | 53 | 97 | 0 | 95 | |
| 2 | 60 | 103 | 4 | 89 |  | 2 | 75 | 97 | 8 | 95 | |
| 3 | 185 | 122 | 59 | 70 |  | 3 | 196 | 81 | 73 | 111 | |
| 4 | 2475 | 129 | 454 | 63 |  | 4 | 2279 | 73 | 533 | 119 | |
| 5 | 87187 | 104 | 3147 | 88 |  | 5 | 76939 | 78 | 3718 | 114 | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
| Pierwszy ruch: | TAK | |  | |  | Pierwszy ruch: |  | | TAK | | |
| Heurystyka: |  | | Narożna | |  | Heurystyka: |  | | Narożna | | |
| Głębokość | minMax | | alphaBeta | |  | Głębokość | minMax | | alphaBeta | | |
| czas [ms] | pkt. | czas [ms] | pkt. |  | czas [ms] | pkt. | czas [ms] | pkt. | |
| 1 | 0 | 94 | 60 | 98 |  | 1 | 1 | 82 | 51 | 110 | |
| 2 | 6 | 114 | 66 | 78 |  | 2 | 8 | 80 | 67 | 112 | |
| 3 | 120 | 110 | 130 | 82 |  | 3 | 113 | 81 | 146 | 111 | |
| 4 | 2455 | 115 | 594 | 77 |  | 4 | 2220 | 72 | 648 | 120 | |
| 5 | 86976 | 115 | 3234 | 77 |  | 5 | 78195 | 72 | 3905 | 120 | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |  |  | |
| Pierwszy ruch: | TAK | |  | |  | Pierwszy ruch: |  | | TAK | |
| Heurystyka: |  | | Środkowa | |  | Heurystyka: |  | | Środkowa | |
| Głębokość | minMax | | alphaBeta | |  | Głębokość | minMax | | alphaBeta | |
| czas [ms] | pkt. | czas [ms] | pkt. |  | czas [ms] | pkt. | czas [ms] | pkt. | |
| 1 | 0 | 95 | 57 | 97 |  | 1 | 1 | 84 | 50 | 108 | |
| 2 | 5 | 95 | 59 | 97 |  | 2 | 8 | 89 | 59 | 103 | |
| 3 | 106 | 111 | 112 | 81 |  | 3 | 120 | 70 | 136 | 122 | |
| 4 | 2388 | 119 | 542 | 73 |  | 4 | 2191 | 63 | 627 | 129 | |
| 5 | 85724 | 114 | 3341 | 78 |  | 5 | 75266 | 88 | 3446 | 104 | |

**4.2.2. Heurystyki przeciwko heurystykom**

Obserwując wyniki, można zauważyć potwierdzenie wcześniejszego spostrzeżenia mówiącego że Heurystyka Narożna poprawia działanie algorytmu. Dla wszystkich testów, przy głębokości przeszukiwania równej 1, algorytm z heurystyką narożną wygrywał pojedynek, niezależnie od tego kto zaczynał rozgrywkę. Nie miało to miejsca przy czystych algorytmach.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pierwszy ruch: | TAK | |  | |  | Pierwszy ruch: |  | | TAK | |
| Heurystyka: | Narożna | | Środkowa | |  | Heurystyka: | Narożna | | Środkowa | |
| Głębokość | minMax | | alphaBeta | |  | Głębokość | minMax | | alphaBeta | |
| czas [ms] | pkt. | czas [ms] | pkt. |  | czas [ms] | pkt. | czas [ms] | pkt. |
| 1 | 57 | 114 | 5 | 78 |  | 1 | 6 | 102 | 57 | 90 |
| 2 | 62 | 118 | 6 | 74 |  | 2 | 9 | 93 | 69 | 99 |
| 3 | 196 | 112 | 68 | 80 |  | 3 | 134 | 70 | 142 | 122 |
| 4 | 2514 | 131 | 511 | 61 |  | 4 | 2250 | 83 | 687 | 109 |
| 5 | 87065 | 104 | 3898 | 88 |  | 5 | 75563 | 88 | 4498 | 104 |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Pierwszy ruch: | TAK | |  | |  | Pierwszy ruch: |  | | TAK | |
| Heurystyka: | Środkowa | | Narożna | |  | Heurystyka: | Środkowa | | Narożna | |
| Głębokość | minMax | | alphaBeta | |  | Głębokość | minMax | | alphaBeta | |
| czas [ms] | pkt. | czas [ms] | pkt. |  | czas [ms] | pkt. | czas [ms] | pkt. |
| 1 | 58 | 90 | 2 | 102 |  | 1 | 2 | 78 | 71 | 114 |
| 2 | 67 | 99 | 7 | 93 |  | 2 | 7 | 74 | 72 | 118 |
| 3 | 221 | 122 | 81 | 70 |  | 3 | 111 | 80 | 147 | 112 |
| 4 | 2508 | 109 | 469 | 83 |  | 4 | 2140 | 61 | 652 | 131 |
| 5 | 87542 | 104 | 3761 | 88 |  | 5 | 77593 | 88 | 3947 | 104 |

