

ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ

1. Введение

Подавляющее большинство инженерных решений принимается на основе максимизации функции многих переменных $\max_x D(x_1, x_2, \dots, x_n)$. В задаче обучения нейронных сетей роль переменных x_1, x_2, \dots, x_n , образующих вектор x , играют синаптические коэффициенты и смещения нейронов $w_{ij}, b_i, i, j \in I$, где I определяет область изменения индексов. Возможно также включение в состав оптимизируемых переменных параметров активационных характеристик нейронов (например, параметр крутизны сигмоиды или параметр “размытия” гауссианы). Максимизируемая функция D в этой задаче отражает соответствие выхода нейронной сети желаемому преобразованию входных данных. В частности, можно максимизировать величину “обратной” квадратичной ошибки $D = -\varepsilon^2 = -\sum_{l,p} (\sigma_l^p - s_l^p)^2$, где σ_l^p, s_l^p - желаемый и реальный выходы l -го нейрона. Учитывая большое число обучаемых параметров нейронной сети, задача ее оптимизации имеет высокую размерность с непредсказуемо сложной функцией адаптивного рельефа $D(w_{ij}, b_i, i, j \in I)$.

Классические подходы к решению поставленной оптимизационной задачи можно разделить на 4 группы:

- различные модификации градиентных методов спуска,
- перебор на дискретной сетке в пространстве оптимизируемых параметров,
- случайный поиск,
- гибридные методы.

Градиентные методы характеризуются сходимостью к локальному (не глобальному) экстремуму, сложностью подбора и настройки шага поиска, необходимостью вычисления направления спуска (часто трудоемкого с вычислительной точки зрения).

Метод перебора практически может быть применен для малых размеров поисковой решетки. Обычно применяют не полный перебор, а специальные схемы перебора, например, динамическое программирование, хотя и в этом случае возникает проблема “проклятия размерности”.

Случайный поиск, если в логику поиска не включена разумная целенаправленная стратегия, мало эффективен, характеризуется медленной сходимостью. Удобство этого метода состоит в том, что в процессе оптимизации выполняется расчет только значений функции D и не требуется оценивание ее производных.

Гибридные методы комбинируют разные перечисленные выше принципы организации поиска, которые выбираются в соответствии со свойствами оптимизируемой функции.

Джон Холланд (John Holland) вместе с коллективом сотрудников мичиганского университета в начале 70-х годов поставил задачу использования в процессе поиска $\max_x D(x_1, x_2, \dots, x_n)$ тех же механизмов, что и в живой природе, то есть механизмов, которые направлены на выживание наиболее приспособленных организмов и их совершенствование в процессе эволюции. Основой для исследования избран *направленный случайный поиск*, который может быть реализован на *параллельно работающих вычислителях*. Первая

книга, в которой изложены идеи метода, математические модели и область приложений, вышла в 1975 году (John Holland, “Adaptation in Natural and Artificial Systems”). Развитые в этой книге идеи позволяют генерировать неограниченное количество новых алгоритмов, адаптированных к конкретным практическим приложениям. Таким образом, можно говорить о *классе генетических алгоритмов*, в которых используются идеи самоадаптации, аналогичные выживаемости наиболее приспособленных организмов в процессе эволюции.

Первоначально Холланд и его коллеги работали над генетическими алгоритмами, которые уместно назвать *бинарными*. Позднее генетические алгоритмы расширились за счет новой модели с *непрерывными параметрами*. В последние годы развиваются генетические алгоритмы, построенные не на *последовательных структурах*, а на *деревьях*.

2. Бинарные генетические алгоритмы

2.1 Постановка и интерпретация задачи

Пусть задана функция $D(x_1, x_2, \dots, x_n)$ и определены области допустимых значений аргументов $x_i \in X_i, i = \overline{1, n}$. Требуется найти $\max_x D(x_1, x_2, \dots, x_n)$.

