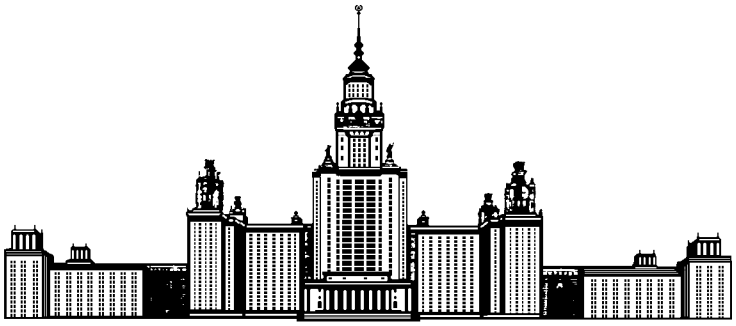
Филиал Московского государственного университета
имени М. В. Ломоносова
в городе Севастополе



студент группы ПМ-401
Машарипов Тимур Алишерович

Эксперименты с основными параметрами генетических алгоритмов в задаче OneMax

Практическая работа по курсу
Генетические алгоритмы

Севастополь, 2021

Содержание

1. Введение

OneMax - это задача оптимизации. Служит аналогом Hello World в мире генетических алгоритмов.

Цель состоит в том, чтобы найти двоичную строку заданной длины, для которой сумма составляющих её цифр максимальна.

Эту задачу возможно решить применяя генетические алгоритмы. Далее мы проведём ряд экспериментов, меняя различные параметры алгоритма.

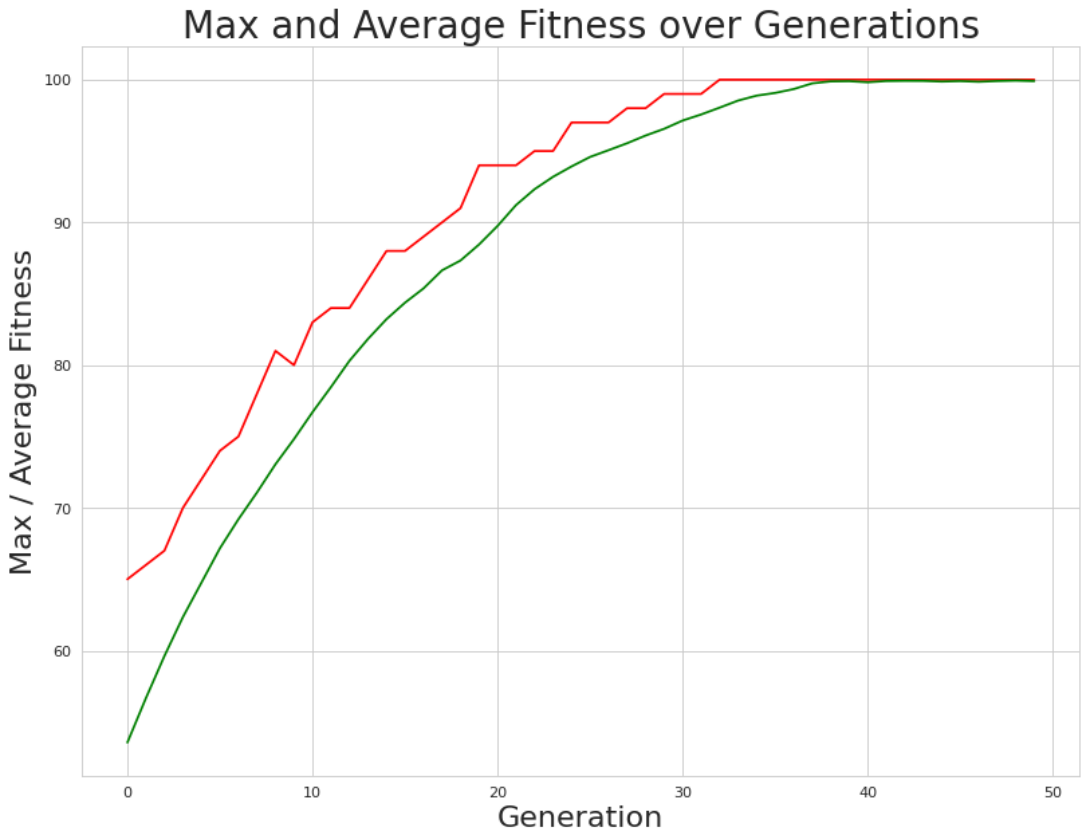
2. Эксперименты

Так выглядят начальные параметры алгоритма:

```
run_experiment(
    population_size=200,
    max_generations=50,
    crossover_operator=tools.cxOnePoint,
    mutation_probability=0.1,
    indpb_multiplier=1.0,
    tournsize=3,
)
```

