

Интернет-журнал «Наукоедение» ISSN 2223-5167 <https://naukovedenie.ru/>

Том 9, №6 (2017) <https://naukovedenie.ru/vol9-6.php>

URL статьи: <https://naukovedenie.ru/PDF/46TVN617.pdf>

Статья опубликована 21.12.2017

Ссылка для цитирования этой статьи:

Карпенко А.П. Основные сущности популяционных алгоритмов глобальной оптимизации. Опыт систематизации // Интернет-журнал «НАУКОВЕДЕНИЕ» Том 9, №6 (2017)

<https://naukovedenie.ru/PDF/46TVN617.pdf> (доступ свободный). Загл. с экрана. Яз. рус., англ.

Работа поддержана проектом РФФИ 16-07-00287

УДК 519.6

Карпенко Анатолий Павлович

НОУ ВПО «Московский государственный Баумана», Россия, Москва¹

Зав. кафедрой

Доктор физико-математических доцент

E-mail: apkarpenko@mail.ru

РИНЦ: https://elibrary.ru/author_profile.asp?id=338847

Основные сущности популяционных алгоритмов глобальной оптимизации. Опыт систематизации

Аннотация. В популяционных алгоритмах (population algorithms) глобальной оптимизации, во-первых, число кандидатов в решение (особей) больше единицы. Во-вторых, на каждой из итераций популяционного алгоритма перемещаются либо все особи, исключая, возможно, лишь некоторые из них (например, «лучшие» особи), либо некоторое число особей, превышающее единицу. В разных публикациях популяционные алгоритмы (П-алгоритмы) глобальной оптимизации называют поведенческими, интеллектуальными, метаэвристическими, вдохновленными (инспирированными) природой, роевыми, многоагентными и т. д.

Рассматриваем П-алгоритмы глобальной непрерывной оптимизации. В связи большим и постоянно увеличивающимся числом таких алгоритмов актуальной является проблема систематизации их выразительных средств. В статье выполнен анализ одной из составляющих этой проблемы – проблемы систематизации сущностей П-алгоритмов.

Представляем постановку задачи глобальной непрерывной оптимизации и общую схему П-алгоритмов ее решения. Выделяем следующие основные сущности: свободные параметры, пространство соседства, особь, популяция и объединение особей, окрестность особи и другие. Раскрываем суть этих сущностей, представляем их характеристики, приводим варианты использования в различных П-алгоритмах.

Ключевые слова: глобальная оптимизация; популяционный алгоритм; метаэвристический алгоритм; особь; популяция; пространство соседства

¹ 105005, Россия, Москва, 2-я Бауманская ул., д. 5, каф. РК6

Введение

Особенностями задач глобальной оптимизации, возникающих в настоящее время во многих приложениях, являются нелинейность, недифференцируемость, многоэкстремальность (мульти-modalность), овражность, отсутствие аналитического выражения (плохая формализованность) и высокая вычислительная сложность оптимизируемых функций, высокая размерность пространства поиска, сложная топология области допустимых значений и т. д.

Различают траекторные и популяционные алгоритмы поисковой глобальной оптимизации. *Траекторные* алгоритмы (*single-state algorithms*) предполагают обновление на каждой итерации положения лишь одного кандидата в решение задачи. При этом общее число кандидатов может быть большим единицы, и на разных итерациях могут перемещаться разные кандидаты. В *популяционных* алгоритмах (*population algorithms*), во-первых, число кандидатов (*особей*) больше единицы, а, во-вторых, на каждой из итераций перемещаются либо все особи, исключая, возможно, лишь некоторые из них (например, «лучшие» особи), либо, по меньшей мере, некоторое число особей, превышающее единицу. В разных публикациях популяционные алгоритмы глобальной поисковой оптимизации (П-алгоритмы) называют поведенческими, интеллектуальными, метаэвристическими, вдохновленными (инспирированными) природой, роевыми, многоагентными и т. д. [1].

Известны многочисленные примеры успешного решения с помощью П-алгоритмов сложных практических задач глобальной оптимизации, например, задач автоматизированного проектирования, синтеза сложных химических соединений, оптимального управления динамическими системами. П-алгоритмы также успешно используются в алгоритмах многокритериальной оптимизации, предполагающих предварительное построение некоторой аппроксимации множества (фронта) Парето соответствующей задачи многокритериальной оптимизации.

Популяционные алгоритмы (П-алгоритмы) многочисленны и весьма разнообразны. Так в книге [2] представлено более 130 таких алгоритмов. Продолжают появляться новые П-алгоритмы. Многообразие П-алгоритмов порождает проблему обобщения и систематизации их сущностей, базовых принципов, используемых операторов и т. д. (см., например, фундаментальные работы [3, 4]). Данная работа посвящена одному из аспектов этой проблемы – проблеме систематизации основных сущностей П-алгоритмов.

Приводим постановку задачи глобальной оптимизации и общую схему П-алгоритмов ее решения. Выделяем следующие основные сущности: свободные параметры, пространство соседства, особь, популяция и объединение особей, окрестность особи и другие. Раскрываем суть этих сущностей, представляем их характеристики, приводим варианты использования в различных П-алгоритмах.

Все П-алгоритмы, которые упоминаются в работе, можно найти в [1, 2]. Приводим также ссылки на оригинальные работы, посвященные этим алгоритмам.

1. Постановка задачи глобальной оптимизации и П-алгоритмы ее решения

Постановка задачи

Рассматриваем *детерминированную* задачу *глобальной безусловной* минимизации

$$\min_{X \in R^{|X|}} f(X) = f(X^*) = f^* \quad (1)$$

где: $|X|$ – размерность вектора варьируемых параметров $X = (x_i, i \in [1:|X|])$; $f(X)$ – целевая функция (Ц-функция) со значениями в пространстве R^1 ; X^* , f^* – искомые оптимальные решение и значение Ц-функции соответственно.

Полагаем, что поиск решения задачи (1) начинается в $|X|$ -мерном параллелепипеде $\Omega_X = \{X \mid X^- \leq X \leq X^+\}$, где $X^- = (x_i^-, i \in [1:|X|])$, $X^+ = (x_i^+, i \in [1:|X|])$ – его нижняя и верхняя границы. Другими словами, полагаем, что в этой области осуществляется инициализация популяции.

Рассматриваем также аналогичную задачу глобальной условной минимизации

$$\min_{X \in \Omega \subset R^{|X|}} f(X) = f(X^*) = f^* \quad (2)$$

где: $\Omega = \{X \mid \Gamma(X) \geq 0\}$ – выпуклое множество допустимых значений вектора X ; $\Gamma(X) = (\gamma_j(X), j \in [1:|\Gamma|])$ – ограничивающая вектор-функция; неравенство понимается покомпонентно.

Испытанием называют операцию однократного вычисления значений Ц-функции $f(X)$ и ограничивающих функций $\Gamma(X)$, а также, быть может, производных указанных функций в некоторой допустимой точке X . Особенность задач оптимизации, представляющих практический интерес, состоит в том, что каждое испытание может требовать больших затрат компьютерного времени.

Приближение к решению задачи (кандидата в решение) называем *особью* и обозначаем $s_i \in S$. Особь s_i определяют ее вектор *состояния* $V_i = V_i(t)$ и соответствующее значение *фитнесс-функции* (Ф-функции) $\varphi_i = \varphi_i(t)$, называемой *приспособленностью* особи. Здесь t – текущий номер итерации (поколения эволюции популяции). Полагаем, что Ф-функция $\varphi(X)$, как и Ц-функция $f(X)$, подлежит минимизации.

Популяция (субпопуляция) есть совокупность особей $S = \{s_i, i \in [1:|S|]\}$. Число особей $|S|$ в популяции (*размер популяции*) является свободным параметром П-алгоритма и может, вообще говоря, меняться в процессе эволюции популяции. Отличие субпопуляции от (одиначной) популяции заключается в том, что особи субпопуляции коммутируют (обмениваются данными) с особями других субпопуляций данной *мультипопуляции*. Субпопуляции эволюционируют, по меньшей мере, на логическом уровне параллельно. Подавляющее большинство П-алгоритмов в их канонической форме использует одиначные популяции.

Процессом эволюции особи $s_i \in S$ называем последовательность *шагов* ее эволюции (шагов поиска) в течение $[0:\hat{t}]$ поколений (в течение всего итерационного процесса), то есть, цепочку ее состояний $V_i(0), V_i(1), \dots, V_i(t), \dots, V_i(\hat{t})$. Здесь $\hat{t}$ – номер последнего поколения (итерации). Аналогично определяются процессы эволюции популяции S и мультипопуляции.

Общая схема П-алгоритма имеет следующий вид.