Вектор $x = (x_1, x_2, \dots, x_n)$ будем рассматривать как организм (особь) с указанными характеристиками. Функция $D(x_1, x_2, \dots, x_n)$ интерпретируется как характеристика приспособленности (выживаемости) организма с заданными свойствами, поэтому она названа функцией приспособленности (*fitness*). Создадим набор разных векторов $x = (x_1, x_2, \dots, x_n)$ (особей), которые образуют популяцию, и организуем эволюционный процесс, в котором особи будут совершенствоваться от поколения к поколению, то есть повышать функцию приспособленности. Особи с низким уровнем приспособленности ”погибают”, то есть удаляются из популяции. В результате эволюционного процесса выделяются лучшие особи, для которых функция приспособленности достигает максимального значения, что и является целью поставленной задачи.

Таким образом, для реализации изложенной идеи максимизации функции $D(x_1, x_2, \dots, x_n)$ нужно организовать эволюционный процесс. Холланд предложил по аналогии с *хромосомой* живого организма представить вектор $x = (x_1, x_2, \dots, x_n)$ в виде линейной структуры - цепочки бинарных символов. Для этого каждая переменная x_i кодируется с заданной точностью двоичным числом (число может содержать знак, целую часть и мантиссу) с числом разрядов l_i . Общая длина всей цепочки бинарных кодов переменных $x_i, i = \overline{1, 2, \dots, n}$, составит

$l = \sum_{i=1}^n l_i$. Эта бинарная цепочка в генетических алгоритмах также называется

хромосомой. Если задана хромосома, то путем декодирования можно восстановить значения $x_i, i = \overline{1, 2, \dots, n}$ (в пределах установленной точности) и рассчитать функцию приспособленности $D(x_1, x_2, \dots, x_n)$.

Пусть первое поколение состоит из N организмов, характеризующихся хромосомами $x^{(p)}, p = \overline{1, 2, \dots, N}$. Номер поколения обозначим t (дискретное время). Чтобы построить и описать эволюционный процесс, сформулируем основные принципы порождения последующих поколений. Эти принципы имеют отношение именно к *бинарным генетическим алгоритмам* (БГА):

1). Генетические алгоритмы работают с *кодами параметров*, а не их значениями.

2). Генетические алгоритмы организуют поиск, отталкиваясь не от единственной точки в пространстве параметров, а от популяции точек. Этот принцип позволяет организовать параллельные вычисления.

3). Реализация генетических алгоритмов требует вычисления только целевой функции и не предусматривает расчета других характеристик (например, производных).

4). Генетические алгоритмы применяют на всех этапах вычислений вероятностные, а не детерминированные правила переходов.

2.2 Основные операции эволюционного процесса

В генетических алгоритмах для организации эволюционного процесса поиска $\max_x D(x_1, x_2, \dots, x_n)$ применяются следующие процедуры:

- репродукция,
- кроссинговер,
- селекция,
- мутация.

Репродукция состоит в отборе из текущего t -поколения хромосом некоторого числа хромосом и создания их копий. Таким образом, формируется пул для порождения новых особей (хромосом). Отбор производится случайно и таким образом, чтобы в пул попали хромосомы, которым главным образом соответствует высокий показатель выживаемости, хотя могут допускаться и “мало приспособленные” особи для предотвращения вырождению эволюционного процесса.

Из репродуцированных хромосом образуются родительские пары, которые обмениваются частями хромосом и формируют дочерние хромосомы. Эта операция названа *кроссинговер*. Обмен генетическим материалом производится случайно по аналогии с живой природой.

Для создания следующего $(t+1)$ -поколения производится отбор хромосом (*селекция*) из объединенного множества хромосом t -поколения и дочерних хромосом. Лучшая с точки зрения значения функции приспособленности хромосома всегда попадает в следующее поколение. Другие хромосомы могут отбираться случайно. Допускается вариант отбора только лучших особей (элитарная стратегия), хотя в этом случае возможна потеря некоторых глобальных максимумов, если их несколько, или торможение процесса эволюции около локального максимума.

