

Оптимізація графіку відключень електроенергії

Підготували студенти групи КМ-13

Оленич Х. М. та Резник Д. О.

Ролі кожного розробника

Резник Д. О. - розробка програмної реалізації генетичного алгоритму

Оленич Х. М. - ідея проекту, адаптація генетичного алгоритму до теми задачі, покращення модульності та ефективності програми

Формалізація постановки задачі

- Мета роботи: створення щотижневого графіку рівномірних відключень електроенергії на основі даних про її споживання окремими одиницями(вузлами), необхідність постачання, обсяги можливого постачання тощо.
- Об'єкт дослідження: оптимізація електроенергії та підходи до неї
- Предмет дослідження: генетичний алгоритм для пошуку оптимізованого графіку
- Кінцевий результат: бінарний графік станів роботи об'єктів в електромережі для кожного інтервалу дня протягом тижня.

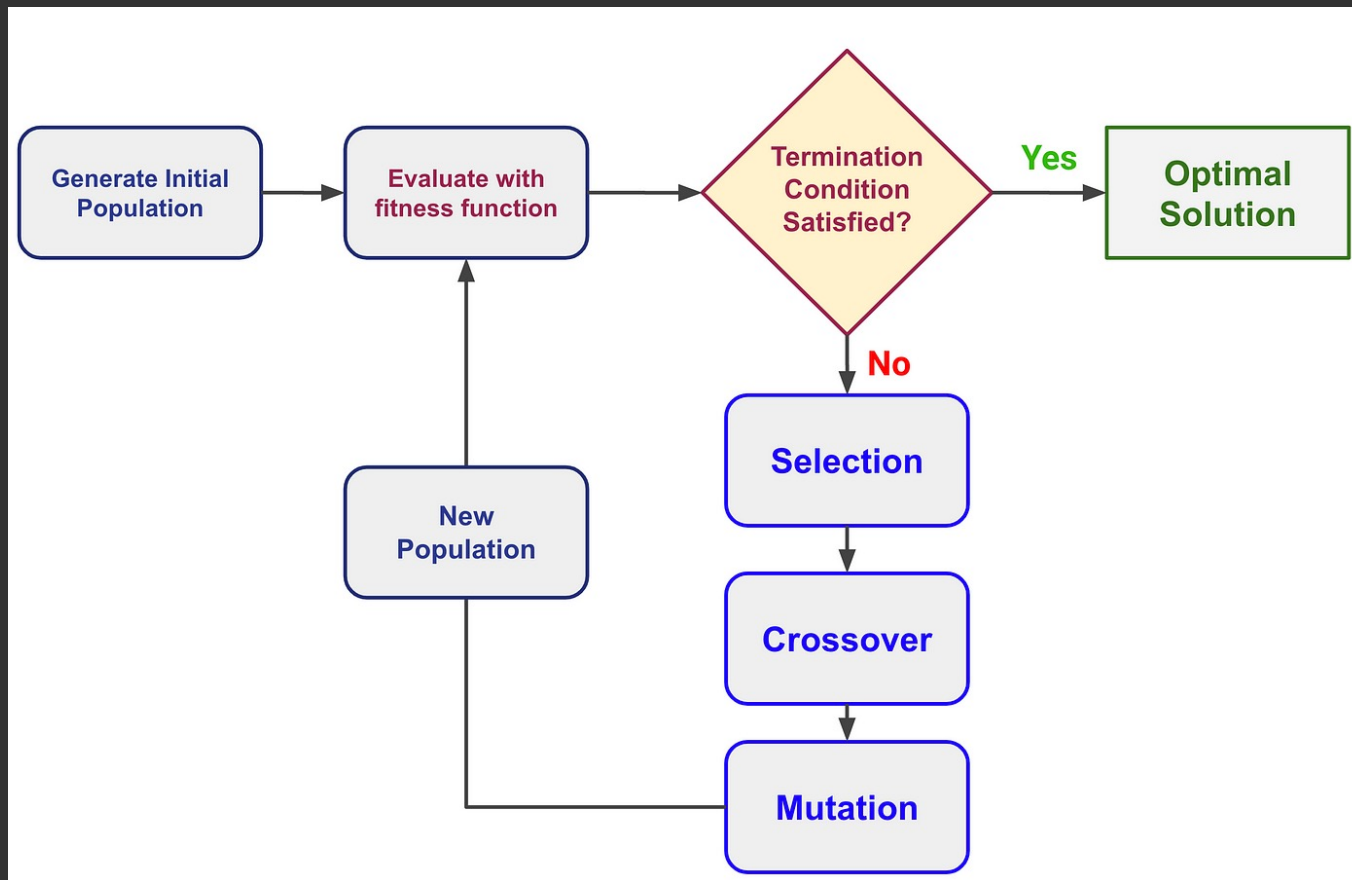
Вступ до задачі

- ▶ Можна використати наступне спрощення: розбиваємо часовий проміжок у 24 години на підпроміжки у 4 години - у сумі 6 інтервалів.
- ▶ Враховуємо пріоритетність за допомогою додаткового списку. Пріоритетність об'єктів для житлових будинків встановлюється однаковою, хоча і може змінюватись. У випадку стратегічних об'єктів - пріоритетність найвища.
- ▶ Задача є подібною до задачі про рюкзак: треба підключити у певних межах споживання оптимальним способом найважливіші об'єкти. Але важливо враховувати часову складність алгоритму, яка є високою для стандартних алгоритмів(динамічне програмування та інші).

