

Sztuczna inteligencja
Egzamin
1 lipca 2022

Progi: 50p - dost, 60 – dost+, 70 – db, 85 – db+, 100 – bdb.

Zadanie 1. (48p) Mamy agenta, który porusza się po szachownicy o wymiarach $N \times N$. Agent ma pełną wiedzę o świecie. W świecie agenta obowiązują następujące zasady:

- i. W jednym kroku agent może wykonać akcję przesunięcia o jedno pole w każdym z 8 kierunków, otworzyć parasol (jeżeli jest zamknięty), zamknąć (jeżeli jest otwarty) albo nie wykonywać żadnego ruchu (odpocząć). Czyli ma do dyspozycji 10 akcji (zmień stan parasola traktujemy jako jedną akcję). Na początku gry agent *stoi* na polu (0,0)
- ii. Niektóre pola są dołami, jeżeli agent wpadnie do dołu, to przegrywa i kończy rozgrywkę (nagroda -1000)
- iii. Niektóre pola są murkami, agent nie może na nie wejść za pomocą normalnej akcji marszu
- iv. Niektóre pola są skarbami, agent, który dojdzie do pola ze skarbem automatycznie go zabiera, jeżeli teraz dojdzie do pola startowego to otrzymuje nagrodę +1000 i kończy grę.
- v. Na planszy wieje wiatr, w jednym z 8 kierunków (N, NE, E, SE, S, SW, W, NW). Wiatr nie ma wpływu na agenta, który ma zamknięty parasol. Agent może iść z zamkniętym bądź otwartym parasolem. Na początku rozgrywki wiatr wieje na północ.
- vi. Wiatr przesuwają agenta w kierunku, w którym wieje, o k pól, gdzie $k \in \{1, 2, 3, 4\}$, przy czym p_k oznacza prawdopodobieństwo, że ruch będzie dokładnie o k (czyli wszystkie p_k są dodatnie, a $p_1 + p_2 + p_3 + p_4 = 1$. Wiatr może umieścić agenta na murku (lub w dole), może go też przedmuchać nad murkiem (nad dołem)
- vii. Poza szachownicą są doły, czyli zepchnięcie agenta z planszy oznacza koniec gry.
- viii. Wiatr może nie zmieniać kierunku (z prawdopodobieństwem p_w), albo zmienić na „sąsiedni”, czyli zmienić kierunek o 45 stopni. (z prawdopodobieństwem $\frac{1}{2} \cdot (1 - p_w)$, dla każdej z dwóch możliwych zmian)
- ix. Kolejność poszczególnych faz jest następująca: ruch agenta, potencjalna zmiana kierunku wiatru, działanie wiatru. Po każdej fazie sprawdzane jest, czy agent wpadł do dziury i czy znalazł skarb.

Uwzględniając powyższy opis świata:

- a) Napisz program probabilistyczny (czyli program, który zawiera w sobie odwołania do funkcji (pseudo)losowych), symulujący działanie agenta w pełni losowego (który losuje akcje z rozkładem jednostajnym) **(8p)**
- b) Wykorzystując powyższy program, napisz program, który (dla danej instancji świata) oblicza, jakie jest prawdopodobieństwo tego, że agent z poprzedniego punktu otrzyma nagrodę (pozytywną), pod warunkiem, że kiedyś odwiedził pole o współrzędnych (5,5). **(5p)**
- c) Od tego momentu rozważamy agentów niewykonywujących ruchów losowych. Chcemy modelować ten świat za pomocą MDP (Markowskich procesów decyzyjnych). Jak sprawić, żeby agent dążył do jak najszybszego zdobycia skarbu (podać dwa sposoby zmiany nagradzania agenta) **(5p)**
- d) Czym jest algorytm A*? **(3p)** Czym jest relaksacja zadania i do czego służy? **(3p)** Jeżeli agent steruje wiatrem¹, to powyższy problem jest przeszukiwaniem przestrzeni stanów. Wymień składniki przestrzeni stanów i zaproponuj prostą, użyteczną, optymistyczną heurystykę dla tego zadania