Далее селектированные для следующего поколения хромосомы подвергаются операции *мутации*, которая состоит в случайном точечном изменении бинарных разрядов хромосом. Мутации необходимы для того, чтобы эволюционный процесс не вырождался, чтобы появлялся новый генетический материал для поиска более высоких значений функции приспособленности. Кроме того, мутации предотвращают “застывание” поискового процесса в локальных экстремумах.

Таким образом, БГА в каждом t -поколении зондирует с помощью N хромосом пространство признаков и с помощью операций репродукции и кроссинговера реализует случайное, но направленное изменение бинарных кодов параметров x_i , $i=1, 2, \dots, n$, которое позволяет *локализовать* поисковую зону вблизи предполагаемых максимумов.

2.3 Пример

Рассмотрим простейший генетический алгоритм на примере максимизации функции $D(x) = x^2$. Здесь число параметров $n=1$. Пусть поисковая область значения параметра x является интервалом $[0, 31]$, так что при заданной точности 1 значения параметра x он может быть закодирован пятью двоичными разрядами. Следовательно, длина хромосомы также равна 5.

Допустим, что каждое поколение содержит 4 хромосомы. Первое поколение генерируется случайно с помощью датчика равномерно распределенных псевдослучайных чисел. В Таблице представлен результат начальной генерации поколения и последующей работы генетических операций в процессе формирования второго поколения.

Таблица

№ хромосомы	Хромосомы 1-го поколения	Функция приспособленности 1-го поколения	% от суммарной приспособленности	Репродукция	Позиция кроссинговера	Дочерние хромосомы	Функция приспособленности дочерних хромосом	Результат селекции	Результат мутации (2-е поколение)
1	01101	169	14.4	11000	4	11001	625	11011	11011
2	11000	576	49.2	01101		01100	144	11001	11000
3	01000	64	5.5	11000	2	11011	729	11000	11010
4	10011	361	30.9	10011		10000	256	10011	10011
Комментарий	Случайная генерация цифр 0 и 1 с вер. 0.5	Сумма значений 1170	Сумма значений 100 %	Результат случайного выбора пропорционально % от суммарной приспособленности	Значения получены случайно в диапазоне [1, 4] для каждой из пар	Результат кроссинговера	Сумма значений 1754	В этом алгоритме отбираются лучшие хромосомы среди дочерних + предыдущего поколения	Наблюдаются 2 мутации. Лучшая хромосома не подвергается мутации.

В модификации генетического алгоритма, рассмотренного в примере, для репродукции хромосомы отбираются случайно. Вероятность выбора из первого поколения i -й хромосомы пропорциональна значению максимизируемой функции, которое на ней достигается. В частности, вероятность выбора первой хромосомы равна 0.144, а второй – 0.492. Вторая хромосома, как это видно из 5-го столбца Таблицы, была отобрана для репродукции дважды.

Для порождения дочерних хромосом образуются две пары хромосом. В примере первую пару образуют хромосомы 1 и 2, а вторую – 3 и 4 из репродуцированных. Далее реализуется операция кроссинговера для обмена генетическим материалом между хромосомами в одной паре. В примере рассмотрен простейший вариант кроссинговера, когда случайно

устанавливается позиция одной точки и бинарные разряды каждой хромосомы, предшествующие этой точке, объединяются с бинарными разрядами другой хромосомы, следующими за точкой.

Для дочерних хромосом производится переход от кодов параметров к их значениям x , что позволяет рассчитать значения функции приспособленности. Эти значения указаны в восьмом столбце таблицы.

Далее производится операция селекции – отбор хромосом в следующее поколение. Здесь использован метод отбора лучших хромосом (по критерию значения функции приспособленности) среди хромосом 1-го поколения и дочерних. Лучшая из хромосом не подвергается мутации и включается в состав 2-го поколения без изменений. В остальных случайным образом инвертируются бинарные разряды с достаточно малой вероятностью.

