

Projektowanie Algorytmów i Metody Sztucznej Inteligencji

Kółko i krzyżyk

Prowadzący: Mgr inż. Marcin Ochman

Autor: Adrian Sobecki

Numer indeksu: 248942

Termin zajęć: Wtorek 15:15-16:55

Termin oddania: 05.06.2020

Wstęp

Naszym zadaniem była implementacja wybranej przez nas gry w oparciu o strategię Minimax. Wybrałem grę kółko i krzyżyk. Początkowo zaimplementowałem rozgrywkę dla dwóch graczy i w ten sposób testowałem poprawność algorytmu sprawdzającego zwycięstwo. Po poprawieniu błędów nastąpił czas na napisanie algorytmu Minimax.

Opis wybranego algorytmu

Początkowo wybrałem implementację algorytmu Minimax bez zastosowania cięć alfa-beta. Przy testach szybko okazało się, że drzewo problemu rozrasta się bardzo szybko wraz ze wzrostem wielkości planszy do gry. Wykorzystanie cięć alfa-beta przyspieszyło działanie mojego algorytmu.

Algorytm Minimax polega na wygenerowaniu drzewa możliwych realizacji rozgrywki. Następnie dla każdego możliwego ruchu zwraca jego użyteczność pod względem dążenia do wygranej. Wybierany jest oczywiście ten ruch, który najbardziej przybliży nas do zwycięstwa. Cięcia alfa-beta są rozszerzeniem algorytmu Minimax, polegają na obcięciu gałęzi drzewa, które nie rokuje lepszym wynikiem.

Sposób oceny

Algorytm zwraca wartość użyteczności każdego ruchu w zależności od głębokości drzewa. Im szybciej możemy uzyskać zwycięstwo tym wartość tego ruchu jest większa. Jeżeli po naszym ruchu nie osiągamy ani zwycięstwa ani remisu wyznaczana jest użyteczność danego stanu. W moim algorytmie gdy po wykonanym ruchu przeciwnik może zwyciężyć jest zwracana bezużyteczność stanu, w przeciwnym wypadku funkcja zwraca wartość równą zero.

Głębokość drzewa

Głębokość przeszukiwań, na którą możemy sobie pozwolić zależy od wielkości planszy. Testy pomogły mi oszacować największe głębokości mojego algorytmu dla poszczególnych rozmiarów.

- a) rozmiar ≤ 4 – głębokość równa 6,
- b) rozmiar = 5 – głębokość równa 5,
- c) rozmiar ≤ 7 – głębokość równa 4,
- d) rozmiar > 7 – głębokość równa 3.

Faktem jest, że algorytm dla rozmiaru planszy ≥ 7 oraz głębokości równej 3 może powodować dłuższe działanie. Dlatego przy starcie programu, przed deklaracją rozmiaru zamieściłem informację, iż powinna ona zostać wybrana z przedziału [3-7].

Poglądowa realizacja algorytmu

najlepszy_ruch(kogo_ruch, czy_max)

najlepszy_wynik = -infinity;

dla każdej wolnej pozycji

plansza[i] = kogo_ruch;

wynik = minimax(0, czy_max, $-\infty$, ∞);

plansza[i] = ' ';

jeżeli wynik > najlepszy_wynik

najlepszy_wynik = wynik;

najlepszy_ruch = i;

zwróć najlepszy ruch

minimax(glebokosc, czy_max, alfa, beta)

jeżeli (czy_max==true i wygrana 'O') **zwróć** -200+glebokosc*10

jeżeli (czy_max==false i wygrana 'X') **zwróć** 200-glebokosc*10

jeżeli (remis) **zwróć** 0

jeżeli (głębokość>niż zadana) **zwróć** ocenę stanu

jeżeli (czy_max==true)

najlepszy_wynik = -infinity;

dla każdej wolnej pozycji

plansza[i] = 'X';

wynik = minimax(glebokosc + 1, false, alfa, beta);

plansza[i] = ' ';

najlepszy_wynik = max(wynik, najlepszy_wynik);

alfa = max(wynik, alfa);

jeżeli (alfa >= beta) **zwróć** najlepszy_wynik;

zwróć najlepszy_wynik;

jeżeli (czy_max==false)

najlepszy_wynik = infinity;

dla każdej wolnej pozycji

plansza[i] = 'O';

wynik = minimax(glebokosc + 1, false, alfa, beta);

plansza[i] = ' ';

najlepszy_wynik = min(wynik, najlepszy_wynik);

beta = min(wynik, beta);

jeżeli (alfa >= beta) **zwróć** najlepszy_wynik;

zwróć najlepszy_wynik;

Opis gry

Standardowa gra w kółko i krzyżyk. Gracz lub jego przeciwnik wygrywają gdy jeden z nich osiągnie określoną ilość swoich znaków w kolumnie/rzędzie/skose.

