ДКР на тему

«Генетичний алгоритм розв’язання задачі прорюкзак»

Виконав студент групи ІС-72

Гороховський Іван

ЭТАП І (теоретичний)

Для вирішення задачі комбінаторної оптимізації про рюкзак будемо використовувати генетичний алгоритм. Відповідь будемо представляти у вигляді бінарного вектора X, в якому на i-му місті стоїть 1 якщо ми беремо і-ту реч і 0 якщо не беремо.

Обмеженням данній задачі є максимальна вага яку ми можемо покласти у рюкзак, тобто якщо W - вектор ваги предметів, а w - максимальна допустима вага, то:

X \* ≤w

Цільовою функцією є максимізація вартості предметів рюкзака, тобто якщо P - вектор вартостей предметів, то

Z = X \*

1. Сформувати популяцію без повторів

Для формування початкової популяції (матриці G) будемо використовувати наступний алгоритм:

Випадковим чином заповнемо вектор X, далі перевіримо якщо вектор з такими самими значеннями вже присутній, то будемо генерувати новий.

1. “Тіло” алгоритму

2.1 Выбір батьків

Для кожного наступного нащадка будемо використовувати аутбридінг.

Аутбридинг — первый родитель выбирается случайно, а вторым выбирается такой, который наименее похож на первого родителя.

Також зауважемо, що для генерації нащадків ми не будемо повторно використовувати одних і тих самих батьків.

2.2 Створення нащадків

Для створення нащадків будемо використовувати наступний алгоритм:

Позначимо X1 - першого батька, X2 - другога батька, X - вектор нащадка, m1 - значення метрики для першого батька, m2 - значення метрики для другого батька:

Якщо X1i = X2i, тоді Xi := X1i, інакше будемо використовувати випадковий підхід для того щоб не “застряти” в локальному екстремумі.

P(Xi := X1i)=

P(Xi := X2i)=

Таким чином ми будемо більш схилятися до значення кращого з батьків, але залишимо елемент стохастики.

2.3 Перевірка допустимості нащадка і його реанімація

2.3.1 Перевірка допустимості нащадка

В нашій задачі ми маємо лише одне обмеження:

X \* ≤w

Якщо воно виконується - нащадок є допустимим. Інакше переходимо до пунтку 2.3.2.

2.3.2 Реанімація нащадка

Для недопутимого нащадка X, візьмемо всі об’єкти де xi = 1 і відсортуємо їх за вартістю. Поки не буде виконуватись умова допустимості нащадка X, будемо викидати найменш цінну реч.

2.4 Мутація нащадка

Якщо L - довжина вектора X, то з ймовірністю в і-ій клітинці може замінитись значення на протилежне.

2.5 Локальне покращення

Застосуємо локальне покращення - будемо додавати до рюкзака предмети, що залишилися, починаючи з найменшої маси, до тих пір, поки не буде порушена умова допустимості.

2.6 Оновлення популяції

З поточної популяції залишаємо N найкращих за значенням ЦФ нащадків.

ЭТАП IІ (практичний)

3.1 Задати умову персональної задачі

* Кількість предметів (L) = 12
* Вектор ваг W:

[8, 5, 4, 4, 15, 18, 6, 10, 17, 12, 9, 16]

* Вектор вартості P:

[10, 7, 23, 8, 14, 28, 11, 15, 14, 15, 20, 23]

* Максимальна вага рюкзака (w) = 50
* Кількість осіб популяції (N) = 15
* Ймовірність мутації, згідно формули вище (p) = 2.1%

3.2 Рішення персональної задачі

Для розв’язку персональної задачі, мною була написана програма на мові Python.