В результате проведенных операций значение лучшей функции приспособленности во втором поколении возросло от 576 до 729 и в принципе не может уменьшиться.

После формирования второго поколения описанные процедуры повторяются до тех пор, пока достигнутое лучшее решение не станет повторяться на протяжении нескольких поколений. Оно и рассматривается как найденный глобальный максимум.

2.4 Некоторые особенности построения бинарных генетических алгоритмов

Для повышения эффективности случайного поиска желательно найти лучшие стартовые условия для генетического алгоритма. Поэтому на начальном этапе (перед организацией эволюционного процесса) случайным образом генерируют несколько поколений и выбирают лучшее из них в качестве стартового для генетического алгоритма.

Разнообразны способы выбора родительских пар для порождения следующего поколения. Простейший способ ориентирован на элитарную стратегию. Согласно этому подходу 50% лучших хромосом выделяются для формирования родительских пар, а остальные 50% отбрасываются. В этом варианте селекции возможен процесс вырождения, когда следующее поколение не содержит достаточного генетического разнообразия для поиска новых областей предполагаемых максимумов функции приспособленности. Можно препятствовать процессу вырождения увеличением интенсивности мутаций, однако это может приводить к неустойчивости процедуры поиска или к затягиванию процесса сходимости к финальному решению.

Элитарная стратегия требует проведения трудоемкой операции сортировки хромосом по значению функции приспособленности. Можно существенно сократить вычисления, если отбирать для репродукции только те хромосомы, функция приспособленности которых превышает некоторый заданный порог. Порог можно подстраивать по мере продвижения к экстремуму.

Случайный способ формирования родительских пар, рассмотренный в предыдущем пункте и предложенный Холландом, является универсальным, хотя и не дает такой высокой скорости сходимости, которая может наблюдаться при использовании элитарной стратегии.

Вероятности отбора особей для репродукции при использовании случайной стратегии можно рассчитывать по другой методике, которая опирается на ранг (порядковый номер) хромосомы n в последовательности, упорядоченной по возрастанию значения функции приспособленности. Если

поколение содержит N хромосом, то вероятность p_n отбора хромосомы ранга n определяется формулой:

$$p_n = \frac{N - n + 1}{\sum_{i=1}^N i},$$

которая удовлетворяет условию $\sum_{n=1}^N p_n = 1$.

Для организации репродукции в генетических алгоритмах может быть использована и соревновательная стратегия: из поколения случайно отбирается малая группа хромосом, лучшая из которых используется в качестве родительской. Эта процедура повторяется столько раз, сколько необходимо для дальнейшего формирования родительских пар.

Мутации в генетических алгоритмах влияют на процесс оптимизации в двух направлениях:

- мутации предотвращают слишком быструю сходимость лучшей хромосомы к *локальному экстремуму*, при которой теряется генетический материал для зондирования новых областей поиска,
- мутации затягивают поисковый процесс за счет постоянного случайного разрушения хромосом, соответствующих хорошему решению задачи.

Интенсивность мутаций определяется значением вероятности инвертирования одного бинарного разряда в хромосоме поколения. Обозначим этот параметр генетического алгоритма p_μ . Среднее число мутаций в поколении при условии их независимости в разных разрядах хромосомы и в разных хромосомах равно LNp_μ с учетом введенных ранее обозначений. В начале оптимизационного процесса следует задавать относительно большие значения p_μ для активного просмотра поисковой зоны, а по мере сходимости процедуры оптимизации следует значение p_μ уменьшать. Мутации не рекомендуется применять к лучшей из хромосом.