Використаний метод

- ▶ Переваги: **Комбінаторна оптимізація:** генетичні алгоритми справляються з розв'язанням задач комбінаторної оптимізації, коли простір рішень великий і складний. Вони особливо корисні, коли важко знайти точне рішення.
- ▶ **Евристичний характер:** Генетичні алгоритми є евристичними за своєю природою, тобто відшукує прийнятне рішення, але необов'язково оптимальне, що робить їх добре придатними для великих, складних проблем, де пошук оптимального рішення обчислювально дорогий або непрактичний.
- ▶ **Адаптивність:** Генетичні алгоритми можуть адаптуватися до різних структур задачі та менш чутливі до змін у формулюванні проблеми. Ця адаптивність робить їх придатними для сценаріїв реального світу зі змінними обмеженнями.

Генетичний алгоритм



Математична модель

- ▶ Формалізуємо задачі з математичної точки зору:
- ▶ Генотип (G) - певний розв'язок, який представляється у вигляді бінарних векторів $\{0, 1\}$ для 6 інтервалів, 7 днів та 7 об'єктів.
- ▶ Популяція (P) - множина всіх потенційних розв'язків.
- ▶ Функція fitness (f) - функція, яка приймає 3d матрицю і видає значення оптимальності розв'язку.

Математична модель

Кроки генетичного алгоритму:

Ініціалізація - випадкове генерування множини $F_0 = \{G_1, G_2, \dots, G_N\}$

Оцінка оптимальності рішення - $Fitness(G_i) = f(G_i) = \sqrt{\frac{\sum_{i=1}^n w_i (x_i - \mu)^2}{n * m * v}}$ - взважене середнє квадратичне відхилення суми споживання об'єктів.