**4.3. Parzysta liczba pól gry ‘36’**

**Większe szanse na wygraną ma algorytm zaczynający rozgrywkę jako drugi, z uwagi na to, że drzewo rozgrywki jest już ograniczone przez ruch przeciwnika.**

**4.3.1. Heurystyki przeciwko czystym algorytmom**

W tym przypadku nie występuje powtarzalność punktów, zróżnicowana jedynie tym, który gracz zaczyna. Zaczynający gracz wciąż w większości przypadków wygrywa.

Porównując z wynikami z *punktu 3:*

* Heurystyka Narożna zwiększa szansę na wygraną.
* Heurystyka Środkowa zmniejsza szansę na wygraną.
* Algorytmy oparte na heurystykach działają dłużej z uwagi na złożoność obliczeniową sortowania możliwych ruchów

**4.3.2. Heurystyki przeciwko heurystykom**

Obserwując wyniki, można zauważyć potwierdzenie wcześniejszego spostrzeżenia mówiącego że Heurystyka Narożna poprawia działanie algorytmu. Dla wszystkich testów, przy głębokości przeszukiwania równej 1, algorytm z heurystyką narożną wygrywał pojedynek, niezależnie od tego

1. **Heurystyki oceny stanu gry**

**5.1. Heurystyka losowa**

Heurystyka ocenia stan gry przez brak analizy jego stanu i wybór losowy.

W tym przypadku nigdy nie wygrywa algorytm losowy. Osiąga on jednak zaskakująco wysokie wyniki, jeśli zaczyna rozgrywkę. Wynika to z faktu, iż plansza 7x7 zawiera 49 pól, więc zaczynający algorytm wykonuje jeden ruch więcej.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pierwszy ruch: | TAK | |  | |  | Pierwszy ruch: |  | | TAK | |
| Głębokość | minMax | | random | |  | Głębokość | minMax | | random | |
| czas [ms] | pkt. | czas [ms] | pkt. |  | czas [ms] | pkt. | czas [ms] | pkt. |
| 1 | 65 | 127 | 1 | 65 |  | 1 | 55 | 114 | 2 | 78 |
| 2 | 66 | 143 | 0 | 49 |  | 2 | 66 | 112 | 1 | 80 |
| 3 | 178 | 118 | 0 | 74 |  | 3 | 192 | 104 | 1 | 88 |
| 4 | 2557 | 131 | 1 | 61 |  | 4 | 2284 | 112 | 1 | 80 |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Pierwszy ruch: | TAK | |  | |  | Pierwszy ruch: |  | | TAK | |
| Głębokość | alphaBeta | | random | |  | Głębokość | alphaBeta | | random | |
| czas [ms] | pkt. | czas [ms] | pkt. |  | czas [ms] | pkt. | czas [ms] | pkt. |
| 1 | 67 | 142 | 1 | 50 |  | 1 | 56 | 112 | 0 | 80 |
| 2 | 65 | 126 | 0 | 66 |  | 2 | 68 | 109 | 1 | 83 |
| 3 | 144 | 129 | 1 | 63 |  | 3 | 146 | 118 | 2 | 74 |
| 4 | 745 | 116 | 3 | 76 |  | 4 | 794 | 137 | 1 | 55 |

1. **Podsumowanie**

Algorytm MinMax oraz AlphaBeta są bardzo ciekawymi sposobami rozwiązania przedstawionego zadania. Algorytmy znacznie różnią się od siebie czasem przeszukiwania drzewa gry.

Badania dla planszy gry o 49 polach pokazują, że zaczynający gracz ma przewagę. Wynika to z faktu, że wykonuje jeden ruch więcej, co zapewnia mu więcej punktów.

Dla planszy gry o 36 polach, wygrywa zwykle gracz zaczynający jako drugi. Wynika to z faktu, że dzięki wcześniejszemu ruchowi przeciwnika, otrzymuje już ograniczone drzewo rozgrywki.

**Tutaj jeszcze o heurystykach.**