Следующий график демонстрирует различные результаты для каждого поколения:

Рис. 1: Результат работы алгоритма при начальных параметрах

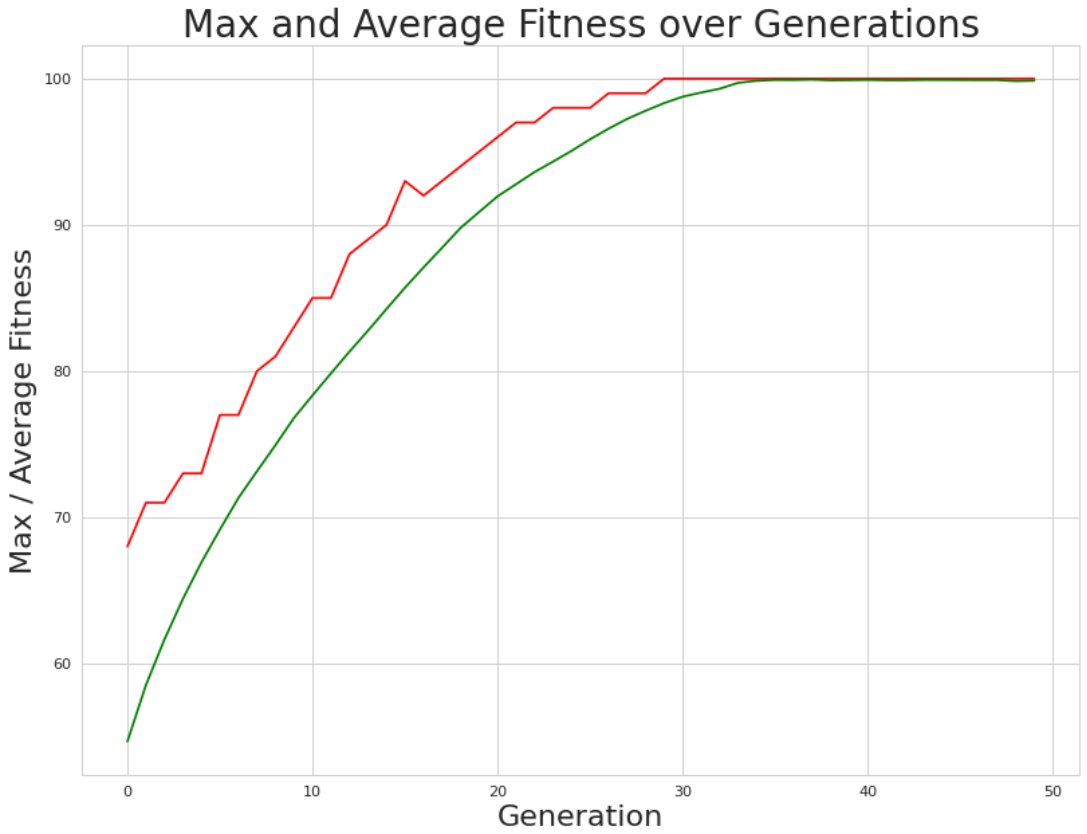


2.1. Размер популяции

Начнём с изменения параметра *размер популяции*, установив его в значение 500. При этом заметим, что алгоритм стал быстрее сходится.

