

Sztuczna inteligencja

Ćwiczenia 3

Zajęcia w tygodniu 16-20 kwietnia

Każde zadanie warte jest 1 punkt. Zadanie z gwiazdką nie wlicza się do maksimum.

Zadanie 1. (1p★) Przeczytaj i opowiedz o tym, czym są algorytmy mrówkowe (ant colony algorithms). Powiedz, jak zastosować je do rozwiązania problemu komiwojażera.

Zadanie 2. (1p) Algorytmy BFS i DFS można traktować jako algorytmy wyznaczające plan (sekwencję działań) prowadzącą w pełni znanym (od początku) środowisku ze znanego stanu początkowego do pożądanego stanu końcowego. Przyjmijmy w tym zadaniu, że środowisko nie jest w pełni znane i że w każdym stanie agent zdobywa pewną dodatkową wiedzę (tzn. wykonuje obserwacje) i dopiero znając wyniki tych obserwacji może określić wyniki swoich działań. Innymi słowy, stan do którego przejdziemy, jest funkcją stanu w którym byliśmy, akcji oraz wyniku obserwacji, które zrobiliśmy (możemy przyjąć, że wynikiem obserwacji jest element skończonego, znanego zbioru).

Czym w takim środowisku będzie plan agenta? Napisz pseudokod (lub program) dla algorytmu DFS, który znajduje plan dla tak określonego zadania przeszukiwania.

Zadanie 3. (1p) Opisz poniższe algorytmy (możesz użyć nazw, jeżeli znasz):

- local beam search dla $k = 1$,
- local beam search z jednym początkowym stanem i bez limitu na liczbę zachowanych stanów po generacji następnika,
- symulowane wyżarzanie z $T = 0$ przez cały czas działania algorytmu,
- symulowane wyżarzanie z $T = \infty$ przez cały czas działania algorytmu,
- algorytm genetyczny z populacją wielkości 1.

Zadanie 4. (1p) Podobnie jak burze, obrazki logiczne można również rozwiązać za pomocą więzów w Prologu. Opisz, jak modelować obrazki logiczne używając więzów dostępnych w SWI-Prologu.

Zadanie 5. (1p) Rozważmy uproszczone zadanie układania planu lekcji, w którym (między innymi) nie przejmujemy się dostępnością sal.

- Mamy pewną liczbę zajęć do rozmieszczenia.
- Każde zajęcia mają następujące parametry: klasa (w znaczeniu „grupa uczniów”) oraz nauczyciel.
- Każdemu zajęciu należy przypisać termin: liczbę od 1 do 50. Liczby od 1 do 10 oznaczają poniedziałek, od 11 do 20 – wtorek, itd.
- Żadna klasa nie może mieć „okienka”, czyli przerwy w zajęciach. Czyli każdego dnia przychodzi na jakąś godzinę, ma zajęte wszystkie kolejne godziny, a potem idzie do domu i do szkoły przychodzi następnego dnia.

Przedstaw to zadanie jako problem więzowy, który da się wyrazić za pomocą konstrukcji dostępnych w SWI-Prologu. Przy definiowaniu warunków na bezokienkowość planu możesz potrzebować dodatkowych zmiennych.

Zadanie 6. (1p), ★ To zadanie jest z gwiazdką, bo dotyczy wiedzy 'nieobowiązkowej'. Adresowane jest do osób, które znają Prolog.

- Predykat `labeling` można napisać korzystając z bardziej elementarnych predykatów. Twoim zadaniem jest napisanie najprostszej wersji, w której zmienne obsługiwane są od lewej strony, w której nie przejmujemy się kolejnością wyboru wartości. Możesz używać predykatu `indomain/1`.
- Jak samodzielnie zdefiniować predykat `indomain`, korzystając z predykatu `fd_dom(+Var, -Dom)` odczytującego dziedzinę zmiennej¹.

¹W dokumentacji jest zasugerowane, jak to zrobić, ale z użyciem mechanizmu DCG. W Twoim rozwiązaniu nie powinieneś z niego korzystać.

- c) W programach prologowych, które powstawały do tej pory, **labeling** był na końcu głównego predykatu rozwiązującego. Zastanów się, jak wpłynęłyby na poprawność i efektywność procedury wyszukiwania rozwiązania inne pozycje tego predykatu.

Zadanie 7. (1p) Heurystyczne algorytmy można łączyć ze sobą w dość dowolne kombinacje. Zaproponuj sensowne² połączenie:

- a) algorytmów ewolucyjnych i hill climbing,
- b) A^* oraz local beam search,
- c) symulowanego wyżarzania i algorytmów ewolucyjnych (inaczej niż w podpunkcie a, z symulowanego wyżarzania powinniśmy wziąć jedynie ogólną ideę),
- d) taboo search z algorytmami ewolucyjnymi,
- e) hill climbing z BFS.

W razie potrzeby krótko wyjaśnij, jak działają oryginalne algorytmy.

Zadanie 8. (1p) Hiperkrzyżówka składa się z K słów o długości K , które tworzą krzyżówkę zawierającą $2K$ wyrazów K -literowych (jak na przykładzie poniżej).

```
semen
klasa
łazik
onuca
naraz
```

Jak przedstawić hiperkrzyżówkę jako problem więzowy? Podaj dwie reprezentacje i zaproponuj algorytm generujący hiperkrzyżówki. Postaraj się, by w maksymalnym stopniu wykorzystywał on specyfikę zadania.

Zadanie 9. (1p) Rozważamy jakąś grę turową. Załóżmy, że znamy algorytm, którym posługuje się nasz oponent (i jest to algorytm deterministyczny). Wyjaśnij, dlaczego w tym momencie można potraktować zadanie znalezienia ruchu jako zadanie przeszukiwania?

- a) Które algorytmy rozwiązywania zadań przeszukiwania można by zastosować do gier o złożoności szachów czy warcabów?
- b) Zaproponuj jakiś sposób użycia takiego algorytmu w prawdziwej grze.

Zadanie 10. (1p) Wybierz co najmniej jedną z następujących gier: lis i gęsi, breakthrough (board game), pentago, skoczki (gra planszowa). Zaproponuj dla wybranej gry heurystyczną funkcję oceniającą sytuację na planszy. Wyraźnie wskaż, jakie ta funkcja ma parametry.

Uwaga: zapytanie: `<gra-X> wikipedia` zawsze prowadzi do odpowiedniego opisu.

²Trudno zdefiniować precyzyjnie sensowność. Na potrzeby tego zadania przyjmujemy, że sensowne oznacza przekonanie autora, że warto takie połączenie sprawdzić w pewnych zadaniach – i że może ono zadziałać lepiej niż każdy ze składników kombinacji