

Генетический алгоритм для задачи оптимизации непрерывной функции.

Цель работы

Целью данной работы является получение студентом навыки разработки и анализа эволюционных операторов генетического алгоритма для решения задачи оптимизации непрерывной вещественнозначной функции.

Реализация

1. Генерация кандидатов была реализована как сэмплирование из $\sim U[-5; 5]^n$. Также в качестве распределения для генерации тестировалось $\mathcal{N}(0, 5/3)$, но оно показало худшие результаты.
2. Оператор мутации реализован как наложение нормального шума $\xi \sim \mathcal{N}(0, c)$, $c \sim U[0, 1]$ на cnt случайных координат элемента популяции. Координаты выбирались независимо. Лучший результат в экспериментах показал оператор с фиксированным значением $\text{cnt} = 1$.
3. В качестве оператора кроссовера был использован **Direction-Based Exponential Crossover Operator** из статьи [Amit Kumar Das & Dilip Kumar Pratihar](#). Данный оператор опирается на предполагаемое направление к оптимуму функции и показывает себя лучше, чем стандартные операторы (линейные комбинации предков, комбинаторные комбинации координат). Данный оператор использует параметр α . В ходе экспериментов мы установили значение $\alpha = 3.4$.

Результаты экспериментов

Размер проблемы	Размер популяции	Количество итераций	Результат
2	10	200	9.99
10	20	1000	9.98
20	10	7000	9.99
50	10	7000	9.95
100	50	9999	9.66

Вопросы

1. Кроссовер - детерминированная операция для любой фиксированной пары из популяции, в то время как некоторые операторы мутации могут покрыть всю область определения функции с некоторой вероятностью. Если убрать мутацию из алгоритма, то с помощью кроссовера будет найден локально оптимальный представитель решений, однако этот локальный минимум, скорее всего, невозможно будет покинуть на дальнейших итерациях. При этом если убрать

кроссовер и оставить мутацию, то алгоритм вырождается в некоторое подобие случайного блуждания. Такой подход все еще может найти оптимум функции, но никто не может гарантировать его сходимость за разумное время.

2. Такой параметр как размер популяции сильно влияет на быстродействие алгоритма оптимизации, т.к. заставляет проводить дополнительные вычисления целевой функции. Однако в ходе экспериментов данный параметр не показал значительного влияния на качество искомого решения, в отличие от количества эпох.
3. Область определения переменных целевой функции в некоторых случаях знать необходимо, т.к. если целевая функция не определена в какой-то области поиска, то и кандидатов из этой области оценить не выйдет. В других же случаях знание о границах области определения позволяет разумно генерировать(мутировать) кандидатов и ускорять сходимость.