

# Распределенные эволюционные алгоритмы

## Цель работы

Целью данной работы является освоение принципов построения распределенных и параллельных эволюционных алгоритмов для повышения их производительности и эффективности.

## Реализация

В качестве операторов генетического алгоритма были взяты результаты лабораторной работы 2.

1. Генерация кандидатов была реализована как сэмплирование из равномерного распределения.
2. Оператор мутации реализован как наложение нормального шума на  $n$  случайных координат элемента популяции. Координаты выбирались независимо. Лучший результат в экспериментах показал оператор с фиксированным значением  $n = 1$ .
3. В качестве оператора кроссовера был использован **Direction-Based Exponential Crossover Operator** из статьи [Amit Kumar Das & Dilip Kumar Pratihari](#). 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  $a$ . В ходе экспериментов мы установили значение  $a = 3.4$ .

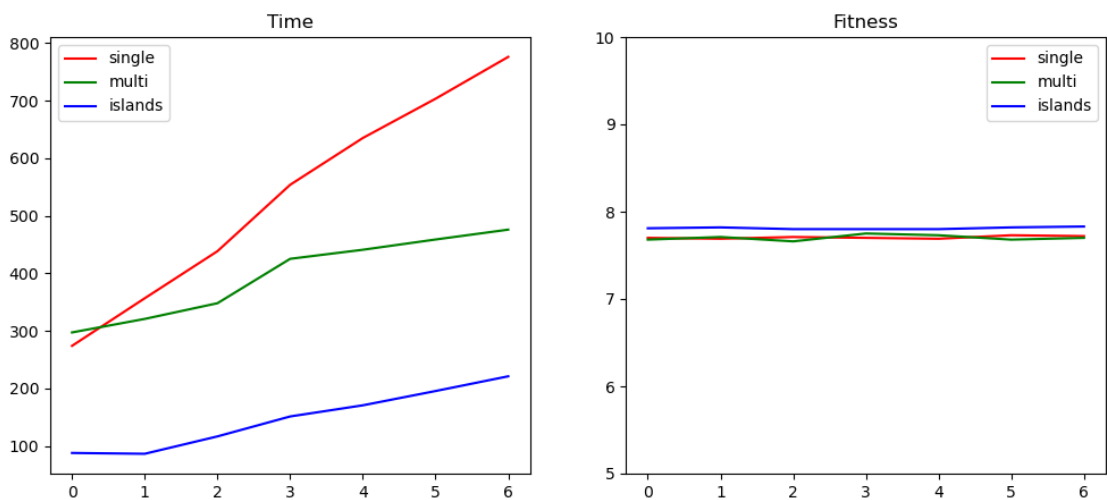
MasterSlaveAlg взят без изменений.

IslandsAlg использует симметричные модели эволюции на 5 островах. Из-за этого размер популяции каждого острова равен 20. Размер эпохи выставлен в 50, поэтому количество эпох в 50 раз меньше показателя **generations** для других алгоритмов (1000). В качестве стратегии миграции используется RingMigration.

## Результаты экспериментов

Complexity	Metric	Algorithm		
		Single	Multi	Islands
1	Time	274,12	297,38	88,06
	Result	7,7	7,68	7,81
2	Time	356,75	320,88	86,62
	Result	7,69	7,71	7,82
3	Time	438,5	348,12	116,81
	Result	7,71	7,66	7,8
4	Time	553,81	425,25	151,44
	Result	7,7	7,75	7,8

Complexity	Metric	Algorithm		
		Single	Multi	Islands
5	Time	635,12	441	170,81
	Result	7,69	7,73	7,8
6	Time	703,31	458,81	195,62
	Result	7,73	7,68	7,82
7	Time	776	475,94	221,19
	Result	7,72	7,7	7,83



Вопросы

- 1. Островной алгоритм показывает более высокое качество и большее быстродействие, однако назвать его однозначно лучшим нельзя, т.к. его параметры не сопоставляются с параметрами MasterSlave 1 к 1. (Надо бы количество вычислений fitness проверить)
- 2. Влияние увеличения размерности проблемы на MasterSlave будет схожим с влиянием на однопоточный алгоритм, хотя и кратно меньшим. На островную модель влияние будет более сильным.
- 3. Оба алгоритма замедлят свою работу из-за более частого вычисления fitness в поколении, однако вероятно вырастет качество найденного решения.
- 4. Мы ограничены лишь вычислительными ресурсами, позволяющими нам производить конечное количество процессов эволюции одновременно. Еще стоит учесть, что процесс миграции является барьером синхронизации для всех островов. И вообще, это все становится похоже на мета-эволюционный алгоритм, в котором остров является представителем решения, стандартные эволюционные операторы - мутации нашего представителя, а миграция - кроссовер.