

Wybrane zagadnienia sztucznej inteligencji laboratorium

Ćwiczenie 1 MCTS, Gry niedeterministyczne

opracowanie: J. Jakubik

Opis ćwiczenia

W ramach ćwiczenia zapoznawać się będziemy z Monte Carlo Tree Search – probabilistyczną metodą przeszukiwania drzewa gry, będącą podstawą obecnych podejść state of the art w zakresie sztucznej inteligencji grach. MCTS stało się de facto standardem w złożonych grach planszowych (np. Go) jak i grach zawierających elementy niedeterministyczne (np. Poker). Głośne w ostatnich latach AlphaGo, pierwsza w historii sztuczna inteligencja zdolna grać w Go na poziomie przewyższającym profesjonalnych ludzkich graczy, oparte było o wariant MCTS rozbudowany o wykorzystanie splotowych sieci neuronowych.

Proponowaną grą do zaimplementowania w zadaniu jest Hearthstone – popularna gra karciana na urządzenia mobilne. Student może zaproponować inną grę o porównywalnym stopniu złożoności i losowości.

Zasady gry

Karcianka Hearthstone stawia naprzeciw siebie graczy reprezentowanych przez Bohaterów, korzystających z kart Stronników i Zaklęć aby pokonać przeciwnika. Gracze naprzemiennie wykonują swoje ruchy, atakując Bohatera przeciwnika i kontrolując Stronników na planszy gry. Dostępne ruchy, a co za tym idzie rozgałęzienia drzewa gry, ogranicza liczba kart w ręce gracza, liczba stworów na planszy oraz dostępny surowiec: Mana.



Mana

Kryształy Many ograniczają możliwe zagrywanie kart z ręki. W swojej n-tej turze gracz ma do dyspozycji n kryształów i jest w stanie zagrać dowolną kombinację kart o **łącznym** koszcie nie większym niż n. Maksymalna liczba kryształów many to 10 – od jedenastej tury wzwyż liczba kryształów nie rośnie.

Karty

Dla uproszczenia zakładamy dwie identyczne talie (oczywiście inaczej potasowane) po 20 kart. **Gracz pierwszy rozpoczyna grę z trzema kartami na ręce, gracz drugi z czterema. Na początku każdej tury gracz dociąga jedną kartę.** Wymagana w zadaniu jest implementacja dwóch typów kart:

Stronnik – karta która po zagraniu pozostaje na planszy. Każdy gracz zagrywa stronników po swojej stronie planszy, może mieć do 7 stronników jednocześnie. (Dopuszczalne jest zmniejszenie tej liczby, jeśli będzie to konieczne ze względów wydajnościowych – należy jednak konsultować to z prowadzącym). Stronnik definiowany przez trzy wartości, **Koszt**, **Atak** i **Życie**. Jeżeli Życie stronnika spadnie do 0 w wyniku pewnych ruchów gracza, jest zdejmowany z planszy.



Zaklęcie – karta która po zagraniu wywołuje pewien natychmiastowy efekt, np. niszczy stronnika. Ma przypisany Koszt i działanie.



Ruchy

W zadaniu zakładamy uproszczoną wersję gry, gdzie możliwe są następujące ruchy:

- Zagranie karty – jeżeli jest to stronnik, **nie będzie on mógł atakować w tej samej turze w której został zagrany.**
- Atak stronnika gracza wykonującego ruch na stronnika przeciwnika – w wyniku tej akcji odejmujemy Atak pierwszego stronnika od Życia drugiego, i vice versa. Jeżeli życie stronnika spadło poniżej 0, zdejmujemy go z planszy. **Każdy stronnik może zaatakować tylko raz na turę.**
- Atak stronnika gracza wykonującego ruch na bohatera przeciwnika – w wyniku tej akcji odejmujemy Atak stronnika od Życia bohatera. Jeżeli Życie bohatera spadło do 0 lub niżej, gra kończy się przegraną zaatakowanego gracza.

W obrębie jednej tury, gracz może wykonać dowolną kombinację ruchów możliwą przy posiadanych kartach w ręce, kryształach many i stworach na planszy.

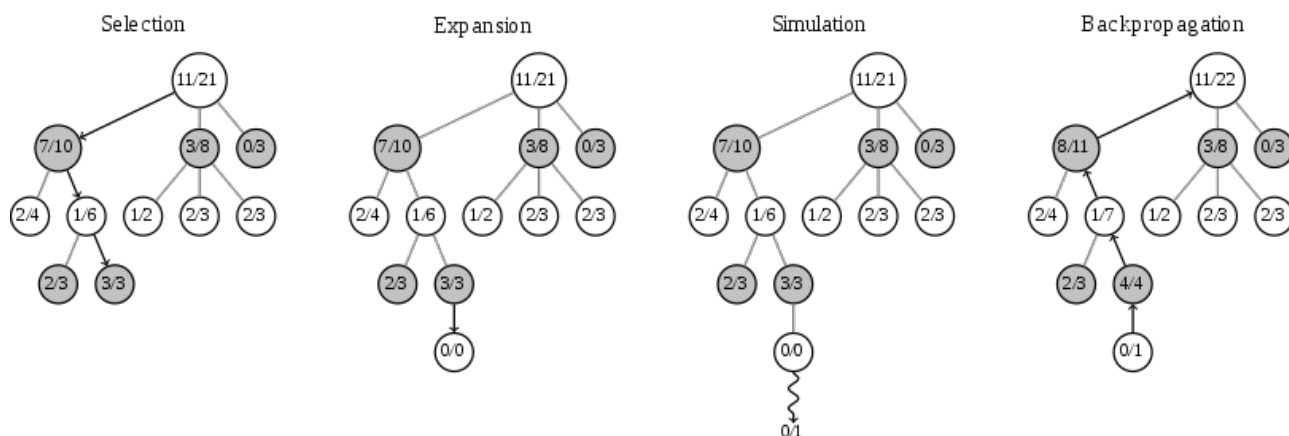
Dodatkowe założenia upraszczające implementację