- 1) Инициализируем алгоритм. Задаем начальное значение счетчика числа итераций $t = 0$, начальные состояния особей $V_i(0)$, $i \in \overline{1:|S|}$, то есть генерируем начальную популяцию, а также задаем значения свободных параметров алгоритма.
- 2) Выполняем, вообще говоря, свои для каждой из составляющих текущего вектора $V_i = V_i(t)$ эволюционные операторы данного П-алгоритма, в результате чего особь приобретает новое положение в области поиска $X_i(t+1) = X'_i$, новую скорость $\Delta X_i(t+1) = \Delta X'_i$ и новое ускорение $\Delta^2 X_i(t+1) = \Delta^2 X'_i$; $i \in \overline{1:|S|}$.
- 3) Проверяем выполнение условий завершения эволюционного процесса. Если эти условия не выполнены, полагаем $t = t + 1$ и возвращаемся к шагу 2. В противном случае, принимаем лучшее из найденных положений особей в качестве приближенного решения задачи.

С достаточно общей точки зрения эволюцию особи S_i в ходе итераций определяет функция (эволюционный оператор, эволюционная формула) вида

$$V_i^{t+1} = \Sigma(V_j^\tau, f(X_j^\tau), \Gamma(X_j^\tau), j \in \overline{1:|S|}, \tau \in \overline{1:t}), i \in \overline{1:|S|}, t+1 \leq \hat{t}. \quad (3)$$

Смысл формулы (3) заключается в том, что новое состояние особи S_i определяют, вообще говоря, все предыдущие состояния всех особей популяции, включая данную особь, а также соответствующие значения Ц- и ограничивающих функций.

В П-алгоритмах оператор (3) является стохастическим, то есть включает в себя один или несколько параметров, которые на различных итерациях принимают случайные значения с заданными законами распределения. Если популяция является неоднородной (типизированной), то итерационная формула (3) может быть различной для разных типов особей (п.4). Если процесс эволюции популяции включает в себя микрошаги (см. ниже), то ее эволюционная формула представляет собой последовательность эволюционных операторов вида (3). В многостадийном П-алгоритме [1] на каждой из стадий используются свои эволюционные операторы, которые в качестве исходных данных для последующей стадии используют данные предыдущей или предыдущих стадий.

В качестве приближенного решения задачи оптимизации принимаем вектор $\tilde{X}^*$, который определяет равенство

$$\tilde{f}^* = f(\tilde{X}^*) = \min_{t,i} f(X_i^t), t \in \overline{1:\hat{t}}, i \in \overline{0:|S|}. \quad (4)$$

Классификация П-алгоритмов

П-алгоритмы могут быть классифицированы на основе большого числа классификационных признаков.

Природные прототипы. Среди П-алгоритмов выделяем следующие классы алгоритмов: эволюционные алгоритмы, включая генетические; алгоритмы, вдохновленные живой природой; алгоритмы, вдохновленные неживой природой; алгоритмы, инспирированные человеческим обществом; прочие алгоритмы [1]. Заметим, что в работе [2] использована более детальная классификация, выделяющая алгоритмы вдохновленные биологией (*Biology-based*

Algorithms), физикой (*Physics-based Algorithms*), химией (*Chemistry-based Algorithms*), математикой (*Mathematics-based Algorithms*).

Уровень гибридации. Различаем простые (не композиционные) и композиционные П-алгоритмы. К *простым* относим П-алгоритмы, которые либо не используют гибридацию, либо используют лишь низкоуровневую гибридацию вложением, когда имеет место сращивание гибридируемых алгоритмов так, что, по сути, ее результатом является новый алгоритм, который не может быть декомпозирован на составляющие его алгоритмы [1]. К *композиционным*, полагаем, принадлежат П-алгоритмы, построенные на основе всех способов гибридации, исключая низкоуровневую гибридацию вложением.

Число предыдущих учитываемых итераций. В ситуации, когда при вычислении состояния $V_i(t+1)$, $i \in [1:|S|]$ учитывают информацию только об одной (предыдущей) итерации, П-алгоритм называется *одношаговым*. Если при вычислении этого состояния учитывается информация не более чем о $r > 1$ предыдущих итерациях, то П-алгоритм называется *многошаговым* (точнее, r -шаговым) или алгоритмом *с памятью*.

Тип итераций. Различаем П-алгоритмы с простыми и составными итерациями. В первом случае на каждой из итераций эволюционный оператор (либо их некоторая совокупность) используется однократно. Во втором случае каждая итерация состоит из нескольких итерационных микрошагов. Например, в алгоритме бактерий *BFO* (Bacterial Foraging Optimization) на каждой итерации выполняется три шага – хемотаксис, репродукция, ликвидация и рассеивание [5].

Основные обозначения и определения

Для сущностей П-алгоритмов используем общее обозначения вида

$$\Xi (Arg^{\Xi}, P^{\Xi}) = \Xi | A^{\Xi} (Arg^{\Xi}, P^{\Xi}),$$

где: A^{Ξ} – алгоритм или совокупность алгоритмов, определяющих сущность Ξ ; Arg^{Ξ} – набор аргументов этого алгоритма; P^{Ξ} – набор *свободных* параметров, используемых сущностью Ξ . Там, где это не вызывает опасности неверного чтения, опускаем в записях Arg^{Ξ} , P^{Ξ} указание на принадлежность к сущности Ξ и определяем эту сущность в виде

$$\Xi (Arg, P) = \Xi | A^{\Xi} (Arg, P).$$

На этапе инициализации П-алгоритма осуществляется инициализация его сущностей, заключающаяся в задании (начальных) значений аргументов алгоритма A^{Ξ} и/или инициализации значений его свободных параметров.

С каждой из сущностей П-алгоритма может быть связана одна или несколько *характеристик* этой сущности, в общем виде обозначаемая как

$$Feature(\Xi, Arg^{Feature}, P^{Feature}) = Feature | A^{Feature, \Xi} (Arg^{Feature}, P^{Feature}).$$

Различаем две группы характеристик сущности: величины, использованные при определении этой сущности; вычисляемые (с помощью алгоритма $A^{Feature, \Xi}$) характеристики. Различаем также статические и динамические характеристики. *Статическими* называем

характеристики, которые определяет «статический» алгоритм $A^{Feature, \Xi}$, то есть алгоритм, который не изменяется в процессе эволюции популяции на протяжении всего поиска – до окончания завершения итерационного процесса. Напротив, *динамическими* объявляем характеристики сущности, значения которых меняются в процессе эволюции популяции вследствие эволюции алгоритма $A^{Feature, \Xi}$.

Эволюция алгоритма A^{Ξ} в процессе поиска (итерационного процесса) может являться следствием наличия в алгоритме свободных динамических параметров (п.2), а также следствием прямой зависимости алгоритма от номера итерации t . Кроме того, «динамичность» сущностей может быть обусловлена их однократной или многократной «переинициализацией». Например, модификация COAC (Continuous Orthogonal Ant Colony) муравьиного П-алгоритма использует варианты такой инициализации, которые составляют суть процедуры *модуляции* [6].

Подчеркнем следующее обстоятельство. Характеристики каждой из сущностей П-алгоритма могут менять свои значения в силу эволюции популяции в процессе поиска. В этом смысле любая из сущностей может рассматриваться как динамическая. Мы, однако, рассматриваем как динамические только те сущности, значения характеристик которых меняются вследствие изменения алгоритма A^{Ξ} в процессе эволюции популяции.

Соответственно классификации характеристик сущностей на статические и динамические, все сущности разделяем на статические и динамические. Выделяем динамические *неадаптивные (программные)* и динамические *адаптивные* сущности, частным случаем которых являются динамические сущности, *управляемые событиями* в эволюции популяции [1].

2. Свободные параметры

Одной из важных особенностей П-алгоритмов является наличие в них *свободных* параметров, значения которых, в конечном счете, лицо, принимающее решения, (ЛПР) устанавливает, исходя из своих предпочтений и/или особенностей исходной оптимизационной задачи. Свободные параметры могут быть бинарными, дискретными, натуральными, целыми, вещественными и комплексными. Практически во всех П-алгоритмах свободным параметром является размер популяции $|S|$. Как правило, авторы П-алгоритмов предлагают свои рекомендованные значения свободных параметров этих алгоритмов.

В качестве общего обозначения набора свободных параметров П-алгоритма используем

$$P(\bar{P}) = P|A^P(\bar{P})$$

, где A^P – алгоритм или правила определения значений компонентов

набора параметров P ; $\bar{P}$ – набор свободных *мета-параметров* алгоритма A^P , то есть *вспомогательных* свободных параметров, которые определяют значения *основных* свободных параметров P (см. ниже).

Значения компонентов набора P ЛПР может определять непосредственно или опосредованно (алгоритмически), задавая, например, диапазон возможных значений того или иного параметра и указывая, что конкретное значение этого параметра должно быть равномерно случайно распределено в указанном интервале. В первом случае используем обозначение $P = P|A_{DM}^P$ (от англ. *Decision-Maker*), а во втором – $P = P|A_{Alg}^P$.

Очевидно, возможно комбинированное определение компонентов набора P , когда часть этих компонентов ЛПР определяет непосредственно, часть – алгоритмически. Пусть, например, $P = \{a, b\}$. Тогда возможна запись вида $a = A_{DM}^P$, $b = A_{Alg}^P$.

