

## Введение

Многие задачи, возникающие в таких фундаментальных науках как физика, химия, молекулярная биология, а также во многих прикладных науках, сводятся к задачам непрерывной глобальной оптимизации.

Отличительной особенностью таких задач часто является нелинейность, недифференцируемость, многоэкстремальность (мульти-modalность), овражность (когда у функции в одном направлении нет изменений, а в другом есть), отсутствие аналитического описания (плохая формализованность), высокая вычислительная сложность, большая размерность пространства поиска, сложная область допустимых значений и т.д.

С самой общей точки зрения указанные особенности объясняют отсутствие универсального алгоритма их решения. Число таких алгоритмов оптимизации увеличивается, растёт количество их параллельных версий, ориентированных на различные классы параллельных вычислительных систем.

Для решения такого класса задач в 80-ых годах XX-го века интенсивно стали развиваться стохастические поисковые алгоритмы (в некоторых публикациях их также называют поведенческими, интеллектуальными, метаэвристическими, вдохновлёнными – или инспирированными – живой природой, роевыми, многоагентными).

Наиболее широко распространено понятие “популяционный алгоритм”. Популяционные алгоритмы предполагают одновременную обработку нескольких вариантов решения задачи оптимизации и представляют собой альтернативу классическим траекторным алгоритмам, в которых в области поиска эволюционирует только один кандидат на решение.

Задачи глобальной оптимизации можно разделить на два класса: детерминированные и стохастические. В первом случае оптимизируемая функция и функции, ограничивающие область решений задачи, являются детерминированными (точно определёнными), во втором случае одна или несколько указанных функций содержат случайные параметры.

Среди задач глобальной оптимизации выделяют статические, т.е. стационарные, и динамические. В статических задачах оптимизируемая функция и область её допустимых значений не меняются во времени, т.е. остаются неизменными положения в области поиска её локальных и глобальных экстремумов. В динамических задачах положения экстремумов могут меняться во времени и задача состоит в отслеживании движения экстремума.

Таким образом предметом рассмотрения являются детерминированные статические задачи глобальной безусловной (без ограничений на варьируемый параметр) и условной (с ограничением на это значение) непрерывной оптимизации.

По способу определения направления движения к экстремуму алгоритмы поисковой оптимизации делят на алгоритмы детерминированного (или регулярного) поиска (алгоритм градиентного спуска) и алгоритмы стохастического или случайного поиска.

Алгоритмы поисковой оптимизации разделяют ещё на два следующих класса: использующие как пробные, так и рабочие шаги алгоритма; алгоритмы, в которых эти шаги совмещены.

Все популяционные алгоритмы являются эвристическими, то есть для них сходимость к глобальному решению не доказано, но экспериментально установлено, что они дают достаточно хороший результат достаточно быстро.

В качестве общего названия для элементов популяции будем использовать термин агент или особь. Также возможны другие вариации (частица, индивид).

Общая схема всех популяционных алгоритмов включает следующую последовательность шагов:

1. Инициализация популяции. В области тем или иным образом создаётся некоторое число начальных приближений к искомому решению задачи, т.е. происходит процесс инициализации популяции агентов.
2. Миграция агентов. С помощью некоторого набора миграционных операторов, специфических для каждого популяционного алгоритма, агенты перемещаются в области поиска, чтобы приблизиться к искомому значению оптимизируемой функции.
3. Завершение поиска. Осуществляется проверка выполнения условий окончания итераций: если они выполнены – завершаем вычисления, принимаем лучшие значения агентов за приближённое решение задачи, иначе – переход к шагу 2.

При инициализации популяции можно использовать детерминированные или случайные алгоритмы. Формирование начальной популяции, агенты которой находятся вблизи глобального экстремума, может существенно сократить время решения задачи. Обычно априорная информация о местоположении экстремума отсутствует, поэтому агентов начальной популяции распределяют равномерно по всей области поиска.

Миграцию агентов, например, в генетическом алгоритме (ГА) реализуют генетические операторы (кроссовер, мутация).

В качестве условий окончания поиска, как правило, используется достижение заданного числа итераций или поколений. Также используется условие стагнации (проверка на изменения значений целевой функции: если изменений значения целевой функции практически нет, то останавливаем алгоритм).

Из представленной общей схемы следует, что популяционные алгоритмы обладают ярко выраженной модульной структурой, что позволяет легко получать большое число вариантов любого из алгоритмов путём варьирования и комбинации правил инициализации, миграции и завершения поиска.

Агенты популяции обладают следующим рядом свойств:

1. Автономность. Агенты движутся в пространстве хотя бы частично независимо друг от друга.
2. Стохастичность. Процесс миграции агентов всегда содержит случайную компоненту.

3. Ограниченность представления. Каждый агент обладает информацией только об исследуемой им части области поиска и возможных об окружении некоторых соседних агентов.

4. Децентрализация. Отсутствие агентов, управляющих процессом поиска.

5. Коммуникабельность.

Агенты тем или иным способом могут обмениваться информацией о топологии (ландшафте) оптимизируемой функции.

Даже если стратегия поведения каждого агента достаточно проста, указанные свойства обеспечивают формирование роевого интеллекта, определяющие организацию и сложное поведение популяции в целом.

Одной из особенностей популяционных алгоритмов является то, что в подавляющем большинстве случаев они имеют аналогии в человеческом обществе, живой и неживой природе: алгоритмы эволюции разума, колонии муравьёв, медоносных пчёл, сорной травы, гравитационного взаимодействия и т.д.

Популяционные алгоритмы в сравнении с классическими неоспоримые преимущества: прежде всего в задачах высокой размерности мультимодальных (мультиэкстремальных) и плохо формализуемых задачах. Важно также, что популяционные алгоритмы позволяют эффективнее классических искать субоптимальное решение (приблизительное).

Популяционные алгоритмы практически наверняка проиграют точному при решении задачи небольшой размерности. В случае гладкой мультимодальной функции популяционные алгоритмы менее эффективны, чем, например, градиентный спуск. К недостаткам также можно отнести зависимость от значений свободных параметров.

Важнейшим понятием популяционного алгоритма является fitness-функция (целевая функция, функция пригодности, функция приспособленности). Она используется для оценки качества агентов популяции. Часто, но не всегда fitness-функция совпадает с оптимизируемой функцией.

С используемой fitness-функцией связана суть популяционных алгоритмов, заключающаяся в обеспечении более высокой средней приспособленности популяции агентов данного поколения по сравнению с приспособленностью предыдущего поколения.

Одна из основных проблем конструирования популяционных алгоритмов – обеспечение баланса между интенсивностью (скоростью сходимости) и шириной поиска, т.е. диверсификацией.

Первые поколения популяционного алгоритма занимаются исследованием области поиска, агенты последних поколений заняты уточнением исключительно найденных решений. Баланса между интенсивностью и шириной можно добиться, используя механизмы адаптации и самоадаптации.

Основными критериями эффективности популяционных алгоритмов являются надёжность, т.е. вероятность локализации глобального экстремума; а также скорость сходимости, т.е. оценка матожидания необходимого числа испытаний.

Классификация популяционных алгоритмов:

1. ЭА (эволюционные алгоритмы), в том числе ГА (генетические алгоритмы).
2. Популяционные алгоритмы, вдохновлённые живой природой.
3. Алгоритмы, вдохновлённые неживой природой.
4. Алгоритмы, инспирированные человеческим обществом.
5. Прочие алгоритмы.

**Постановка и классификация алгоритмов решения детерминированной задачи  
поисковой оптимизации**