

Отчёт: лабораторная работа №6

Студент: Золин И. М.

Группа: M4245

GitHub: <https://github.com/IMZolin/genetic-algs-lab6>

Задание: освоить принципов построения распределенных и параллельных эволюционных алгоритмов для повышения их производительности и эффективности.

Инициализация: генерация начальной популяции с равномерным распределением в диапазоне допустимых значений $[-5, 5]$.

Функция: используется Ackley function, усреднённая по параметру сложности (complexity).

Кроссовер: Арифметический кроссовер. Потомки формируются как линейная комбинация родителей:

$$c_1 = \alpha p_1 + (1 - \alpha)p_2, \quad c_2 = (1 - \alpha)p_1 + \alpha p_2$$

Мутация: используется гауссовская мутация генов (случайные флуктуации, $\sigma = 0.25$). После мутации значения ограничиваются допустимым диапазоном $[-5, 5]$.

Single-thread: все операции выполняются в одном потоке. Служит baseline для оценки скорости и качества.

Master-Slave: параллельная оценка фитнес-функции; повторения разбиваются на потоки, результаты усредняются. Ускоряет вычисления, не повышая качество решения.

Islands: популяция разбивается на несколько субпопуляций (островов), которые эволюционируют частично независимо и периодически обмениваются особями (миграция). Улучшает глобальный поиск, снижает риск застревания в локальных минимумах.

Параметры эксперимента: dimension=20, pop size=100, elitism=10, islands (число островов) - варьируется

Эксперименты

Метод	Complexity	Число итераций	Время выполнения	Результат
Single-thread	1	3000	0.296	9.954
Master-slave	1	3000	0.383	9.971
Islands	1	20	0.256	9.991
Single-thread	2	3000	0.360	9.932
Master-slave	2	3000	0.586	9.926

Islands	2	20	0.348	9.996
Single-thread	3	3000	0.477	9.951
Master-slave	3	3000	0.518	9.939
Islands	3	20	0.295	9.982
Single-thread	4	4000	0.619	9.962
Master-slave	4	4000	0.604	9.976
Islands	4	20	0.296	9.996
Single-thread	5	4000	0.694	9.961
Master-slave	5	4000	0.590	9.958
Islands	5	20	0.334	9.919
Single-thread	10	4000	0.944	9.969
Master-slave	10	4000	0.823	9.958
Islands	10	20	0.757	9.913
Single-thread	20	4000	1.60	9.963
Master-slave	20	4000	0.898	9.5831
Islands	20	20	0.803	9.902

- Islands демонстрируют лучшее качество решения при малом числе итераций.
- Master-Slave ускоряет вычисления (особенно при высокой сложности), но качество не улучшается.
- Single-thread стабилен, но медленный для больших размерностей и сложностей.
- При увеличении dimension качество решений снижается, но Islands остаются более устойчивыми к локальным минимумам.

Ответы на вопросы:

1. Какая модель алгоритма лучше при каких условиях?
 - Master-slave - при больших вычислительных нагрузках для ускорения.
 - Islands - для функций с множественными локальными минимумами (например, Ackley), обеспечивая глобальный поиск.
 - Single-thread - baseline, подходит для маленьких задач или тестирования.
2. Как повлияет увеличение размерности проблем на алгоритмы?
 - Экспоненциальный рост пространства поиска.
 - Увеличение числа локальных минимумов.
 - Замедление сходимости, сложность нахождения глобального оптимума растёт.

3. Как повлияет увеличение размера популяции?

- Плюсы: выше разнообразие, снижение риска преждевременной сходимости, лучшее качество решений.
- Минусы: линейное увеличение вычислительной сложности, рост времени на каждую итерацию.

4. Есть ли ограничение для количества островов?

- Формально ограничения нет.
- Практически: количество островов лучше соизмерять с числом процессоров/потоков, чтобы эффективно использовать параллелизм.