Разделяем свободные параметры на статические и динамические. Значения *статических* свободных параметров остаются постоянными на протяжении всего поиска, то есть, до завершения процесса эволюции популяции (итерационного процесса). *Динамические* свободные параметры П-алгоритма изменяют свои значения в процессе поиска (обычно, по простым, но эффективным правилам). Значительным стимулом для использования динамических параметров является тот факт, что применение даже простого правила варьирования значений этих параметров часто дает существенный выигрыш в эффективности алгоритма по сравнению с использованием оптимальных статических значений. Свободные мета-параметры всех классов параметров являются, как правило, статическими.

2.1 Статические свободные параметры

Значения статических свободных параметров ЛПР может назначать непосредственно на этапе инициализации П-алгоритма, либо задавать опосредованно с помощью того или иного

правила, так что в этом случае имеем
соответственно.

$$P(\bar{P}) = P \Big| A_{Alg}^P(\bar{P}) \quad \text{или} \quad P(\bar{P}) = P \Big| A_{DM}^P(\bar{P})$$

Инициализацию значений статических свободных параметров (а также начальных значений динамических свободных параметров) реализуют на основе их значений или диапазонов значений, рекомендованных авторами соответствующего П-алгоритма. Эти рекомендации могут быть заданы в двух формах:

- фиксированные значения или интервалы значений;
- функциональные зависимости этих значений от параметров исходной задачи оптимизации и/или от значений других свободных параметров алгоритма, переменных состояния популяции, характеристик особей и их объединений и т. д.

Примером фиксированных рекомендованных значений статических свободных параметров являются значения весов инерционной b_I , когнитивной b_C и социальной b_S составляющих скорости особей в каноническом алгоритме роя частиц *PSO* (Particle Swarm Optimization), равные $b_I = 0,7298$, $b_C = b_S = 1,49618$ [7]. Фиксированные интервалы значений свободных параметров, рекомендуют, например, авторы алгоритма поиска гармонии *HS* (Harmony Search) для размера популяции $|S|$, вероятности ξ_h случайного выбора (*random selection*) из памяти гармоний, вероятности ξ_p изменения шага (*Pitch Adjusting Rate, PAR*): $|S| \leq 50$; $\xi_h = 0,7 \div 0,95$; $\xi_p = 0,1 \div 0,5$ [8].

В качестве примера определения значений статических свободных параметров в форме функциональной зависимости от значений параметров задачи приведем определение размера популяции в муравьином алгоритме *CACO* (Continuous Ant Colony Optimization) [9]:

$$|S| = |S|^+ (1 - \exp(-0,1|X|)) + 5.$$

Здесь: $|S|^+$ – максимально допустимый размер популяции (свободный мета-параметр).

Если рекомендации авторов П-алгоритма даны в форме интервалов возможных значений свободных параметров, то эти значения, во-первых, могут назначаться ЛПП в указанных интервалах, исходя из его неформальных соображений, например, на основе предыдущего опыта эксплуатации данного П-алгоритма. Во-вторых, могут быть использованы случайные значения параметров в тех же интервалах. При этом, как правило, используют равномерные законы распределения. Например, в алгоритме летучих мышей *BI* (Bat-Inspired), начальные значения частот ϖ_i , громкостей a_i и частот повторения импульсов r_i полагают равномерно распределенными в заданных интервалах $[\varpi^-; \varpi^+]$, $[a^-; a^+]$, $[0; 1]$ [10].

2.2 Динамические свободные параметры

Свободные динамические параметры требуют, во-первых, задания правил инициализации своих начальных значений. Во-вторых, для этих параметров должны быть определены законы изменения их значений в процессе поиска. На этом основании полагаем, что алгоритм A^P в определении динамических параметров $P = P(t)$ представляет собой совокупность двух алгоритмов, то есть $A^P = \{A^{P.init}, A^{P.evol}\}$, где алгоритмы $A^{P.init}$, $A^{P.evol}$ реализуют решение первой и второй из указанных задач соответственно.

На значения динамических параметров могут накладываться разного рода ограничения. Область допустимых значений набора динамических свободных параметров в общем виде обозначаем D_P . Обычно используют двухсторонние ограничения на значения параметров вида $p^- \leq p \leq p^+$, где $p^- < p^+$ – нижняя и верхняя границы допустимых значений параметра $p \in P$ соответственно. Величины p^- , p^+ являются свободными мета-параметрами П-алгоритма: $\{p^-, p^+\} \in \bar{P}$. Например, алгоритм оптимизации косяком рыб *FSS* (Fish School Search) использует такую характеристику особи S_i как ее вес w_i [11], и ограничения на значения этой характеристики имеют вид $1 \leq w_i \leq w^+$, так что свободными мета-параметрами являются величины $w^- = 1$; w^+ ; $i \in [1:|S|]$.

Динамические детерминированные свободные параметры

Динамические неадаптивные (программные) параметры. В П-алгоритмах используют простые алгоритмы $A^{P.evol}$ программного изменения значений этих параметров, которые обычно определяют монотонно убывающие или возрастающие линейные или нелинейные непрерывные или ступенчатые функции текущего номера итерации t .

Пусть, например, речь идет о генетическом алгоритме и операторе гауссовой мутации вида

$$x'_{i,j} = x_{i,j} + N_1(0, \sigma), \quad i \in [1:|S|], \quad j \in [1:|X|], \quad (5)$$

где: величина σ (шаг мутации) является свободным параметром алгоритма. Известное неадаптивное эвристическое правило изменения этой величины определяет формула

$$\sigma(t) = 1 - 0,9 \frac{t}{\hat{t}}, \quad (6)$$

в соответствии с которой с ростом числа итераций t шаг мутации линейно уменьшается с единицы при $t = 0$ до 0,1 в конце итераций при $t = \hat{t}$ [12]. Таким образом, в данном случае алгоритм $A^{P.evol}(\bar{P})$ определяет формула (6), и свободным мета-параметром можно считать величину 0,9.

Если значения некоторого свободного параметра P должно линейно уменьшаться (увеличиваться) с ростом числа итераций t в диапазоне $[p^+; p^-]$ ($[p^-; p^+]$), то правило изменения значений этого параметра, то есть (алгоритм $A^{P.evol}(p^-, p^+)$ имеет, очевидно, вид

$$p(t) = p^+ - \left(\frac{p^+ - p^-}{\hat{t}} \right) t \quad \left(p(t) = p^- - \left(\frac{p^- - p^+}{\hat{t}} \right) t \right).$$

Неадаптивное рекуррентное правило изменения значений некоторого свободного параметра P можно определить формулой вида

$$p' = p(t+1) = b p; \quad p(0) = p^0, \quad b \in (0; 1); \quad \{p^0, b\} \in \bar{P}.$$

Динамические адаптивные параметры. Примером адаптивного динамического правила изменения значения свободного параметра σ (алгоритм $A^{P.evol}(t_a, c_m)$) в операторе гауссовой мутации (5) является правило $\frac{1}{5}$ Рехенберга (*I. Rechenberg*) [13], которое может быть записано в виде следующей ступенчатой функции:

- если номер текущий итерации t кратен периоду адаптации t_a , то

$$\sigma(t) = \begin{cases} \frac{\sigma(t-t_a)}{c_m}, & \varpi_s < \frac{1}{5}, \\ \sigma(t-t_a) c_m, & \varpi_s > \frac{1}{5}, \\ \sigma(t-t_a), & \varpi_s = \frac{1}{5}; \end{cases} \quad (7)$$

- в противном случае $\sigma(t) = \sigma(t-1)$.

Здесь: ϖ_s – средняя относительная доля успешных мутаций, измеренная на основе результатов эволюции популяции в течение t_a поколений (свободный мета-параметр); $c_m \in [0,817; 1,0]$ – заданная константа (свободный мета-параметр). Успешной называется мутация, порождающая потомка лучшего, чем родитель. Ситуацию, когда величина ϖ_s

отклоняется от своего значения, равного $\frac{1}{5}$, можно интерпретировать как событие в эволюции популяции.

Стохастические динамические свободные параметры

Динамические неадаптивные параметры. Значения стохастических динамических неадаптивных параметров могут быть получены следующими путями:

- генерация последовательности случайных чисел, подчиненных требуемому закону распределения;
- генерация хаотичной последовательности;
- рандомизация соответствующей детерминированной зависимости.

Например, случайное неадаптивное изменение параметра $p \in P$ может быть выполнено по следующим правилам (алгоритмы $A_{uniform}^{P.evol}(p_0)$, $A_{norm}^{P.evol}(p_0, \alpha)$ соответственно):

$$p' = p_0 + \frac{U[0;1]}{2}, \quad p' = p_0 + \alpha N(0;1).$$

Здесь: p_0, α – свободные мета-параметры; $U[0;1]$ – случайное вещественное число, равномерно распределенное в интервале $[0;1]$; $N[0;1]$ – аналогичное число, распределенное по нормальному закону с нулевым математическим ожиданием и единичным средне-квадратичным отклонением.

Динамические адаптивные параметры. Значения стохастических динамических адаптивных параметров получают путем рандомизации соответствующих детерминированных зависимостей. Например, можно предложить стохастический вариант правила $\frac{1}{5}$ Рехенберга (7)

$$\sigma(t) = N(0; \beta) \begin{cases} \frac{\sigma(t-t_a)}{c_m}, & \varpi_s < \frac{1}{5}, \\ \sigma(t-t_a) c_m, & \varpi_s > \frac{1}{5}, \\ \sigma(t-t_a), & \varpi_s = \frac{1}{5}, \end{cases},$$

где: $\beta \in P$.

