Algorytmy i modele inspirowane

biologicznie

- **Ⅲ** Oceny
- Algorytmy i modele inspirowane biologicznie
- ☐ Podstawowe informacje
- ☐ Wykłady
- Laboratoria
- **Strona** główna
- Kokpit
- **H** Kalendarz
- Prywatne pliki
- Moje kursy
- Staże Al Tech
- Algorytmy i modele inspirowane biologicznie
- Master Thesis Proposals

Algorytmy i modele inspirowane biologicznie

Strona główna / Moje kursy / WliT / Informatyka / Stacjonarne / II stopień / Sztuczna Inteligencja / Semestr 2 [WliT-Inf-st-II-si] / Algorytmy i modele inspirowane biologicznie / Laboratoria / Lab. 6. Transformacja między przestrzeniami: z przestrzeni GP do przestrzeni projektowania ewolucyjnego



czas **Ocena** Jeszcze nie ocenione

Na wykładzie w Części 3 omówiliśmy programowanie genetyczne (GP, przejrzyj teraz pdf wykładowy). DEAP potrafi optymalizować GP (przykłady kodu są też w pdf wykładowym). Stwórz mapowanie przestrzeni GP do jednej, wybranej reprezentacji genetycznej we Framsticks. Po stronie GP potrzebne będą własne węzły funkcyjne i terminale oraz funkcja oceny drzewa, która musi wyprodukować z niego genotyp w docelowej reprezentacji Framsticks.

Najpierw pytanie (odpowiedz Tak lub Nie): czy optymalizacja (np. algorytmem ewolucyjnym) w tworzonej w tym zadaniu przestrzeni GP (transformującej każdy genotyp GP do jakiegoś genotypu w natywnej reprezentacji Framsticks) jest w stanie uzyskiwać lepsze wyniki, niż analogiczna optymalizacja z natywną reprezentacją i jej operatorami we Framsticks? Pamiętaj, że reprezentacja GP przy takim podejściu nie jest w stanie wytworzyć niczego, czego nie da się wytworzyć w natywnej reprezentacji (do której GP jest transformowane). Uzasadnij swoją odpowiedź. To pytanie można uogólnić do dwóch reprezentacji A i B, jednej tłumaczonej do drugiej, zupełnie abstrahując od Framsticks i GP – i odpowiedź na takie uogólnione pytanie oraz jej uzasadnienie będą takie same.

Optymalizacja algorytmem ewolucyjnym w przestrzeni GP może dawać lepsze wyniki (w pewnym zadanym czasie), ponieważ stosuje inne operatory, które mają potencjał prowadzić optymalizację skuteczniej (przykładowo są odpowiednio silne lub rzadziej prowadzą co niepoprawnego genotypu).

Pytanie 1 Zakończone Punkty maks.: 1,00

Pytanie **2**

Zakończone

Punkty maks.: 9,00

♥ Oflaguj pytanie

♥ Oflaguj pytanie

Nawigacja w teście



Zakończ przegląd

Po stronie GP zaimplementuj własne węzły funkcyjne i terminale oraz funkcję oceny drzewa, która generuje z drzewa genotyp w docelowej reprezentacji Framsticks (ten genotyp jest potem oceniany dokładnie tak, jak na wcześniejszych laboratoriach). Wybierz jeden z problemów optymalizacyjnych, jakie do tej pory rozważaliśmy: "vertpos" bez sieci neuronowej (**f0**, **f1** albo **f4**) albo "velocity" z siecią neuronową (**f0** albo **f1**). Użyj takiej samej sekwencji plików *.sim, jaką używaliśmy przy wybranym teraz przez Ciebie problemie.

Porównaj skuteczność ewolucji przeszukującej przestrzeń genotypów za pomocą natywnych operatorów we Framsticks oraz tej przeszukującej przestrzeń GP za pomocą gotowych operatorów w DEAP. Możesz przetestować kilka swoich pomysłów i różnych wariantów transformacji jednej przestrzeni do drugiej. Opisz dokładnie zrealizowane pomysły (możesz ew. zamieścić kluczowe, krótkie fragmenty kodu) i uzasadnij swój sposób rozwiązania problemu. Wygeneruj te same wykresy porównawcze, co zwykle, zinterpretuj je i opisz wnioski.

Problem optymalizacyjny: vertpos w f1.

Do zbioru elementów GP dodaję funkcje:

- 1. dodającą stick'a na froncie fragmentu genotypu
- 2. tworzącą rodzica węzła, czyli dodającą '()' wokół fragmentu genotypu
- 3. tworząca rozgałęzienie, czyli dodająca "," pomiędzy dwa fragmenty
- 4. serię funkcji dodających modyfikator (np. R, r, C...) przed fragment genotypu

Takie funkcje wyczerpują możliwości tworzenia genotypu w reprezentacji f1.

Osobniki w natywnej reprezentacji ponownie zmagają się z problemem kiepskiej jakości początkowej, co powoduje, że w pierwszych iteracjach optymalizacja w GP jest skuteczniejsza. Finalnie wyniki w wersji natywnej są wyższe. Obserwujemy zatem standardowe zróżnicowanie algorytmów w różnych horyzontach czasowych. Potencjalnym powodem takiego zjawiska w tym eksperymencie może być różny, nie porównywalny limit na złożonościach genotypów - GP ma drzewa w zakresie wysokości 1-10, a oryginalne f1 ma ograniczenia przy wywołaniu (-max_numparts 15, -max_numjoints 30, -max_numneurons 20, max_numconnections 30). Dodatkowo drzewa powstają od razu całkiem złożone (wypełnione) co może być przyczyną skutecznych osobników już na początku.

Czas działania natywnie jest krótszy (przy stałej liczbie iteracji) niż w GP najpewniej z powodu potrzeby dodatkowej translacji genotypu między reprezentacjami.

wykresy_razem.png

Zakończ przegląd