a) Rozgrywka 3x3

```
$ ./projekt4.exe
GRA KOLKO I KRZYZYK
WYBIERZ ROZMIAR PLANSZY [3-7]:
3
WYBIERZ ILOSC W RZEDZIE DO WYGRANEJ:
3
KTO ZACZYNA?
'1' - GRACZ
'2' - KOMPUTER
1
  | |
--+--+
  | |
--+--+
  | |
Podaj nr. pola
4
  | |
--+--+
  | o |
--+--+
  | |
0
x | |
--+--+
  | o |
--+--+
  | |
Podaj nr. pola
1
x | o |
--+--+
  | o |
--+--+
  | |
7
x | o |
--+--+
  | o |
--+--+
  | x |
Podaj nr. pola
6
  x | o |
--+--+
    | o |
--+--+
  o | x |
Podaj nr. pola
2
  x | o | x
--+--+
    | o |
--+--+
  o | x |
Podaj nr. pola
3
  x | o | x
--+--+
  o | o |
--+--+
  o | x |
Podaj nr. pola
5
  x | o | x
--+--+
  o | o | x
--+--+
  o | x |
Podaj nr. pola
8
  x | o | x
--+--+
  o | o | x
--+--+
  o | x | o
REMIS
```

b) Rozgrywka 4x4

```
$ ./projekt4.exe
GRA KOLKO I KRZYZYK
WYBIERZ ROZMIAR PLANSZY [3-7]:
4
WYBIERZ ILOSC W RZEDZIE DO WYGRANEJ:
3
KTO ZACZYNA?
'1' - GRACZ
'2' - KOMPUTER
2
  |  |  |
--+--+--+
  |  |  |
--+--+--+
  |  |  |
--+--+--+
  |  |  |
Podaj nr. pola
5
  |  |  |
--+--+--+
  | x |  |
--+--+--+
  |  |  |
--+--+--+
  |  |  |
Podaj nr. pola
1
  | o |  |
--+--+--+
  | x |  |
--+--+--+
  |  |  |
--+--+--+
  |  |  |

6
  | o |  |
--+--+--+
  | x | x |
--+--+--+
  |  |  |
--+--+--+
  |  |  |
Podaj nr. pola
4
  | o |  |
--+--+--+
o | x | x |
--+--+--+
  |  |  |
--+--+--+
  |  |  |
7
  | o |  |
--+--+--+
o | x | x | x
--+--+--+
  |  |  |
--+--+--+
  |  |  |
WYGRAL GRACZ: X
```

c) Rozgrywka 5x5

```
$ ./projekt4.exe
GRA KOLKO I KRZYZYK
WYBIERZ ROZMIAR PLANSZY [3-7]:
5
WYBIERZ ILOSC W RZEDZIE DO WYGRANEJ:
4
KTO ZACZYNA?
'1' - GRACZ
'2' - KOMPUTER
1
| | | |
+---+
| | | |
+---+
| | | |
+---+
| | | |
+---+
| | | |

Podaj nr. pola
6
| | | |
+---+
| o | | |
+---+
| | | |
+---+
| | | |
+---+
| | | |

0
x | | | |
+---+
| o | | |
+---+
| | | |
+---+
| | | |

Podaj nr. pola
7
x | | | |
+---+
| o | o | |
+---+
| | | |
+---+
| | | |
+---+
| | | |

5
x | | | |
+---+
x | o | o | |
+---+
| | | |
+---+
| | | |
+---+
| | | |

Podaj nr. pola
3
x | | | |
+---+
x | o | o | o |
+---+
| | | |
+---+
| | | |

9
x | | | |
+---+
x | o | o | o | x
+---+
| | | |
+---+
| | | |
+---+
| | | |

Podaj nr. pola
12
x | | | |
+---+
x | o | o | o | x
+---+
| | o | |
+---+
| | | |
+---+
| | | |

10
x | | | |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
| | | |
+---+
| | | |

Podaj nr. pola
15
x | | | |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | | | |
+---+
| | | |

1
x | x | | |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | | | |
+---+
| | | |

20
x | x | | |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | | | |
+---+
o | | | |

16
x | x | | |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | x | | |
+---+
o | | | |

Podaj nr. pola
24
x | x | | |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | x | | |
+---+
o | | | | o

18
x | x | | |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | x | | x |
+---+
o | | | o

Podaj nr. pola
23
x | x | | |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | x | | x |
+---+
o | | | o | o

3
x | x | | x |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | x | | x |
+---+
o | | | o | o

Podaj nr. pola
22
x | x | | x |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | x | | x |
+---+
o | | o | o | o

2
x | x | x | x |
+---+
x | o | o | o | x
+---+
x | | o | |
+---+
o | x | | x |
+---+
o | | o | o | o

WYGRAL GRACZ: x
```

Wnioski

Strategia Minimax będzie podejmować najbardziej optymalne ruchy jeżeli drzewo przeszukiwań będzie pełne. Bardzo szybki wzrost możliwych ruchów wraz ze wzrostem rozmiaru planszy uniemożliwia nam to. Ograniczenie maksymalnej głębokości oraz zastosowanie cięć alfa-beta skraca działanie algorytmu. Im generowane drzewo jest mniejsze tym ruchy nie zawsze będą podejmowane najbardziej optymalne. Funkcja zwracająca ocenę stanu zwraca wartość danego ruchu bazując na aktualnym stanie planszy. Sposób oceniania zależy od nas i ma on istotny wpływ na przebieg rozgrywki przy większej planszy.

Literatura

[http://wazniak.mimuw.edu.pl/index.php?title=Sztuczna_inteligencja/SI_Modu%C5%82_8 -
_Gry_dwuosobowe](http://wazniak.mimuw.edu.pl/index.php?title=Sztuczna_inteligencja/SI_Modu%C5%82_8_-_Gry_dwuosobowe)

<https://www.youtube.com/watch?v=trKjYdBASyQ>

<https://www.youtube.com/watch?v=l-hh51ncgDI>

https://eduinf.waw.pl/inf/utils/002_roz/p011.php