3. Пространства соседства особей и объединений особей

О соседстве особей и объединений особей (п.5) можно говорить в терминах следующих *пространств соседства*: фитнес-пространство R_Φ ; пространство поиска R_X ; *топологическое* пространство R_T . Элементами этих пространств являются, соответственно, значения Ф-функции $\varphi(X)$, компоненты вектора варьируемых параметров X и сами особи $s_i, i \in [1:|S|]$. Таким образом, пространство R_Φ является одномерным непрерывным,

пространство R_X – $|X|$ -мерным непрерывным, а пространство R_T является дискретным и имеет мощность, равную $|S|$.

Отношение (непосредственного) соседства особей $s_i, s_j \in S$ в пространстве R_α записываем в виде $s_i \leftrightarrow s_j; \alpha \in \{\Phi, X, T\}$. Аналогично, тот факт, что объединения особей G_i, G_j являются соседями в пространстве R_α , записываем как $G_i \leftrightarrow G_j; G_i, G_j \subseteq S$. Алгоритмы, определяющие пространства соседства особей и их объединений, обычно строят на основе метрик близости вида $\mu_{R_\alpha}(s_i, s_j), \mu_{R_\alpha}(G_i, G_j)$ соответственно.

По общему правилу различаем статические и динамические пространства соседства. В первом случае указанные метрики не изменяются в процессе эволюции популяции, а во втором случае – программно или адаптивно изменяются с ростом номера поколения.

3.1 Пространства соседства особей

Общее определение пространства соседства особей имеет вид

$$Space_{R_\alpha}(Arg, P) = Space_{R_\alpha} \Big| A^{Space.R_\alpha}(Arg, P), \alpha \in \{\Phi, X, T\}.$$

Алгоритм $A^{Space.R_\alpha}$, построенный на основе метрики близости $\mu_{R_\alpha}(s_i, s_j)$, определяет выражение

$$s_i \leftrightarrow s_j \Big| \mu_{R_\alpha}(s_i, s_j) \leq b, i, j \in [1:|S|], i \neq j,$$

где: b – свободный параметр.

Статические пространства соседства

Фитнесс-пространство $R_\Phi = R^1$. Простейшая метрика близости $\mu_{R_\Phi}(s_i, s_j)$ особей s_i, s_j в этом пространстве имеет вид

$$\mu_{R_\Phi}(s_i, s_j) = \text{abs}(\varphi(X_i) - \varphi(X_j)), i, j \in [1:|S|], i \neq j.$$

Вместо функции $\varphi(X)$ в определении метрики μ_{R_Φ} может быть использована аддитивная свертка этой функции с одной или несколькими вспомогательными функциями.

Примером метрики $\mu_{R_\Phi}(s_i, s_j)$, определенной более сложно, является метрика, используемая алгоритмом динамических сеток VMO (Variable Mesh Optimization) [14]:

$$\mu_{R_\Phi}(s_i, s_j) = \frac{1}{1 + |\varphi(X_i) - \varphi(X_j)|} \in [0; 1].$$

Легко видеть, что если $\varphi(X_i) \ll \varphi(X_j)$, то $\mu_{R_\varphi}(X_i, X_j) \approx 0$, и наоборот, если $\varphi(X_i) \approx \varphi(X_j)$, то $\mu_{R_\varphi}(X_i, X_j) \approx 1$.

Пространство поиска $R_X = R^{|X|}$. В качестве метрики близости $\mu_{R_X}(s_i, s_j)$ особей s_i, s_j в этом пространстве чаще всего используют евклидово расстояние $\|\cdot\|_E$ между особями:

$$\mu_{R_X}(s_i, s_j) = \mu_{R_X}(X_i, X_j) = \|X_i, X_j\|_E, i, j \in [1:|S|], i \neq j.$$

Известно значительное число других функций, которые могут быть использованы в качестве метрики близости $\mu_{R_X}(s_i, s_j)$: манхеттоновское расстояние $\|X_i, X_j\|_M^1$, аналогичное расстояние $\|X_i, X_j\|_M^{|X|}$ и другие [1].

Приведем в качестве примера алгоритм поиска системой зарядов CSA (Gravitational Search Algorithm) [15]. Расстояние $r_{i,j} = \mu_{R_X}(s_i, s_j)$ между частицами s_i, s_j в этом алгоритме определяет выражение

$$r_{i,j} = \frac{\|X_i - X_j\|_E}{\left\| \frac{X_i - X_j}{2} - X_{ib} \right\|_E + \varepsilon}, i, j \in [1:|S|], i \neq j,$$

где: положительная малая константа ε (свободный параметр) введена для того, чтобы избежать сингулярности; X_{ib} – текущее положение лучшей особи популяции.

Топологическое пространство R_T определяет топология соседства особей (*neighborhood topology*), которую задает неориентированный граф соседства $NG(S)$. Вершины этого графа соответствуют особям популяции, а ребра связывают непосредственных соседей. В алгоритме роя частиц PSO, например, используют топологии соседства клика, кольцо, двумерный тор, кластерная топология [1].

Метрику близости $\mu_{R_T}(s_i, s_j)$ особей $s_i, s_j, i, j \in [1:|S|], i \neq j$ в пространстве R_T определяют как расстояние между соответствующими вершинами графа $NG(S)$, то есть как минимальное число ребер, соединяющих вершины s_i, s_j этого графа. Поэтому максимальное значение этой метрики для данной топологии соседства равно, очевидно, диаметру графа $NG(S)$.

Динамические пространства соседства

По общему правилу, «динамичность» каждого из пространств соседства $R_\alpha, \alpha \in \{\Phi, X, T\}$ может быть обусловлена эволюцией метрики μ_{R_α} вследствие наличия в

функции $\mu_{R_\alpha}(\cdot)$ свободных динамических параметров или вследствие прямой зависимости этой функции от номера итерации t .

Пусть, например, в пространстве R_α определены статические метрики соседства $\mu_{R_\alpha}^1(s_i, s_j)$, $\mu_{R_\alpha}^2(s_i, s_j)$. Тогда динамическое неадаптивное пространство соседства можно определить с помощью динамической метрики вида

$$\mu_{R_\alpha}(s_i, s_j) = \begin{cases} \mu_{R_\alpha}^1(s_i, s_j), & t \leq t_{adapt}, \\ \mu_{R_\alpha}^2(s_i, s_j), & \text{иначе,} \end{cases} \quad i, j \in [1:|S|], \quad i \neq j.$$

Здесь: $t_{adapt} \in (0; \hat{t})$ – номер итерации, на которой изменяется пространство соседства (свободный параметр).

Динамическое адаптивное пространство соседства можно определить использованием какой-либо из характеристик популяции, например, с помощью метрики ее разнообразия $\mu_{div}(S)$, которой может служить текущий диаметр $d(S)$ популяции в пространстве R_X . С использованием метрик $\mu_{R_\alpha}^1(s_i, s_j)$, $\mu_{R_\alpha}^2(s_i, s_j)$ и метрики $\mu_{div}(S)$ это пространство определяет динамическая адаптивная метрика вида

$$\mu_{R_\alpha}(s_i, s_j) = \begin{cases} \mu_{R_\alpha}^1(s_i, s_j), & \mu_{div}(S) \leq d', \\ \mu_{R_\alpha}^2(s_i, s_j), & \text{иначе,} \end{cases} \quad i, j \in [1:|S|], \quad i \neq j,$$

где: $d' \in P$.

Топологическое пространство R_T становится динамическим при использовании программно или адаптивно изменяющейся топологии соседства [1].

3.2 Пространства соседства объединений особей

Метрики $\mu_{R_\alpha}(G_i, G_j)$, в отличие от аналогичных метрик для пространств соседства особей, оперируют не с переменными состояниями и характеристиками особей, но с состояниями и характеристиками их объединений; $\alpha \in \{\Phi, X, T\}$, $G_i, G_j \in S$.

Статические пространства соседства объединений особей

Фитнесс-пространство R_Φ . В качестве метрики близости объединений особей $G_i, G_j \in S$ в этом пространстве может быть использована метрика

$$\mu_{R_\Phi}(G_i, G_j) = \text{abs}(\varphi(\bar{X}_i) - \varphi(\bar{X}_j)), \quad i, j \in [1:|G|], \quad i \neq j, \quad (8)$$

где: $\bar{X}_k = \frac{1}{|G_k|} \sum_{l=1}^{|G_k|} X_l$, $k = i, j$ – центры «тяжести» указанных объединений соответственно, с которыми (центрами тяжести), вообще говоря, не обязательно ассоциируются какие-либо особи этих объединений.

Как и в аналогичной метрике близости особей, вместо функции $\varphi(X)$ здесь может использоваться аддитивная свертка этой функции с одной или несколькими вспомогательными функциями. С содержательной точки зрения малое значение метрики (8) означает близость значений Ф-функции $\varphi(X)$ в центрах «тяжести» объединений особей G_i, G_j .

