

Zadanie domowe nr 1

Algorytm genetyczny

Uwaga: zakres zadania domowego obejmuje Lab.2 (Algorytm genetyczny) i Lab.3 (Badanie algorytmu genetycznego).

Wybierz problem:

- a) 3-SAT
- b) Nonogram
- c) Nurikabe
- d) Labirynt*

z zadania 2 z laboratoriów 2 (lub inny, za zgodą prowadzącego zajęcia) i rozwiąż go za pomocą algorytmu genetycznego.

* Problem przechodzenia przez labirynt jest nieco ciężki dla algorytmów genetycznych dlatego nie zachęcam do jego wybierania, ale jak ktoś bardzo chce się z nim zmierzyć to powinienem poeksperymentować z różnymi funkcjami fitness. Można także porównać algorytm genetyczny z innymi algorytmami szukającymi drogi w labiryncie np. A*.

Problemy można rozwiązać z użyciem R-Studio i paczki `genalg` tak, jak zrobiono to w zadaniu 1 lab.2 dla problemu plecakowego. Głównym wyzwaniem będzie napisanie funkcji fitness, która we wszystkich przypadkach wymaga trochę umiejętności programistycznych. Można też jednak użyć innych języków programowania i paczek (np. Python, C#, Java, JS), jeśli ktoś ma inne upodobania. Ważne jest jednak, żeby mieć wkład w rozwiązanie (przede wszystkich chodzi o napisanie funkcji fitness), rozwiązania ściągnięte z Internetu są niedozwolone.

Wymagania szczegółowe co do rozwiązania:

- Rozwiązanie zadania powinno składać się z:
 - pliku(ów) zawierającego kod programu (np. wszystkie funkcje z R, łącznie z funkcją fitness),
 - ewentualnie dołączonych plików z zakodowanymi problemami (formuły CNF lub łamigłówki),
 - sprawozdania w formie pliku PDF, w którym zamieścimy opis wykonania zadania domowego wraz z wykresami.
- W sprawozdaniu PDF opisz chromosomy i funkcję fitness. Wskaż na najważniejsze fragmenty kodu i objaśnij je swoimi słowami. Przedstaw działanie funkcji fitness na kilku chromosomach. Mile widziane jest przetestowanie kilku funkcji fitness i wybranie najbardziej efektywnej.
- Wybierz instancję problemu (nie za dużą i nie za małą), korzystając z zasobów internetowych. Dla jakich parametrów (szansa mutacji, wielkość populacji, elitizm, rodzaj krzyżowania, rodzaj selekcji, funkcja fitness) algorytm działa najszybciej i najefektywniej? Dokonaj eksperymentów. Opisz je słownie i/lub za pomocą wykresów w sprawozdaniu PDF.

- Wybierz wiele instancji problemów o różnej złożoności korzystając z zasobów internetowych (baz danych, automatycznych generatorów łamigłówek itp.), tzn.
 - Dla formuł 3-SAT będą to formuły o różnej długości (liczbie klauzul) i liczbie zmiennych. Można rozważyć formuły o długości 10,20,40,60,80,100 klauzul.

Pomocne strony:

<https://www.cs.ubc.ca/~hoos/SATLIB/benchm.html>

<https://people.sc.fsu.edu/~jburkardt/data/cnf/cnf.html>

- Dla Nonogramów i Nurikabe obrazki o różnych wielkościach np. 10x10, 15x15, 20x20, 30x30, 40x40, 50x50. Pomocne strony:

<https://github.com/mikix/nonogram-db>

<http://liacs.leidenuniv.nl/~kosterswa/nono/sjoerd/indexeng.html>

<https://puzzlemadness.co.uk/archive/nurikabe/>

<https://www.gmpuzzles.com/blog/category/shading/nurikabe/>

A następnie porównaj czasy działania algorytmu genetycznego dla każdej z tych instancji. Jak szybko wzrasta czas działania algorytmu względem rozmiaru problemu? Liniowo? Zestaw dane na wykresie liniowym przyjmując na osi X wielkość problemu, a na Y czas działania algorytmu. Zinterpretuj wynik.

- Jako trochę alternatywny projekt, można porównać szybkość i skuteczność działania algorytmu genetycznego z innym algorytmem rozwiązującym problem. Przykłady: Genetyczny vs A* dla Labiryntu, Genetyczny vs DPLL dla 3SAT.

Rozwiązanie należy pokazać najpóźniej na piątych zajęciach w semestrze (lab.5) i krótko o nim opowiedzieć, następnie odpowiedzieć na parę pytań prowadzącego zajęcia.

Rozwiązanie (spakowane), należy też umieścić w Internecie np. na serwisach Dropbox lub ZipShare, itp. i przesłać email do prowadzącego zajęcia o tytule „PD-1-genalg” i treści w formacie:

PD-1-genalg

Jan Kowalski, nr indeksu: 123456

<http://tutaj-link-do-rozwiazanej-pracy-domowej.com>