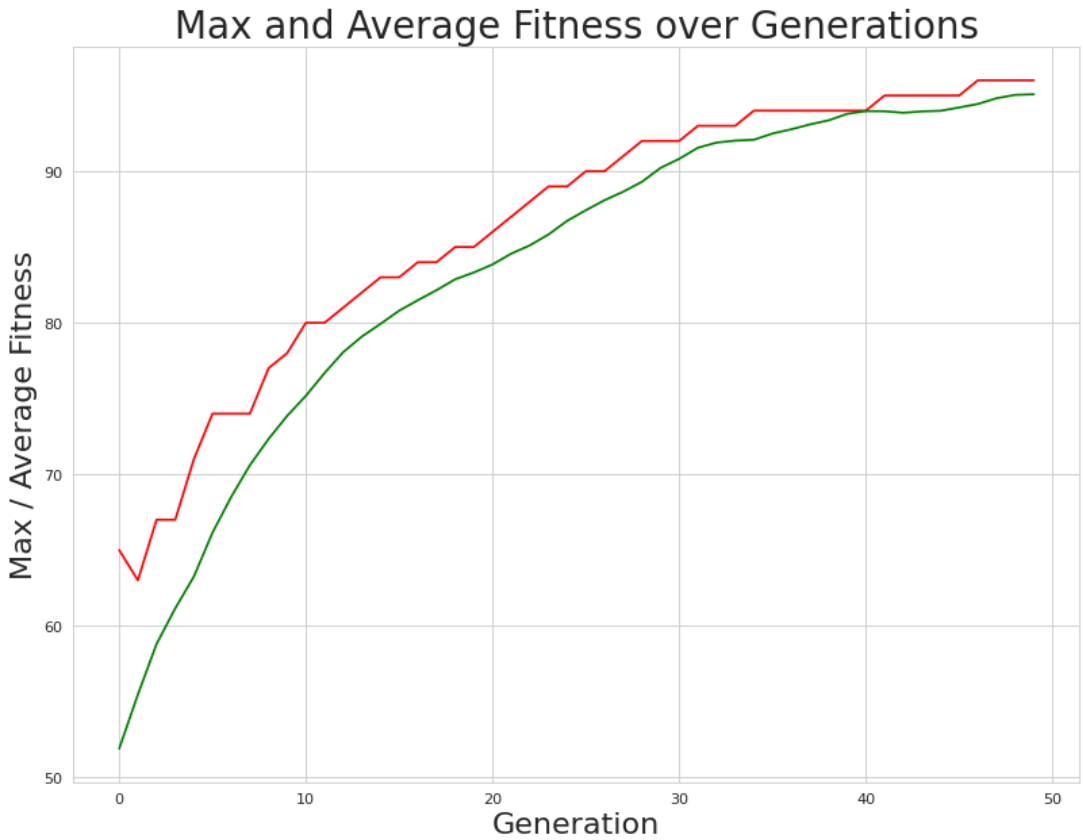
```
run_experiment(
    population_size=500,
    max_generations=50,
    crossover_operator=tools.cxOnePoint,
    mutation_probability=0.1,
    indpb_multiplier=1.0,
    tournsize=3,
)
```

Рис. 2: Размер популяции = 500



При значении размера популяции 100, сходимость замедляется. Более того, лучшее решение так и не нашлось.