Для того чтобы избежать вычисления значений функции $\varphi(X)$ в точках $\bar{X}_i, \bar{X}_j$, вместо этих точек могут быть использованы положения $\tilde{X}_i, \tilde{X}_j$ ближайших к ним особей соответствующих объединений, то есть можно принять

$$\tilde{X}_i = X_{i_k} = \min_l \|X_l - \bar{X}_i\|, \quad \tilde{X}_j = X_{j_k} = \min_l \|X_l - \bar{X}_j\|, \quad l \in [1:|G_j|],$$

где: $\|\cdot\|$ – некоторая векторная норма пространства R_X .

Другим примером метрики $\mu_{R_\varphi}(G_i, G_j)$ может служить функция

$$\mu_{R_\varphi}(G_i, G_j) = \text{abs}(\bar{\varphi}(G_i) - \bar{\varphi}(G_j)), \quad i, j \in [1:|G|], \quad i \neq j,$$

где: $\bar{\varphi}(G_k) = \frac{1}{|G_k|} \sum_{l=1}^{|G_k|} \varphi(X_l)$ – среднее по объединению G_k значение Ф-функции; $k = i, j$.

Пространство поиска R_X . Метрика близости $\mu_{R_X}(G_i, G_j)$ объединений особей G_i, G_j в пространстве R_X может быть построена на основе евклидова расстояния между центрами «тяжести» этих объединений:

$$\mu_{R_X}(G_i, G_j) = \|\bar{X}_i, \bar{X}_j\|_E, \quad i, j \in [1:|G|], \quad i \neq j.$$

Можно предложить ряд аналогичных метрик, основанных на других нормах векторов в пространстве $R^{|X|}$. Подобно тому, как было показано выше, вместо точек $\bar{X}_i, \bar{X}_j$ могут быть использованы положения X_{i_k}, X_{j_k} ближайших к ним особей соответствующих объединений.

Топологическое пространство R_T . По аналогии с таковым пространством особей, пространство объединений особей определяет их топология соседства, которой соответствует граф соседства $NG(G)$. Мерой близости $\mu_{R_T}(G_i, G_j)$ объединений G_i, G_j , $i, j \in [1:|G|], i \neq j$ в указанном пространстве является минимальное число ребер указанного графа, соединяющих его вершины G_i, G_j .

Динамические пространства соседства объединений особей

Как и в случае динамических пространств соседства особей, «динамичность» пространства соседства объединений особей может быть обеспечена за счет эволюций соответствующей функции $\mu_{R\alpha}$ в процессе поиска вследствие наличия в ней свободных динамических параметров или вследствие прямой зависимости этой функции от номера итерации t .

4. Особи

Как мы отмечали в п. 1, особью $s_i, i \in [1:|S|]$ называется приближение к решению задачи (кандидат в решение). Состояние особи s_i в общем случае определяют векторы ее положения X_i , скорости ΔX_i и ускорения $\Delta^2 X_i$, так что, вообще говоря, вектор состояния этой особи равен

$$V_i = X_i \cup \Delta X_i \cup \Delta^2 X_i, i \in [1:|S|], |V_i| = 3|X|.$$

Области допустимых значений составляющих $\Delta X_i, \Delta^2 X_i$ по аналогии с множеством Ω_X обозначаем

$$\Omega_{\Delta X} = \{\Delta X \mid \Delta X^- \leq \Delta X \leq \Delta X^+\}, \Omega_{\Delta^2 X} = \{\Delta^2 X \mid \Delta^2 X^- \leq \Delta^2 X \leq \Delta^2 X^+\}.$$

Основной характеристикой особи s_i , определяющей ее «качество», является значение Ф-функции $\varphi_i = \varphi(X_i)$. Кроме того, в число характеристик особи могут входить следующие величины: текущая «успешность» особи (лучшее достигнутое за время $[0; t]$ значение Ф-функции, среднее этих значений, средне квадратичное отклонение и т. д.); признак того, являлся ли на итерации t данная особь глобально лучшей (худшей); аналогичный признак лучшей (худшей) особи в ее некоторой текущей окрестности (п.6); «вес» особи (например, в алгоритме FSS [11] или в алгоритме GSA [15]) и т. д.

Выделяем характеристику $type(s_i)$ особи s_i , называемую *типом*, которая определяет особенности эволюции этой особи. Тип особи может определять также ее принадлежность к тому или иному объединению особей (п.5). Например, в алгоритме колонии пчел BA (Bees Algorithm) [16] различают рабочих пчел (*worker bees*), пчел-разведчиков (*scout bees*) и пчел-наблюдателей (*onlooker bees*); в муравьином алгоритме CACO [9] выделяют муравьев, предназначенных для глобального и локального поиска. И так далее.

Различаем характеристики особи, основанные на ее текущих состояниях и на ее следе [1].

Характеристики, основанные на текущих состояниях особи

Значения данных характеристик могут быть определены на основе «самооценки» особи и на основе оценки данной особи другой особью или особями.

Основной самооценкой особи s_i является ее приспособленность $\varphi_i = \varphi(X_i)$. Относительное качество особи определяет ее такая характеристика, как *ранг*, который равен ее номеру в списке всех особей популяции, отсортированных в порядке убывания их текущих

приспособленностей. Особь с большим рангом является более успешной по сравнению с особью, имеющей меньший ранг (напомним, что имеется в виду минимизации Ф-функции).

Оценку данной особи другой особью определяется *привлекательность* (attractiveness). В пространстве $R_\alpha \in \{R_\phi, R_X, R_T\}$ алгоритм $A^{Attact.s}$, вычисляющий привлекательность $\beta(s_i, s_j) = \beta_{i,j}$ особи s_j для особи s_i , в общем случае определяет функция вида

$$\beta_{i,j} = \beta(\mu_{R_\alpha}(s_i, s_j)), i, j \in [1:|S|], i \neq j,$$

где: $\mu_{R_\alpha}(s_i, s_j)$ – некоторая метрика расстояния между особями s_i, s_j в пространстве R_α . В алгоритме светлячков FA (Firefly Algorithm) [17], например, в качестве привлекательности используют функции

$$\beta(r_{i,j}) = b_0 \exp(-b_1 r_{i,j}^{b_2}), \beta(r_{i,j}) = \frac{b_0}{1 + b_1 r_{i,j}^2}, b_2 > 1,$$

$$\text{где: } r_{i,j} = \mu_{R_X}(s_i, s_j) = \mu_{R_X}(X_i, X_j) = \|X_j - X_i\|_E.$$

Характеристики, основанные на следе особи также разделяем на «самооценочные» и внешние.

В качестве характеристик первого сорта могут быть использованы минимальные, максимальные, средне взвешенные и среднеквадратичные отклонения указанных выше самооценок особи s_i во всех точках ее следа. В П-алгоритмах часто используют характеристику особи вида

$$\chi_i = \chi_i(t) = \sum_{\tau=t-\Delta t_i}^t b_\tau \varphi_i(\tau); i \in [1:|S|], \quad (9)$$

которая называется текущим *здоровьем* (health) особи s_i . Можно сказать, что эта величина представляет собой самооценку успешности особи в процессе ее эволюции в течение Δt_i итераций. Таким образом, например, определяют состояние здоровья бактерии в бактериальном алгоритме [5]. В муравьином алгоритме COAC в качестве весовых коэффициентов b_τ логично использовать количество феромона $\theta_i(t, \tau)$ [6].

Вместо величин $\varphi_i(\tau)$ в формуле (9) могут использоваться соответствующие нормализованные приращения приспособленности особи, так что эта формула приобретает вид

$$\chi_i = \chi_i(t) = \sum_{\tau=t-\Delta t_i+1}^t b_\tau \Delta \tilde{\varphi}_i(\tau), i \in [1:|S|],$$

где

$$\Delta \tilde{\varphi}_i(\tau) = \frac{\varphi_i(\tau) - \varphi_i(\tau-1)}{\max(\varphi_i(\tau), \varphi_i(\tau-1))}. \quad (10)$$

На основе подобной оценки здоровья особи s_i вычисляют, например, ее *вес* в операторе кормления (feeding) алгоритма FSS [11].

Нормализация разности приспособленностей $\varphi_i(\tau) - \varphi_i(\tau - 1)$ в формуле (10) может быть выполнена и иными способами. Например, в эволюционном операторе *инстинктивно-коллективного плавания* того же алгоритма FSS используют нормализацию вида

$$\Delta \tilde{\varphi}_i(\tau) = \frac{\varphi_i(\tau) - \varphi_i(\tau - 1)}{\sum_j (\varphi_j(\tau) - \varphi_j(\tau - 1))}, \quad j \in [1:|S|]$$

В муравьином алгоритме COAC [9] здоровье особи (там эта величина называется *слабостью* (weakness) или *возрастом* (age) особи) вычисляют по правилу

$$\chi'_i = \begin{cases} \chi_i + \Delta\chi, & \varphi'_i > \varphi_i, \\ \chi_i - \Delta\chi, & \text{иначе.} \end{cases}$$

Другими словами, здоровье особи повышают на величину $\Delta\chi$ (свободный параметр алгоритма), если ее новое положение X'_i увеличивает приспособленность особи. В противном случае, здоровье уменьшают на ту же величину. По общим принципам, величина $\Delta\chi$ может меняться с ростом номера итерации и может быть различной для разных особей (например, особей разного типа). Положительные и отрицательные приращения здоровья особи могут быть различными по абсолютной величине.