- Identyeczni bohaterowie, po 20 punktów Życia.
- Dwie identyczne talie po 20 kart.
- Gramy w otwarte karty (ręka przeciwnika jest znana)
- Skład talii: po 2 egzemplarze 10 kart: 7 stronników, 3 zaklęcia.
- 2 stronników powinno mieć dodatkową specjalną umiejętność, 5 może być prostymi kartami posiadającymi wyłącznie Koszt, Atak i Życie

Karty można znaleźć na stronie: <https://www.hearthpwn.com/cards>

Przykład umiejętności ciekawie wpływającej na rozgałęzianie drzewa gry: Prowokacja. Umiejętność działa przez cały czas kiedy stronnik jest na planszy. Stronnik z Prowokacją sprawia, że nie można atakować innych Stronników ani Bohatera po jego stronie, o ile oni sami nie mają Prowokacji. Korzyść z tej umiejętności polega więc na ograniczaniu przestrzeni dostępnych ruchów przeciwnika.

Monte Carlo Tree Search



https://en.wikipedia.org/wiki/Monte_Carlo_tree_search

Algorytm Monte Carlo Tree Search przeszukuje drzewo stanów gry w sposób częściowo losowy, w ogólnym schemacie czterech kroków:

1. Selekcja: wybieramy węzeł drzewa, który będziemy rozbudowywać. Wybór musi równoważyć **eksplorację** nieprzeszukanych dotychczas gałęzi drzewa i **eksploatację** gałęzi dających obiecujące wyniki
2. Ekspansja: rozbudowujemy drzewo o nowe stany gry, dodając do liścia możliwe ruchy gracza wynikające z obecnego stanu gry
3. Symulacja: z liścia **rozgrywamy** grę do końca (ang. **playout**) w sposób losowy
4. Propagacja wsteczna: W zależności od wyniku, aktualizujemy wartości w węzłach na ścieżce prowadzącej do obecnego ruchu. Wartości reprezentują liczbę wygranych/liczbę rozegranych playoutów

Węzły niedeterministyczne

Aby umożliwić rozgrywanie niedeterministycznych gier, musimy rozwinąć powyższy algorytm o specjalny rodzaj węzła w drzewie gry. Reprezentuje on zmianę stanu gry nie w wyniku ruchu gracza, ale przez losowe elementy gry.

Zadania

Należy zaimplementować logikę gry i gracza MCTS, w sposób umożliwiający przeprowadzanie gier zarówno ruch po ruchu (aby pokazać wykonywanie poprawnych ruchów przez graczy) jak i bez wyświetlania stanu gry, dla przyspieszenia badań.

GUI nie jest wymagane, stan gry powinien być jednak pokazywany w taki sposób, aby widoczne były wszystkie istotne informacje.

Ćwiczenie podsumowane będzie sprawozdaniem z badań wydajności i efektywności MCTS w grze. W sprawozdaniu nie jest wymagany wstęp teoretyczny, należy jednak dokładnie opisać, co się zaimplementowało np. jakie karty, jak dokładnie wygląda funkcja celu gracza zachłannego itd.

W badaniu wydajności należy zaprezentować zależność od czasu wartości takich, jak

- liczba wykonanych playoutów
- średnia, maksimum i mediana głębokości liścia
- średni procent eksplorowanych dzieci danego węzła
- etc.

W badaniu umiejętności gry, będzie konieczne zaimplementowanie naiwnych **zachłannych graczy heurystycznych**. Tzn. algorytmów rozgrywających turę tak, aby zmaksymalizować obecną wartość pewnej funkcji oceny stanu gry. Należy zaimplementować następujące modele graczy:

- losowy
- agresywny – w pierwszej kolejności maksymalizuje obrażenia zadane bohaterowi przeciwnika, następnie kontroluje stan planszy
- kontrolujący – w pierwszej kolejności kontroluje stan planszy, atakuje bohatera tylko jeśli nie ma żadnych stronników przeciwnika

Ze względu na wysoki stopień losowości, należy symulować przynajmniej 100 rozgrywek gracza MCTS z graczem zachłannym.

2 z 10 punktów możliwych do zdobycia za ćwiczenie to **punkty za kreatywność** – aby je zdobyć, należy rozbudować ćwiczenie o dodatkowe, nieuwzględnione powyżej komplikacje. Może to być na przykład heurystyczna funkcja „prowadząca” playouty zamiast playoutów zupełnie losowych, implementacja kart o wysokim stopniu złożoności i zbadanie ich wpływu na wyniki itp. Otrzymanie punktów jest możliwe tylko, jeśli dodatkowe funkcje zostaną **przebadane i pokazane w sprawozdaniu**. Nie można ich otrzymać za samą implementację.

Harmonogram oczekiwanych postępów

Tydzień 1: Wprowadzenie do ćwiczenia

Tydzień 2: Implementacja logiki gry

Tydzień 3: Implementacja agentów zachłannych, początek implementacji MCTS

Tydzień 4: Gotowa implementacja MCTS w stopniu umożliwiającym badanie

Tydzień 5: Oddanie sprawozdania z wynikami badań

Za brak systematyczności w tygodniach 2–4 można stracić po punkcie za każdy.

Ocena ćwiczenia

2pkt	Implementacja logiki gry
2pkt	Implementacja algorytmu MCTS
2pkt	Badanie wydajności czasowej algorytmu MCTS
2pkt	Badanie wyników MCTS w starciu z heurystycznymi agentami zachłannymi
2pkt	Punkty za kreatywność