Рассмотрим два способа организации мутаций. Первый способ состоит в последовательном проходе по всем бинарным разрядам всех хромосом в поколении и случайной инверсии разрядов с заданной вероятностью p_μ . В другом способе реализации мутаций их общее количество фиксируется, а положение мутированных разрядов разыгрывается случайно: в матрице размерности $[N, L]$ случайно разыгрываются координаты $i=1,2,\dots,N, j=1,2,\dots,L$, которые определяют позицию для мутации бинарного кода.

Для анализа результата поиска максимума функции приспособленности генетическим алгоритмом рекомендуется контролировать статистическое распределение значений этой функции в каждом поколении. В частности, можно рассчитывать среднее значение функции приспособленности в текущем t -поколении и ее дисперсию. При нормальном протекании процесса среднее значение стремится к максимальному значению функции приспособленности в поколении, а дисперсия – к нулю.

3. Математические основы бинарных генетических алгоритмов

Практические реализации бинарных генетических алгоритмов в разных приложениях неизменно показывают надежную сходимость к глобальному экстремуму. Теоретическое обоснование этого факта было дано с

использованием понятия “схемы”. Схема – это совокупность бинарных разрядов на фиксированных позициях хромосомы. Например, две хромосомы

$$\begin{array}{ccccccc} 0 & 1 & \boxed{1} & \boxed{0} & \boxed{1} & 0 & 0 & 0 & 1 & \boxed{1} \\ 1 & 0 & \boxed{1} & \boxed{0} & \boxed{1} & 1 & 0 & 1 & 1 & \boxed{1} \end{array}$$

содержат одну и ту же схему:

$$**\boxed{101}*****\boxed{1}.$$

Символом * отмечены те позиции, которые могут принимать произвольные значения в хромосоме с заданной схемой. Будем использовать для схем обозначение H .

Хромосому с порядковым номером p в поколении t обозначим $c^{(p)}(t)$. Множество $\Omega(t)$ образовано объединением всех хромосом поколения t . Функция приспособленности для хромосомы $c^{(p)}(t)$ будет далее обозначаться $f^{(p)}(t)$.

Если в процессе теоретического анализа нужно подчеркнуть, что некоторая хромосома содержит схему H , то она будет отмечаться нижним индексом H : $c_H^{(p)}(t)$. Соответствующее значение функции приспособленности будет также отмечаться нижним индексом H : $f_H^{(p)}(t)$. Множество хромосом поколения, содержащих схему H , будем обозначать $\Omega_H(t)$.

Рангом схемы (или ее “порядком”) $o(H)$ (от *order* - порядок) называется число фиксированных бинарных позиций в схеме H . В приведенном выше примере $o(H)=4$. Значение ранга схемы влияет на вероятность ее разрушения в процессе выполнения операции мутации. Если бинарные разряды хромосомы мутируют независимо друг от друга с одной и той же вероятностью p_μ , то вероятность p_H сохранения схемы H в хромосоме $c_H^{(p)}(t)$ после операции мутации определяется выражением:

$$p_H = (1 - p_\mu)^{o(H)} \cong 1 - o(H) p_\mu. \quad (1)$$

В формуле (1) значение p_μ предполагается малым.

Длиной схемы $\delta(H)$ называется число межсимвольных интервалов между первым и последним значимыми разрядами в схеме. В приведенном выше примере $\delta(H)=7$. Если схема содержит один бинарный разряд ($o(H)=1$), то ее длина равна 0. Длина схемы влияет на вероятность ее разрушения в процессе выполнения операции кроссинговера. Если кроссинговер реализуется простейшим способом, который рассмотрен на примере в п. 2.3, то при равновероятной позиции точки кроссинговера в хромосоме длины l вероятность q_H сохранения схемы H длины $\delta(H)$ определяется формулой:

$$q_H = 1 - \frac{\delta(H)}{l-1}. \quad (2)$$

Пусть в поколении t содержится $m_H(t)$ хромосом, включающих схему H . Проследим динамику числа $m_H(t)$. По логике построения генетических алгоритмов “выживаемость” схемы в процессе эволюции тем больше, чем более связана эта схема с большими значениями функции приспособленности. Отсюда следует, что если значение $m_H(t)$ имеет тенденцию к увеличению, то рассматриваемая схема связана с увеличением функции приспособленности и с большой вероятностью принадлежит лучшей хромосоме. Если такого позитивного влияния схемы на значение функции приспособленности нет, то схема должна разрушаться и значение $m_H(t)$ уменьшается.