Характеристики, основанные на внешней оценке особи S_i , могут быть построены по аналогичным схемам. Например, возможна оценка здоровья данной особи с точки зрения особи S_j вида

$$\chi_{i,j} = \chi_{i,j}(t) = \sum_{\tau=t-\Delta t_i}^t b_\tau \beta_{i,j}(\tau), \quad i \in [1:|S|].$$

5. Популяция и объединения особей

Популяция представляет собой совокупность (объединение) $|S|$ особей $S = \{s_i, i \in [1:|S|]\}$. Некоторые П-алгоритмы используют ограничения снизу и сверху $|S|^{\min}, |S|^{\max}$ на размер популяции, которые, как и величина $|S|$, являются свободными параметрами П-алгоритма. Состояние $V(S)$ популяции S определяет набор состояний ее особей: $V(S) = \{V(s_i), i \in [1:|S|]\}$.

Популяция (субпопуляция) может включать в себя некоторое число объединений особей. С другой стороны, особь можно считать частным случаем объединения особей (когда мощность этого объединения равна единице).

Объединение особей, определенное в пространстве R_α , задает выражение вида

$$Union_{R_\alpha}(S, P) = Union_{R_\alpha} \left| A^{Union.R_\alpha}(S, P) \subseteq S, \alpha \in \{R_\phi, R_X, R_T\} \right. \quad (11)$$

Здесь: $A^{Union.R_\alpha}$ – детерминированный или стохастический алгоритм, задающий правила вхождения особей в данное объединение. Максимальное расстояние между особями,

образующими объединение $Union_{R_\alpha}$, в используемой метрике μ_{R_α} пространства R_α называется *диаметром* этого объединения, а половина этой величины – *радиусом* объединения.

Отметим, что определение (11) не требует, чтобы каждая данная особь входила только в одно объединение, так что объединения могут пересекаться. Например, в клубном алгоритме роя частиц (*Clubs-Based PSO*) [1] каждая особь популяции S может входить одновременно в несколько клубов.

На этапе инициализации популяции (с помощью оператора вида $O^{init, Union, R_\alpha}(S, P)$) осуществляется инициализация используемых П-алгоритмом объединений особей, включая определение значений их свободных параметров (вообще говоря, разных для разных объединений). В некоторых П-алгоритмах особи, входящие в данное объединение, называются *соседями* (*neighbors*).

Различаем статические и динамические объединения особей. Статическое объединение не меняет состав особей, включенных в него, на протяжении всей эволюции популяции, то есть на всех итерациях $t \in [t_0 : \hat{t}]$. Напротив, состав динамического объединения меняется в процессе поиска. Полагаем, что объединение является динамическим только в том случае, если его состав меняется вследствие изменения в процессе итераций алгоритма $A^{Union, R_\alpha}(S, P)$.

Объединение особей $Union_{R_\alpha}$ может быть построено на основе окрестности некоторой особи $s_i \in S$ (п. 6) по следующему правилу:

$$Union_{R_\alpha}(S, P) = \{s_{i_j} \in S \mid s_{i_j} \in Area_{R_\alpha}(s_i, P)\} \subseteq S, \alpha \in \{R_\phi, R_X, R_T\}.$$

Обычно, алгоритм A^{Union, R_α} , осуществляющий отбор особей в объединение $Union_{R_\alpha}(S, P)$, основан на близости этих особей в некоторой метрике μ_{R_α} пространства $R_\alpha \in \{R_\phi, R_X, R_T\}$. Так, для любого из пространств R_ϕ, R_X, R_T объединение особей можно определить правилом

$$\{s_{i_1}, s_{i_2}\} \in Union_{R_\alpha}(S, d_1, d_2) \mid \mu_{R_\alpha}(s_{i_1}, s_{i_2}) \in [d_1; d_2], s_{i_1}, s_{i_2} \in S, \quad (12)$$

где: d_1, d_2 – свободные параметры объединения; $d_1 < d_2$. При этом алгоритм $A^{Union, R_\alpha}(S, P)$ определяется неравенством $\mu_{R_\alpha}(s_{i_1}, s_{i_2}) \in [d_1, d_2]$, так что $P = \{d_1, d_2\}$.

Вообще говоря, для формирования объединения $Union_{R_\alpha}(S, P)$, $R_\alpha \in \{R_F, R_X, R_T\}$ могут быть использованы любые из алгоритмов селекции [1].

Объединение особей может быть определено как пересечение объединений, определенных в разных пространствах. Пусть, например, тем или иным образом определены объединения $Union_{R_\phi}(S, P^{R_\phi})$, $Union_{R_X}(S, P^{R_X})$, $Union_{R_T}(S, P^{R_T})$. Тогда мыслимо объединение вида

$$Union_{R_\phi \wedge R_X \wedge R_T}(S, P) = Union_{R_\phi}(S, P^{R_F}) \cap Union_{R_X}(S, P^{R_X}) \cap Union_{R_T}(S, P^{R_T}).$$

Объединения особей могут быть построены с помощью *комбинированных* алгоритмов $A^{Union.R_\alpha}$, которые кроме метрики близости μ_{R_α} используют и другие условия. Например, в алгоритме оптимизации роем светлячков GSO (Glowworm swarm optimization) [18] светлячок S_j считается соседом светлячка S_i , $i, j \in [1:|S|]$, $i \neq j$ при выполнении следующих двух условий: евклидово расстояние между этими светлячками не превышает текущий радиус окрестности r_i ; текущий уровень l_j светимости светлячка S_j превышает этот же уровень l_i светлячка S_i , то есть $l_j > l_i$.

Создать объединение особей в пространствах R_ϕ , R_X , R_T можно путем кластеризации особей на основе их близости в этих пространствах. Примером создания объединений особей по такой схеме является алгоритм роя частиц со стереотипированием (объединение особей здесь называется *кластером*) [1].

В качестве характеристики объединения особей могут быть использованы характеристики, построенные на основе соответствующих характеристик особи. Наряду с этим у объединения особей могут быть определены характеристики, не вытекающие из характеристик особей, входящих в данное объединение.

Статические объединения

Фитнесс-пространство R_ϕ . Кроме правил вида (12), статическое объединение особей в пространстве R_ϕ можно определить с помощью условия

$$s_{i_j} \in Union_{R_\phi}(S, b_1, b_2) \mid b_1 \leq \varphi(s_{i_j}) \leq b_2, s_{i_j} \in S.$$

Здесь: b_1, b_2 – свободные вещественные параметры объединения, удовлетворяющие условию $\varphi^{\min} \leq b_1 < b_2 \leq \varphi^{\max}$; $\varphi^{\min}, \varphi^{\max}$ – текущие минимальное и максимальное значения приспособленности особей популяции S . Алгоритм $A^{Union.R_\alpha}(S, P)$ в данном случае определяет неравенство $b_1 \leq \varphi(s_{i_j}) \leq b_2$; $P = \{d_1, d_2\}$.

Пространство поиска R_X . Формула (12) в этом пространстве обычно имеет вид

$$\{s_{i_1}, s_{i_2}\} \in Union_{R_F}(S, d) \mid \|X_{i_1}, X_{i_2}\|_E \leq d, s_{i_1}, s_{i_2} \in S, \quad (13)$$

где: d – свободный вещественный параметр. Кроме евклидовой метрики возможно применение любых иных метрик расстояния в этом пространстве.

Примером использования объединений в пространства R_X является упомянутый выше алгоритм роя частиц со стереотипированием, в котором используется евклидова метрика близости особей.

Топологическое пространство R_T . Поскольку метрика $\mu_{R_T}(s_i, s_j)$ близости особей s_i, s_j в пространстве R_T имеет смысл расстояния между соответствующими вершинами

графа соседства $NG(S)$, объединение особей в этом пространстве представляет собой подграф графа $NG(S)$, который определяет формула вида

$$\{s_{i_1}, s_{i_2}\} \in Union_{R_T}(S, d) \mid \mu_{R_T}(s_{i_1}, s_{i_2}) \leq d, s_{i_1}, s_{i_2} \in S, \quad (14)$$

где: d – свободный параметр – радиус объединения $Union_{R_T}(S, d)$.

Динамические объединения

По общему правилу «динамичность» объединения особей $Union_{R_\alpha}(S, P)$ может быть обусловлена эволюцией алгоритма $A^{Union_{R_\alpha}}(S, P)$ в процессе поиска вследствие наличия в этом алгоритме свободных динамических параметров, а также следствием прямой зависимости алгоритма от номера итерации t . В процессе итераций П-алгоритма особи популяции S могут входить в и выходить из динамических объединений. Динамическое объединение особей в пространстве R_T использует, например, тасующий алгоритм лягушек *SFL* (Shuffled Frog-Leaping Algorithm), в котором объединения называются *мемплексами* [19]. «Динамичность» объединений в этом алгоритме обусловлена использованием процедуры тасования (*shuffling procedure*) особей.

