

Проблема преждевременной сходимости при поиске глобального экстремума функции векторного аргумента. Приньков А. С.

*Приньков Алексей Сергеевич / Prinkov Alexey Sergeevich – студент,
кафедра прикладной математики, факультет автоматизации и информатики,
Липецкий государственный технический университет, г. Липецк*

Аннотация: в статье анализируются методы эффективной реализации операторов генетического алгоритма и увеличение их стабильности относительно локальных экстремумов.

Abstract: this article analyzes methods of effective realization of genetic algorithm operators and increase of their stability relative to local extremums.

Ключевые слова: методы оптимизации, генетический алгоритм, экстремумы функции,

Keywords: optimization methods, genetic algorithm, maxima and minima

Задачи глобальной оптимизации часто возникают в оптимальном проектировании, планировании, конструировании, экономике, задачах идентификации, распознавания образов [1]. В этой статье будет рассматриваться генетический алгоритм, предложенный Джоном Холландом основанный на принципах естественного отбора Ч. Дарвина. Преобразование одного конечного нечеткого множества промежуточных решений в другое (с условием эффективного использования информации, накопленной в процессе эволюции) называют генетическим алгоритмом [2, с 74]. Он имеет ряд преимуществ перед градиентными методами: независимость от характера функции (дифференцируемость, непрерывность и т.д.), простота реализации, пригодность для решения крупномасштабных проблем оптимизации [3, с 50]. Генетический алгоритм помимо преимуществ, сталкивается с проблемами, в числе которых: изолированные функции, дополнительный шум, многоэкстремальные задачи [Рис. 1], нахождение точного глобального максимума и т.д. В виду большей распространенности и удобства представления в ЭВМ главное место уделим бинарным хромосомам, что позволит более детально рассмотреть операторов. Мы разберем проблему преждевременной сходимости, заключающаяся в нахождении локального экстремума вместо глобального, и некоторые методы ее решения.

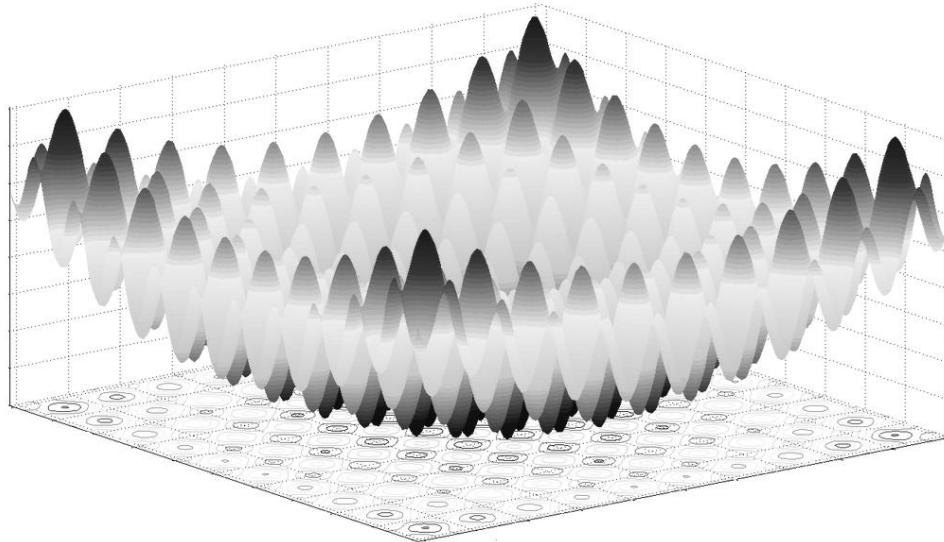


Рис 1. Функция Растригина для 2 переменных, используемая для тестирования алгоритмов оптимизации

Обобщенная схема классического генетического алгоритма такова:

1. Создание первоначальной популяции.
2. Отбор родителей.
3. Скрещивание и мутация.
4. Формирование нового поколения.

Для эффективного подхода к решению, следует использовать наиболее стабильные методики на каждом этапе. Как правило, начальная популяция формируется посредством генератора псевдослучайных чисел, если она окажется неконкурентоспособной, алгоритм вероятнее всего приведет её к благоприятному виду, нежизнеспособные индивиды будут элиминированы.

К отбору родителей целесообразно использовать генотипный и фенотипный аутбридинг, который направлен на предупреждение сходимости алгоритма, заставляя его просматривать новые, неисследованные области. Например, для бинарных хромосом можно использовать расстояние Хемминга как критерий отдаленности, а для вещественных норму в евклидовом пространстве. Также в некоторых случаях эффективны подходы: инбридинг, турнирный отбор, метод рулетки, различные модификации селекции [2].