Проведем расчет значения $m_H(t+1)$ в предположении, что генетический алгоритм выполняет основные генетические операции по правилам,

рассмотренным в п. 2.3. В процессе репродукции в соответствии с этими правилами хромосомы отбираются случайно с вероятностями, пропорциональными значениям функции приспособленности. Для одной хромосомы эта вероятность определяется формулой

$$\frac{f_H^{(p)}(t)}{\sum_{c^{(p)}(t) \in \Omega(t)} f^{(p)}(t)}.$$

Поскольку поколение содержит множество хромосом со схемой H , вероятность репродукции хромосомы из этого множества определяется формулой:

$$\frac{\sum_{c^{(p)}(t) \in \Omega_H(t)} f_H^{(p)}(t)}{\sum_{c^{(p)}(t) \in \Omega(t)} f^{(p)}(t)} = \frac{\overline{f_H}(t)}{\overline{f}(t)} \frac{m_H(t)}{N}. \quad (3)$$

В выражении (3) под $\overline{f_H}(t)$ понимается среднее значение функции приспособленности среди всех хромосом, которые содержат схему H , а под $\overline{f}(t)$ - среднее значение по всему поколению. При выводе формулы (3) принято во внимание, что

$$\frac{\sum_{c^{(p)}(t) \in \Omega_H(t)} f_H^{(p)}(t)}{m_H(t)} = \overline{f_H}(t), \quad \frac{\sum_{c^{(p)}(t) \in \Omega(t)} f^{(p)}(t)}{N} = \overline{f}(t). \quad (4)$$

Так как для репродукции в рассматриваемой схеме реализации генетического алгоритма отбираются N хромосом, то среднее число хромосом в отобранной группе, содержащих схему H , равно

$$\frac{\overline{f_H}(t)}{\overline{f}(t)} m_H(t).$$

Указанное среднее число хромосом со схемой H изменяется в результате применения операций кроссинговера и мутации. Для коррекции используются формулы (1) и (2):

$$m_H(t+1) \approx m_H(t) \frac{\overline{f_H}(t)}{\overline{f}(t)} \left(1 - \frac{\delta(H)}{l-1}\right) (1 - o(H) p_\mu). \quad (5)$$

Полученное приближенное уравнение динамики для переменной $m_H(t)$ можно упростить, учитывая второй порядок малости произведения $\frac{\delta(H)}{l-1} o(H) p_\mu$ в правой части уравнения (5):

$$m_H(t+1) \approx m_H(t) \frac{\overline{f_H}(t)}{\overline{f}(t)} \left(1 - \frac{\delta(H)}{l-1} - o(H) p_\mu\right). \quad (6)$$

Если выполняется условие

$$\frac{\overline{f_H}(t)}{\overline{f}(t)} \left(1 - \frac{\delta(H)}{l-1} - o(H) p_\mu\right) > 1,$$

то наблюдается рост числа хромосом $m_H(t)$ со схемой H в следующем поколении, в противном случае происходит деградация этой схемы и она постепенно в ходе эволюционного процесса вытесняется другими схемами. Из формулы (6) следует, что практически выживают только короткие схемы с небольшим рангом.

Полученный результат составляет основное содержание *теоремы о “выживаемости” схем*, которая теоретически обосновывает закрепление в поколениях тех схем, которые входят в лучшую из хромосом и обеспечивают сходимость поискового процесса.