Характеристики объединений, основанные на характеристиках особей

Характеристики данного класса по аналогии с п. 4 разделяем на характеристики, использующие «самооценки» и внешние оценки.

Характеристики первого сорта представляют собой, например, минимальные, максимальные, средне взвешенные и среднеквадратичные отклонения указанных самооценок особей данного объединения. Например, в качестве характеристики объединения особей G_i может быть использована величина

$$\varphi_i^{**} = \max_j \varphi(X_{i_j}), j \in [1:|G_i|]$$

– лучшее значение Ф-функции, достигнутое особями объединения G_i (алгоритм $A^{\varphi.best.G}$).

Важные характеристики объединения G_i могут быть получены на основе следов особей этого объединения. Так на этой основе может быть вычислена текущая скорость улучшения качества объединения G_i

$$\Delta\varphi_i^{**} = \varphi_i^{**}(t) - \varphi_i^{**}(t-1)$$

или аналогичная величина текущего ускорения

$$\Delta\varphi_i^{**}(\Delta t) = \frac{1}{\Delta t}(\varphi_i^{**}(t) - \varphi_i^{**}(t - \Delta t)); \Delta t \geq 1.$$

Характеристики, основанные на внешних оценках особей данного объединения G_i , могут представлять собой оценки одной особи $s_{j_k} \in G_j$ и оценки всех особей объединения G_j ; $j \neq i$. И в том и в другом случае в качестве характеристик объединения особей могут быть использованы минимальные, максимальные, средние взвешенные и среднеквадратичные отклонения указанных оценок. Пусть, например, известны привлекательности β_{i_l, j_k} , $i_l \in [1:|G_i|]$ всех особей объединения G_i для особи $s_{j_k} \in G_j$. Тогда в качестве искомой характеристики объединения G_i может быть использована средняя привлекательность особей этого объединения для особи $s_{j_k} \in G_j$, равная

$$\bar{\beta}_{i, j_k} = \frac{1}{|G_i|} \sum_{i_l} \beta_{i_l, j_k}, \quad i_l \in [1:|G_i|].$$

Характеристики объединения, не основанные на характеристиках особей

Примером характеристик такого сорта, которая широко используется для оценки качества популяции (субпопуляции) и управления ею, является характеристика *разнообразия* популяции (*population variance*). Меры разнообразия популяции S могут строиться на основе оценки дисперсии соответствующих решений по каждому из $|X|$ измерений пространства поиска:

$$v_j(S) = \frac{\sum_{i=1}^{|S|} (x_{i,j} - \bar{x}_j)^2}{|S|}, \quad \bar{x}_j = \frac{\sum_{i=1}^{|S|} x_{i,j}}{|S|}, \quad j \in [1:|X|].$$

В муравьином алгоритме СОАС используются следующие характеристики особей, принадлежащих данной окрестности (*региону*): вектор координат наиболее активной феромонной точки региона; соответствующее количество феромона; наилучшее значение фитнес-функции активных феромонных точек региона; ранг региона; общее число посещений муравьями данной окрестности [6].

6. Окрестности особи

Окрестностью особи $s_i \in S$ в пространстве R_α , $\alpha \in \{\Phi, X, T\}$ (R_α -окрестностью этой особи), называем область этого пространства, элементы которой в некотором смысле близки к положению особи s_i в этом пространстве. В П-алгоритмах обычно используют окрестности в пространстве R_X . Мы, однако, рассматриваем также окрестности особи в пространствах R_Φ , R_T .

Аналогично определяем окрестность объединения особей $G_i \subseteq S$ в пространстве R_α , $\alpha \in \{\Phi, X, T\}$ (R_α -окрестность этого объединения).

П-алгоритмы используют не единичные окрестности и объединения особей, но наборы этих сущностей, каждый из которых определяется правилами его создания (инициализации) и правилами эволюции (если эта сущность является динамической). Мощности этих наборов

являются обычно свободными параметрами П-алгоритма. Используемые П-алгоритмом окрестности могут пересекаться.

Окрестность особи S_i в пространстве R_α определяем в виде

$$Area_{R_\alpha}(s_i, P) = Area_{R_\alpha} \mid A^{Area.R_\alpha}(s_i, P) \subseteq R_\alpha, \alpha \in \{\Phi, X, T\},$$

где: $A^{Area.R_\alpha}$ – алгоритм, определяющий окрестность; P – набор свободных параметров, используемых алгоритмом $A^{Area.R_\alpha}$. Вектор параметров P обязательно включает в себя параметр или параметры, которые определяют размеры окрестности. Называем эти параметры *радиусами* окрестности. Особь S_i называется *центральной* особью окрестности $Area_{R_\alpha}(s_i, P)$. В разных П-алгоритмах окрестность особи называют по-разному, например, в муравьином алгоритме COAC R_X -окрестность называется *регионом* поиска [6].

На этапе инициализации популяции осуществляется инициализация каждой из используемых П-алгоритмом окрестностей. Эта процедура включает в себя определение центральных особей окрестностей и, по общему правилу, определение значений свободных параметров окрестностей (вообще говоря, разных для разных окрестностей). Например, в том же муравьином алгоритме COAC координаты центров окрестностей являются случайными, равномерно распределенными в области поиска Ω_X .

По общим правилам, радиусы окрестностей и другие свободные параметры алгоритма $A^{Area.R_\alpha}$ могут быть детерминированными или случайными с заданными законами распределения, стационарными или динамическими (программными или адаптивными). Например, в том же алгоритме COAC эти радиусы есть случайные величины, равномерно распределенные в заданном интервале.

Если алгоритм $A^{Area.R_\alpha}$ является статическим, то соответствующую окрестность $Area_{R_\alpha}(s_i, R)$ называем *статической* окрестностью. В случае если алгоритм $A^{Area.R_\alpha}$ эволюционирует в процессе поиска, например, вследствие изменения значений его динамических свободных параметров, то окрестность $Area_{R_\alpha}(s_i, P)$ называется *динамической* окрестностью. В П-алгоритме COAC, например, одной из причин эволюции окрестностей является адаптивность радиусов этих окрестностей [6].

Статические окрестности особи

Обычно R_F, R_X, R_T – окрестности особи S_i определяют с помощью правила вида

$$Area_{R_\Phi}(s_i, r) = \{\varphi \in R^1 \mid \mu_{R_\Phi}(\varphi_i, \varphi) \leq r\}, \quad (15)$$

$$Area_{R_X}(s_i, r) = \{X \in R^{|X|} \mid \mu_{R_X}(X_i, X) \leq r\}, \quad (16)$$

$$Area_{R_T}(s_i, r) = \{s_j \in S \mid \mu_{R_T}(s_i, s_j) \leq r\}, \quad (17)$$

соответственно, где: μ_{R_α} – некоторая метрика близости в пространстве R_α ; r – радиус окрестности (свободный параметр). Алгоритму $Area_{R_\alpha}(s_i, P)$ здесь соответствует формула вида $\mu_{R_\alpha}(\cdot, \cdot) \leq r$.

Центрами окрестностей $Area_{R_\phi}(s_i, r)$, $Area_{R_X}(s_i, r)$, $Area_{R_T}(s_i, r)$ называют величину $\phi(X_i)$, точку X_i и особь s_i соответственно. Центры окрестностей $Area_{R_\phi}(s_i, r)$, $Area_{R_X}(s_i, r)$ могут не ассоциироваться с текущим положением особи. Так, в некоторых П-алгоритмах говорят об окрестности $Area_{R_X}(X, P)$ одной из точек $X \in \Omega_X$, например, некоторой точки следа особи [1].

Фитнесс-пространство R_ϕ . В качестве метрики μ_{R_ϕ} целесообразно использовать функцию вида

$$\mu_{R_\phi}(\phi_i, \phi) = \text{abs}(\phi_i - \phi). \quad (18)$$

В этом случае формула (15) определяет, очевидно, числовой отрезок в пространстве R^1 . Вместо Φ -функции $\phi(X)$ может быть использована Ц-функция либо ее какие-либо преобразования.

Пространство поиска R_X . Если метрика μ_{R_X} построена на основе евклидова расстояния между точками пространства R_X , то неравенство $\mu_{R_X}(X_i, X) \leq r$ определяет шар в этом пространстве с центром в точке X_i . Таким образом, например, определена окрестность в светлячковом П-алгоритме GSO [18].

Примером статической R_X -окрестность особи s_i является окрестность особи (делегата) s_j , $j \in [1:|S|]$ в алгоритме стохастического диффузионного поиска SDS, которую определяет выражение

$$Area_{R_X}(s_i, b) = d(X_i) = \{X \in \Omega_X \mid x_j^l \leq x_j \leq x_j^u, \quad j \in [1:|X|]\},$$

где:

$$x_j^l = \max((x_j - 0,5b(x_j^+ - x_j^-), x_j^-), \quad x_j^u = \min((x_j + 0,5b(x_j^+ - x_j^-), x_j^+),$$

b – свободный параметр [20].