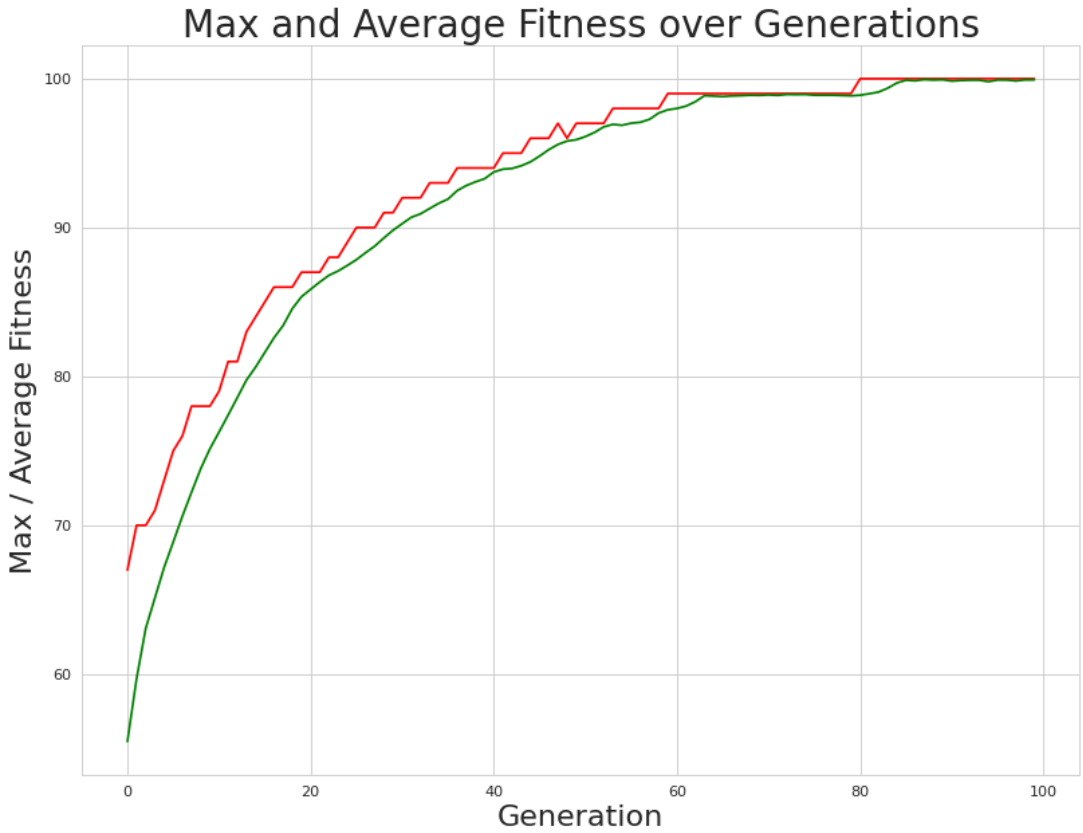
Рис. 3: Размер популяции = 100



2.2. Количество поколений

Чтобы исправить это, изменим максимальное количество поколений до 100. Как видно, решение было найдено чуть ли не за 80 поколений.

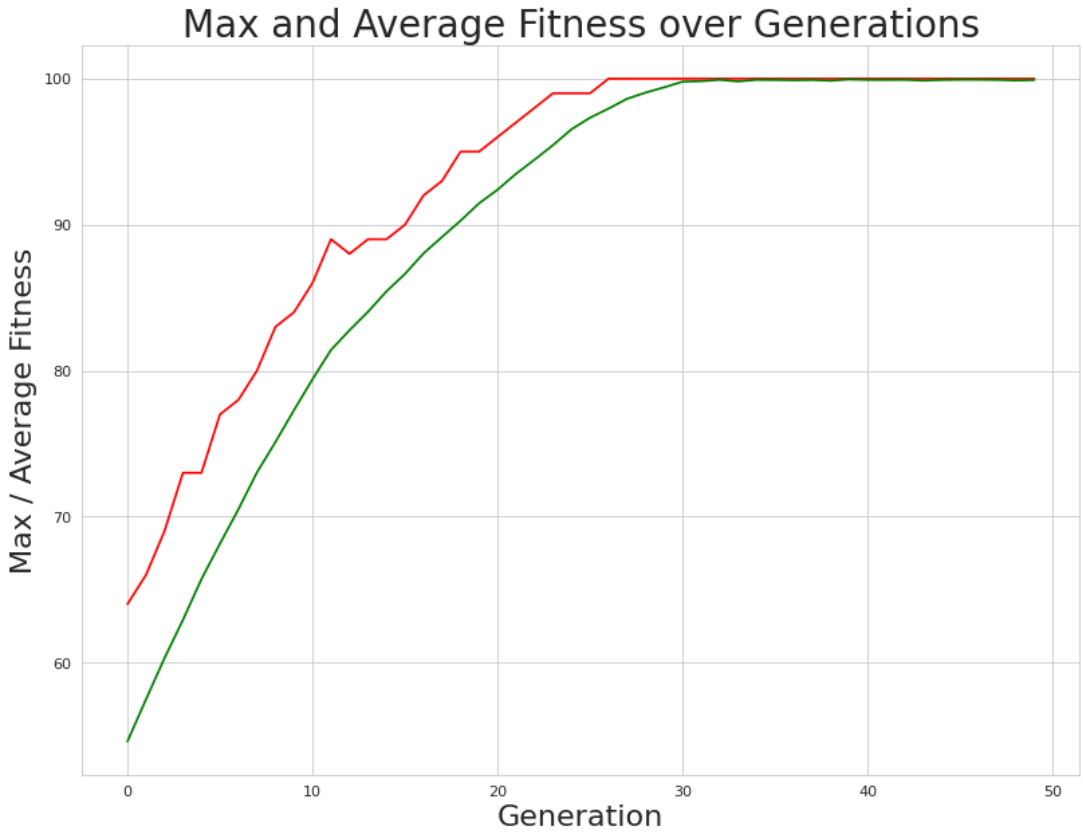
Рис. 4: Максимальное число поколений = 100