Рассмотрим пример. Допустим, что длина хромосомы $l=200$, а длина схемы – $\delta=8$ бинарных разрядов. Порядок схемы допустим равным 4, а вероятность мутаций p_μ положим равным 0.01. Пусть среднее значение функции приспособленности для хромосом, которые содержат рассматриваемую схему, превышает среднее значение функции приспособленности для всего поколения в 1.3 раза:

$$\frac{\overline{f_H}(t)}{\overline{f}(t)} = 1.3.$$

В этих условиях согласно теореме о выживаемости схем

$$m_H(t+1) \approx m_H(t) \cdot 1.3 \cdot (1 - 8/199 - 0.04) \approx 1.196 m_H(t).$$

Полученный результат означает, что схема имеет большой шанс на выживание.

4. Генетические алгоритмы для оптимизации функции с произвольными действительными параметрами

Бинарные генетические алгоритмы имеют ряд ограничений по точности представления переменных их кодами. Кроме того, в процессе реализации генетических операций в бинарном генетическом алгоритме необходимо многократно выполнять декодирование для расчета значения функции приспособленности. Следует отметить также, что избранный тип бинарного кода (допускаются различные бинарные коды, кроме стандартного представления значения переменной в двоичной системе счисления) может влиять на скорость протекания поискового процесса. Все это послужило основанием для построения генетических алгоритмов, работающих непосредственно с действительными числами. Их называют генетическими алгоритмами с непрерывными параметрами.

При использовании действительных параметров теряют смысл операции кроссинговера и мутации в тех вариантах, которые были описаны применительно к бинарным генетическим алгоритмам: действительное число представляет собой единый объект, который не подлежит дроблению.

Аналог операции кроссинговера для непрерывных параметров предложил Радклифф (Radcliff, 1991). Воспользуемся обозначениями $g_i'(t)$ и $g_i''(t)$ для значений i -го параметра в двух хромосомах родительской пары, полученной отбором из t -поколения. Предлагается рассчитать значения i -го признака двух дочерних хромосом по формулам:

$$\begin{aligned} g_i'(t+1) &= \beta g_i'(t) + (1 - \beta) g_i''(t), \\ g_i''(t+1) &= (1 - \beta) g_i'(t) + \beta g_i''(t), \end{aligned} \quad (7)$$

где положительный параметр $\beta \in (0,1)$. Возможны разные модификации этой идеи, связанные с указанием того, какие из параметров подвергаются пересчету по приведенным формулам. Допускается случайный выбор значения β .

Недостаток формулы (7) состоит в том, что новые значения параметров потомков не выходят за пределы интервала значений $(\min g_i, \max g_i)$, где минимальное и максимальное значения рассчитаны среди всех первоначально сгенерированных хромосом первого поколения. Таким образом, область поиска глобального экстремума оказывается жестко ограниченной. Этот недостаток

преодолевается модификацией формул (7), которая допускает отрицательные значения коэффициентов:

$$\begin{aligned} g_i'(t+1) &= \beta g_i'(t) + (1-\beta) g_i''(t), \\ g_i''(t+1) &= (1+\beta) g_i'(t) - \beta g_i''(t), \\ g_i'''(t+1) &= -\beta g_i'(t) + (1+\beta) g_i''(t). \end{aligned} \tag{8}$$

Формулы (8) предполагают формирование трех потомков. Если область поиска имеет ограничения по значениям оптимизируемых параметров, то отбрасываются те потомки, у которых значения признаков выходят за допустимые пределы. В отличие от процедуры (7), в которой выполняется интерполяция значений параметров, в формуле (8) выполняется дополнительно их экстраполяция.

Мутация непрерывных значений параметров, которая необходима в любом эволюционном процессе, производится заменой текущего значения на новое, равномерно распределенное в области допустимых значений параметра.

При рационально организованном эволюционном процессе можно получить близкое к оптимальному решению оптимизационной задачи уже в 7 – 10-ом поколении.