Поскольку пространство R_X является многомерным, возможно определение R_X -окрестности особи s_i с помощью набора правил вида

$$(\mu_{R_X}^1(X_i, X) \leq r_1) \wedge \dots \wedge (\mu_{R_X}^{|X|}(X_i, X) \leq r_{|X|}),$$

где: $r_1, \dots, r_{|X|}$ – радиусы окрестности; $\mu_{R_X}^j(X_i, X) = \text{abs}(x_{i,j} - x_j)$, $j \in [1:|X|]$.

Легко видеть, что так определенная окрестность $\text{Area}_{R_X}(s_i, P)$ представляет собой параллелепипед

$$\{(x_{i,j} - r_j) \leq x_j \leq (x_{i,j} + r_j), j \in [1:|X|]\} \in R_X.$$

Таким образом, например, определена окрестность особи s_i в муравьином алгоритме СОАС [6].

Топологическое пространство R_T . Приведем следующий пример. Пусть пространство R_T определяет топология соседства типа кольцо (граф соседства которой $NG(S)$ представляет собой кольцо) [1] и задан радиус окрестности r , удовлетворяющий условию $2r \leq |S|$. Тогда, например, окрестность $\text{Area}_{R_T}(s_1, r)$ образуют особи $s_2, s_3, \dots, s_{r+1}; s_{|S|}, s_{|S|-1}, \dots, s_{|S|-r+1}$, то есть все особи, расстояние которых в графе $NG(S)$ от особи s_i не превышает r .

Динамические окрестности особи

Динамические окрестности особи порождают, прежде всего, их динамические радиусы. В светлячковом алгоритме GSO, к примеру, используются динамические R_X -окрестности особи, изменяющиеся вследствие изменения радиуса окрестности по правилу

$$r' = \min(b_0, \max(0, r + b_1(m - |Area(s_i)|))), s_i \in S,$$

где: b_0 – минимально допустимый радиус окрестности; m – желательное число соседей; $b_1 > 0$ – весовой коэффициент; $|Area(s_i)|$ – число особей во множестве соседей, определяемом окрестностью $Area(s_i)$ особи s_i [18].

Характеристики окрестности особи

В качестве характеристики окрестности особи могут быть, очевидно, использованы характеристики объединения особей, которое формирует данная окрестность (п.5). К характеристикам собственно окрестности особи можно отнести свободные параметры, которые определяют эту окрестность и, прежде всего, радиусы окрестности. Более содержательными являются характеристики, которые определяют форму окрестности, ее абсолютный и относительный объемы, асимметрию относительно центральной особи и т. д.

Если в фитнес-пространстве R_Φ используется метрика вида (15), то характеристикой окрестности $\text{Area}_{R_\Phi}(s_i, r)$ является величина r . Эта же величина может быть использована в качестве характеристики для метрики $\mu_{R_\Phi}(\varphi_i, \varphi) = \text{abs}(\varphi_i - \varphi)$ и любых иных метрик, основанных на каких-либо преобразованиях Φ -функции.

Более широкий набор характеристик можно предложить для окрестности особи S_i в пространстве поиска R_X . Так, при использовании любой из метрик μ_{R_X} , кроме радиуса или радиусов, характеристикой окрестности $Area_{R_X}(S_i, P)$ является ее объем. Если окрестность отлична от шара, то в качестве ее характеристики может быть использовано отношение максимального и минимального размеров этой окрестности и т. д.

В топологическом пространстве R_T характеристикой окрестности $Area_{R_T}(S_i, r)$ может служить число особей популяции S , принадлежащих этой окрестности. Заметим, что это число определяется не только радиусом окрестности r , но и используемой топологией соседства особей.

Заключение

В силу ограниченности размера статьи, за ее пределами остались такие сущности, как след особи, каналы связей и «тоннели» между особями и их объединениями, события в популяции и другие. По той же причине в работе не представлены сущности мультипопуляционных алгоритмов, являющиеся специфичными для этих алгоритмов.

Обобщение и классификация П-алгоритмов требует еще рассмотрения их базовых операций, поисковых операторов, моделей особей, методов гибридизации, адаптации и метаоптимизации, а также некоторых других выразительных средств, используемых при синтезе П-алгоритмов.

ЛИТЕРАТУРА

1. Карпенко А. П. Современные алгоритмы поисковой оптимизации. Алгоритмы вдохновленные природой. М.: Издательство МГТУ им. Н. Э. Баумана, 2014. 446 с.
2. Bo Xing, Wen-Jing Gao. Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms. – Springer International Publishing Switzerland, 2014. – 451 P.
3. Evolutionary Computation. 1. Basic Algorithms and Operators / Edited by Thomas Bäck, David B. Fogel and Zbigniew Michalevich. – Institute of Physics Publishing, Bristol and Philadelphia, 2004. – 339 P.
4. Evolutionary Computation. 2. Advanced Algorithms and Operators / Edited by Thomas Bäck, David B. Fogel and Zbigniew Michalevich. – Institute of Physics Publishing, Bristol and Philadelphia, 2004. – 339 P.
5. Swagatam Das, et al. Bacterial Foraging Optimization Algorithm: Theoretical Foundations, Analysis, and Applications / Foundations of Computational Intelligence. Global Optimization. – Springer Berlin Heidelberg, 2009, Vol. 3. – P. 23-55.
6. Hu X. M., Zhang J., Li Y. Orthogonal methods based ant colony search for solving continuous optimization problems // Journal of Computer Science and Technology, 2008, No. 23(1). – P. 2-18.
7. Pedersen M. E. H., Chipperfield A. J. Simplifying Particle Swarm Optimization / Applied Soft Computing, 2010, Vol. 10, Issue 2, March 2010. – P. 618-628.

8. Geem Z. W. (Ed.). Music-Inspired Harmony Search Algorithm: Theory and Applications, Studies in Computational Intelligence. – Springer, 2009. – 206 P.
9. J. Dreco, P. Siarry. Continuous interacting ant colony algorithm based on dense heterarchy // Future Generation Computer Systems, 2004, No. 20, P. 841-856.
10. Yang, X.-S. Bat algorithm: literature review and applications. International Journal of Bio-Inspired Computation, 2013, No. 5. – P. 141-149.
11. Carmelo J. A. Bastos Filho, et al. Fish School Search / Nature-Inspired Algorithms for Optimisation. – Springer-Verlag Berlin Heidelberg, 2009. – P. 261-277.
12. Курейчик В. М. Генетические алгоритмы и их применение. Научное издание. – Издательство ТРТУ, 2002. – 242 С.
13. Скобцов Ю. А., Федоров Е. Е. Метаэвристики: монография. – Донецк: Изд-во «Ноулидж» (Донецкое отделение), 2013. – 426 с.
14. Amilkar Puris, et al. Variable mesh optimization for continuous optimization problems // Soft Computing, 2012, Vol. 16, Issue 3. – P. 511-525.
15. Rashedi E., Nezamabadi H., Saryazdi S. GSA: A Gravitational Search Algorithm / Information Sciences, 2009, Vol. 179, Issue 13. – P. 2232-2248.
16. Pham D. T., Castellani M. A comparative study of the bees algorithm as a tool for function optimization // Cogent Engineering, 2015, Vol. 2(1). – P. 1-28.
17. Xin-She Yang. Firefly Algorithms for Multimodal Optimization / Lecture Notes in Computer Science. Stochastic Algorithms: Foundations and Applications. – Springer-Verlag Berlin Heidelberg, 2009. – P. 169-178.
18. Krishnanand K. N., Ghose D. Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions // Swarm Intelligence, 2009, Vol. 3, Issue 2. – P. 87-124.
19. Bozorg-Haddad O., Solgi M., Loáiciga H. A. Shuffled Frog-Leaping Algorithm / Meta-Heuristic and Evolutionary Algorithms for Engineering Optimization, Wiley Online Library, 2017, DOI: 10.1002/9781119387053.ch11.
20. Bishop J. M., Torr P. The Stochastic Search Network / Neural Networks for Images, Speech and Natural Language. – New York, Chapman & Hall. 1992. – P. 370-387.

Karpenk Anatoly Pavlovich

Bauman Moscow state technical university, Russia, Moscow

E-mail: apkarpenko@mail.ru

The main essentials of population-based algorithms of global optimization. Systematization experience

Abstract. In population algorithms of global optimization, the number of candidates to a solution (individuals) is greater than one. At each of the iterations of the population algorithm, either all the individuals move, excluding, perhaps, only some of them (for example, "best" individuals) or a certain number of individuals exceeding one. In different publications, population algorithms (P-algorithms) of global optimization are called behavioral, intellectual, meta-heuristic, inspired nature, swarms, multiagents, etc.

We consider P-algorithms for global continuous optimization. In connection with the large and constantly increasing number of such algorithms, the problem of systematizing their expressive means is topical. The article analyzes one of the components of this problem – the problem of ordering the essentials of P-algorithms.

We represent the formulation of the problem of global continuous optimization and the general scheme of P-algorithms for its solution. We distinguish the following basic essentials: free parameters, the neighborhood space, the individual, the population and association of individuals, the area of the individual, and others. We reveal the essence of these essentials, present their characteristics, give variants of use in various P-algorithms.

Keywords: global optimization; population algorithm; meta-heuristic algorithm; individual; population; neighborhood space