2.3. Оператор скрещивания

Откатив изменения, заменим одноточечное скрещивание двухточечным. Результат весьма ощутим: решение найдено мене чем за 30 поколений.

Рис. 5: Двухточечное скрещивание



2.4. Оператор мутации

Проведём эксперименты с оператором, отвечающим за случайную модификацию потомка. Для начала установим вероятность мутации индивидуума до 0.9. Несмотря на то, что такое значение очень высокое, график не выглядит непредсказуемым, случайным. Напротив, он обладает явной тенденцией.

Это связано с ещё одним параметром: вероятность инвертирования гена (бита). Увеличим его в 10 раз. Как и ожидалось, график заимел более непредсказуемую форму, несмотря на то, что в начале работы алгоритму удаётся улучшать результаты.

Рис. 6: Вероятность мутации индивида = 0.9

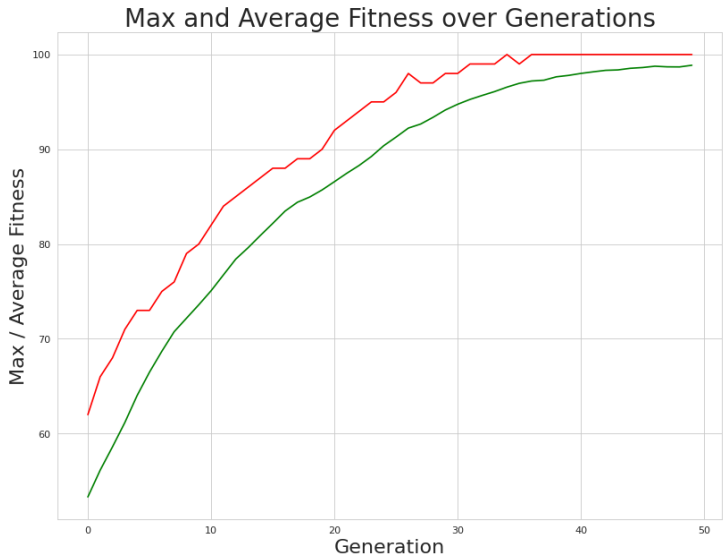
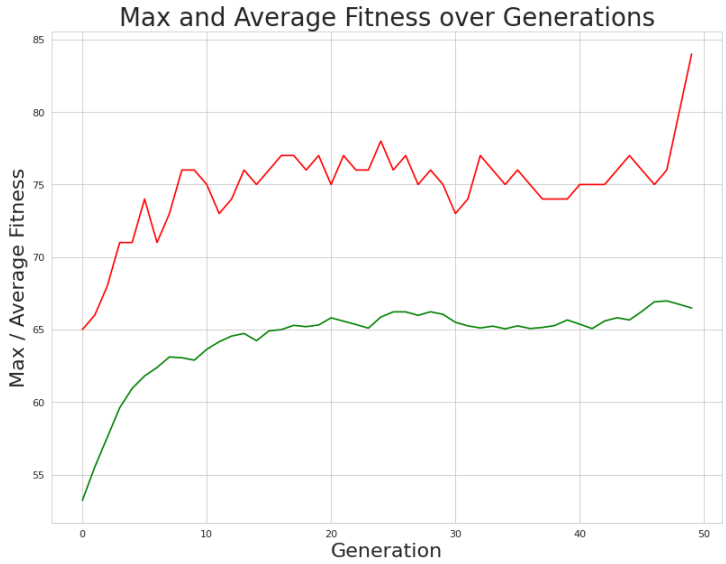