¹Kierunek wiatru ciągle zmienia się o co najwyżej 45 stopni. Agent też decyduje, o ile ma go przesunąć wiatr (o ile ma otwarty parasol)

(wartość heurystyki powinna być obliczana w czasie stałym, czyli za pomocą programu bez pętli). (8p)

- e) Czym jest przeszukiwanie dwukierunkowe? (3p) Jak je wykorzystać w tym zadaniu (w wersji z agentem sterującym wiatrem)? (3p)
- f) Powiedzmy, że agent musi wrócić do bazy wykonując co najwyżej K ruchów. Jeżeli wróci bez skarbu, to dostaje małą ujemną nagrodę, jeżeli krok $K + 1$ zastanie go poza bazą, to umiera (nagroda taka, jak przy wpadnięciu do dołu). Jak modelować ten problem i jaki algorytm zastosować w wersji z wiatrem, który jest sterowany przez wrogów naszego agenta? (5p)
- g) W jakim sensie punkt a) może być użyteczny przy tworzeniu bardziej efektywnego programu dla punktu f)? (5p)

Zadanie 2. (22 p) Będziemy rozważać łamigłówkę Sudoku. Mamy szachownicę o wymiarach 9×9 , w każdym polu tej szachownicy należy wpisać cyfrę od 1 do 9 (niektóre już są wpisane), w taki sposób, że w każdym wierszu, w każdej kolumnie cyfry się nie powtarzają. Szachownica ta podzielona jest na 9 „podszachownic” o wymiarach 3×3 , w każdej podszachownicy cyfry również się nie mogą powtarzać.

- a) Powiedz, jakie składniki ma Problem Spełniania Więzów (CSP). Czym są te składniki dla Sudoku? (5p)
- b) Standardowy algorytm rozwiązywania CSP przeplata dwie fazy: propagacji więzów i przeszukiwań z nawrotami. Powiedz, na przykładzie Sudoku, czym jest propagacja więzów. (5p)
- c) W przeszukiwaniu z nawrotami wybieramy zmienną, której przypiszemy wartość. Zaproponuj sensowną metodę wyboru tej zmiennej, znowu na przykładzie Sudoku. (5p)
- d) Opisz, jak dla instancji Sudoku S skonstruować formułę logiczną $F(S)$, która „zakoduje” instancję S , w taki sposób, że znajdując podstawienie spełniające tę formułę możemy łatwo znaleźć rozwiązanie S . (7p)

Zadanie 3. (20 p) Zdefiniuj krótko następujące pojęcia:

- a) Warstwa sieci neuronowej
- b) Klauzula w logice zdaniowej
- c) Koniunkcyjna postać normalna
- d) Zbiór ewaluacyjny
- e) Strategia ε -zachłanna

Zadanie 4. (24 p) Odpowiedz możliwie zwięźle na poniższe pytania lub wykonaj polecenia z kolejnych punktów:

1. Dlaczego metoda Iterative Deepening może wydawać się marnotrawstwem zasobów i dlaczego mimo tego jest to ważna metoda przeszukiwania przy braku wiedzy o problemie?
2. Uzupełnij zdanie: Algorytm A^* przegląda wszystkie węzły, w których wartość _____ jest _____ niż _____.
3. Załóżmy, że gra planszowa jest na tyle prosta, lub też jesteśmy na tyle blisko jej końca, że jesteśmy w stanie wykonać algorytm MiniMax bez funkcji heurystycznej (tzn. przeanalizować całe drzewo gry). Czy tak znaleziony ruch jest optymalnym zagranie w sytuacji, kiedy wiemy, że naszym przeciwnikiem jest agent losowy? Odpowiedz krótko uzasadnij.
4. Czy algorytm k -NN może popełniać błąd na zbiorze uczącym? Jeżeli tak, to jaki warunek musi spełniać k żeby to było możliwe? (odpowiedź krótko uzasadnij)