Скрещивание в основном разделяют на дискретную рекомбинацию и кроссинговер. Первое в основном используют к хромосомам с вещественными генами. Второе, напротив, для бинарных. Оператор одноточечного кроссинговера выглядит следующим образом: $x_1x_2 \dots x_n x_{n+1} \dots x_m \bullet y_1y_2 \dots y_n y_{n+1} \dots y_m = x_1x_2 \dots x_n y_{n+1} \dots y_m$, где \bullet - операция скрещивания, $x_1x_2 \dots x_n x_{n+1} \dots x_m$ и $y_1y_2 \dots y_n y_{n+1} \dots y_m$ - родительские гены, n - точка разрыва. Методы кроссинговера отличаются количеством точек разрыва. В настоящий момент многие исследователи соглашаются, что двухточечный кроссинговер лучше, чем одноточечный [3, с 23].

Мутация один из важнейших инструментов генетического алгоритма. $x_1x_2 \dots x_n \rightarrow x'_1x'_2 \dots x'_n$, где $\exists x'_i \neq x_i$ хотя бы для одного $i \in \mathbb{N} \cap [1, n]$ Она способствует выходу популяции из локального экстремума, путем случайной подмены одного или нескольких генов. В ряде случаев будут возникать вредные мутации, которые будут уводить популяцию в локальные ямы. Поэтому для генетического алгоритма, мутации должны быть достаточно редкими событиями [4], а именно коэффициент вероятности мутации должен быть много меньше 1.

Наиболее действенным способом формирования нового поколения является отбор вытеснением. Где особям с разными генотипами отдается большее предпочтение. Тем самым достигается генетическая разнообразность и сохранение лучших решений. Этот метод хорошо подходит для обхода ложных оптимумов [5].

Концом работы алгоритма необходимо обозначить тот момент, когда популяция достигает равновесия. Алгоритм работает таким же образом, как и естественный отбор – каждое следующее поколение в среднем лучше, чем предыдущее [6].

Оценим алгоритм [Таблица 2] с его вышеперечисленными уточнениями, используя его реализацию под ЭВМ [Таблица 1], на языке Java. Пороговое время выполнения назначим 10 минут, по истечению которого считается, что программа дала неверный ответ. Для простоты вычисления и интерпретации будем анализировать функции [3, 61 - 62] от скалярного аргумента.

Таблица 1. Технические характеристики ЭВМ

Процессор	CPU CryptoHash	CPU Fibonacci	CPU Blowfish
4 x intel Core i5-4210U 2.7 GHz	287	1.82	4.71

Таблица 2. Реализация алгоритма с различными начальными популяциями, 50 запусков для каждой функции

Функция	Количество запусков	Правильных ответов, %
$f(x) = \sum_{i=1}^n x_i^2, x \in (-5.12; 5.12)$	50	100
$f(x) = \sum_{i=1}^n ix_i^2, x \in [-5.12; 5.12]$	50	100
$f(x) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2), x \in [-2.048; 2.048]$	100	86

$f(x) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i)),$ $x \in (-20; 20)$	100	78
--	-----	----

Можно сделать вывод, что такой способ является достаточно эффективным и, увеличивая количество запусков можно свести вероятность неправильного ответа к минимуму.

Алгоритм является универсальным в силу своей абстрактности. Генами могут выступать любые дискретные структуры, у которых определены критерии фенотипических признаков. С помощью вышеперечисленных рекомендаций можно многократно уменьшить вероятность сходимости к квазиоптимальному решению, как для бинарных, так и для обобщенных хромосом.

Литература

1. Жиглявский А.А., Жилинкас А.Г. Методы поиска глобального экстремума М.: Наука, 1991. 248 с.
2. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы М.: ФИЗМАТЛИТ, 2006. 320 с.
3. Панченко Т.В. Генетические алгоритмы: учебно – методическое пособие Астрахань: Издательский дом «Астраханский университет», 2007. 87 с.
4. Эрвин Шрёдингер Что такое жизнь с точки зрения физики? М.:РИМИС, 2015. 176 с.
5. Koza J. R. Genetic Programming Cambridge: The MIT Press, 1998. 609 с
6. Ричард Докинз Расширенный фенотип: длинная рука гена Москва: АСТ: CORPUS, 2014 – 512 с.