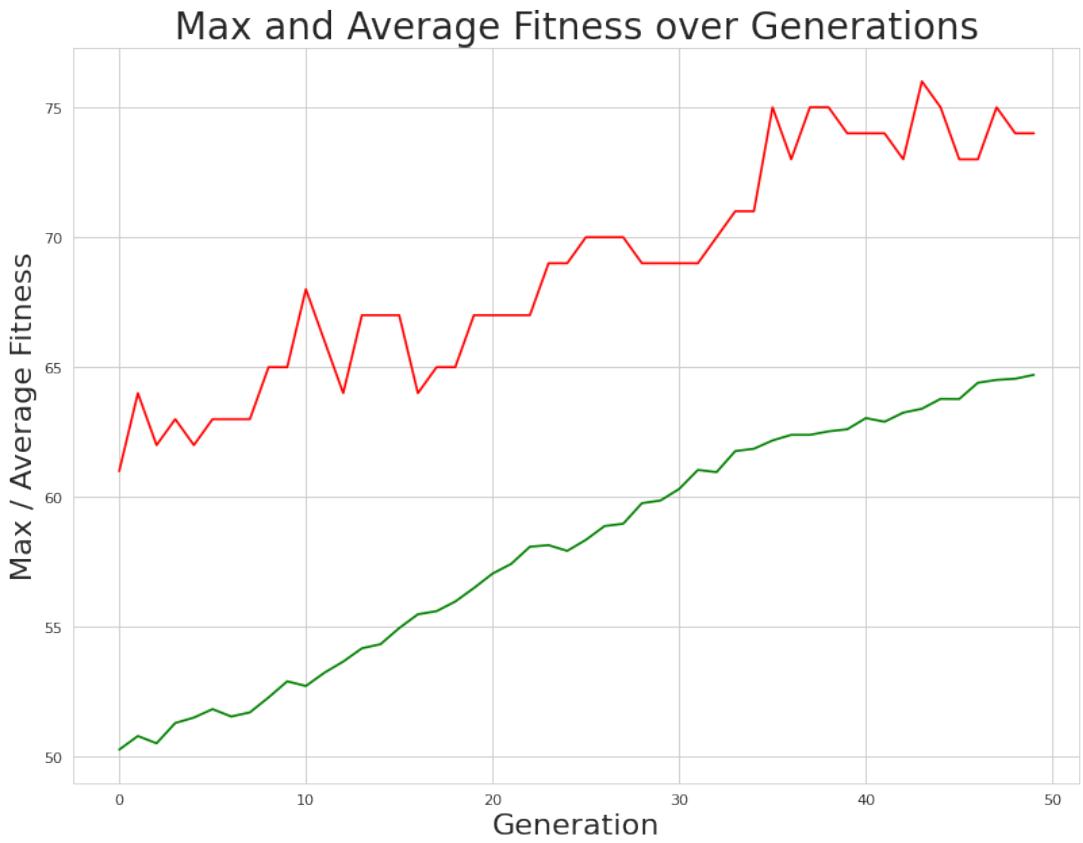
Рис. 7: Увелечение вероятности инвертирования в 10 раз



2.5. Оператор отбора

Теперь заменим турнирный отбор отбором по правилу рулетки.

Рис. 8: Отбор по правилу рулетки



3. Выводы

При выполнении этого задания я познакомился с пакетом для написания генетических алгоритмов DEAP, научился решать задачу OneMax, использовал различные операторы скрещивания и отбора, изменял параметры мутации.

Из самих экспериментов можно сделать следующие выводы:

1. При увеличении популяции для нахождения решения требуется меньше поколений. Однако с ростом размера популяции повышаются требования к вычислительной мощности и памяти.

2. Двухточечное скрещивание – более гибкий способ смешивания генов родителей по сравнению с одноточечным.

3. При высокой вероятности инвертирования бита и мутации алгоритм становится больше похож на случайный поиск.