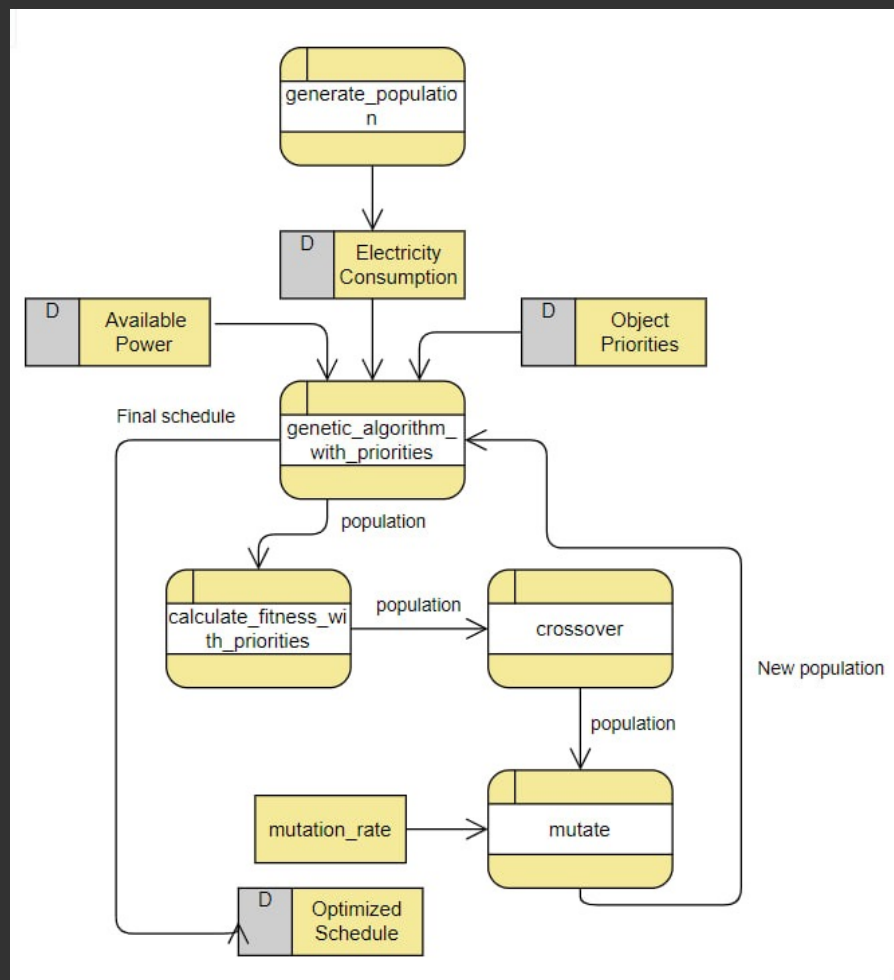
Обирання кращої половини - $G_i = \text{sort}(G_i)[:\text{length}(G_i)/2]$

Кросовер - $c = C(G_i)$

Мутація - $m = M(c)$

Створення нової популяції - $G_{i+1} = m$

DFD-діаграма програми



Використані інструменти

- ▶ Обрана мова програмування - Python
- ▶ Додаткові використані модулі:
- ▶ Matplotlib - візуалізація результатів
- ▶ Numpy - математичні розрахунки та зберігання вхідної та вихідної інформації

Результат роботи програми

- ▶ Результат представлений у наступному вигляді (урізана кількість):

Optimized Schedule with Priorities:

```
[[[0 1 1 1 0 1]
  [1 1 1 0 0 0]
  [1 1 0 0 0 1]
  [0 0 1 1 0 1]
  [1 1 1 0 1 0]
  [1 1 0 1 0 1]
  [0 1 1 1 1 0]]]
```

```
[[[1 0 1 1 1 1]
  [0 1 1 0 1 0]
  [0 0 0 1 1 1]
  [0 1 0 1 0 1]
  [1 1 1 1 0 0]
  [0 1 1 1 0 1]
  [1 0 0 1 1 1]]]
```

```
[[[0 1 0 0 1 0]
  [0 0 1 0 0 1]
  [0 0 0 1 1 1]
  [0 0 0 0 1 1]
  [0 1 0 0 0 0]
  [1 1 0 0 0 1]
  [0 1 1 0 0 1]]]
```

Day 3 Summary:

Object 1 Consumption:

Interval 1: ON, Consumption: 0.83
Interval 2: ON, Consumption: 0.21
Interval 3: OFF, Consumption: 0.18
Interval 4: OFF, Consumption: 0.18
Interval 5: OFF, Consumption: 0.30
Interval 6: ON, Consumption: 0.52

Object 2 Consumption:

Interval 1: OFF, Consumption: 0.60
Interval 2: OFF, Consumption: 0.92
Interval 3: OFF, Consumption: 0.09
Interval 4: ON, Consumption: 0.20
Interval 5: ON, Consumption: 0.05
Interval 6: ON, Consumption: 0.33

Object 3 Consumption:

Interval 1: OFF, Consumption: 0.52
Interval 2: OFF, Consumption: 0.43
Interval 3: OFF, Consumption: 0.03
Interval 4: ON, Consumption: 0.11
Interval 5: ON, Consumption: 0.03
Interval 6: ON, Consumption: 0.64

Total Electricity Consumption on Day 3: 9.44 kWh

Результат роботи програми

- ▶ Ефективність моделі представлена у наступному вигляді:

