

Генетические алгоритмы

Генетические Алгоритмы - адаптивные методы поиска, которые в последнее время часто используются для решения задач оптимизации (поиска оптимального решения). Они основаны на генетических процессах биологических организмов: биологические популяции развиваются в течение нескольких поколений, подчиняясь законам естественного отбора и по принципу "выживает наиболее приспособленный", открытому Чарльзом Дарвином. Подражая этому процессу генетические алгоритмы способны "развивать" решения реальных задач, если те соответствующим образом закодированы. Например, ГА могут использоваться, чтобы проектировать структуры моста, для поиска максимального отношения прочности/веса, или определять наименее расточительное размещение для нарезки форм из ткани. Они могут также использоваться для интерактивного управления процессом, например на химическом заводе, или балансировании загрузки на многопроцессорном компьютере.

В природе особи в популяции конкурируют друг с другом за различные ресурсы, такие, например, как пища или вода. Кроме того, члены популяции одного вида часто конкурируют за привлечение брачного партнера. Те особи, которые наиболее приспособлены к окружающим условиям, будут иметь относительно больше шансов воспроизвести потомков. Слабо приспособленные особи либо совсем не произведут потомства, либо их потомство будет очень немногочисленным. Это означает, что гены от высоко адаптированных или приспособленных особей будут распространяться в увеличивающемся количестве потомков на каждом последующем поколении. Комбинация хороших характеристик от различных родителей иногда может приводить к появлению "суперприспособленного" потомка, чья приспособленность больше, чем приспособленность любого из его родителя. Таким образом, вид развивается, лучше и лучше приспособляясь к среде обитания.

ГА используют прямую аналогию с таким механизмом. Они работают с совокупностью особей, каждая из которых представляет возможное решение данной проблемы.

На первом шаге алгоритма формируется исходная популяция особей, где каждая особь популяции представляет собой решение задачи, иначе говоря, является кандидатом на решение. Формирование исходной популяции, как правило, происходит с использованием какого-нибудь случайного закона, в ряде случаев исходная популяция может быть также результатом работы другого алгоритма. Необходимо отметить, что особи популяции могут состоять как из одной, так и из нескольких хромосом. Каждая хромосома особи в свою очередь состоит из генов, причем количество генов в хромосоме определяется количеством варьируемых параметров решаемой задачи (аргументов целевой функции).

Для применения генетических операторов значения этих параметров должны быть представлены в виде двоичной последовательности, т.е. двоичной строки, состоящей из нескольких бит. Количество бит при кодировании гена (хромосомы) зависит от требуемой точности решаемой задачи. Оптимальным считается такой выбор, когда выполняется соотношение

$$n \geq \log_2 \left(\frac{x_{\max} - x_{\min}}{\Delta} \right)$$

где

x_{\max} , x_{\min} - максимальное и минимальное значение аргумента целевой функции;

Δ - погрешность решения задачи;

n – количество бит, используемых для кодирования значения аргумента.

На втором шаге работы генетического алгоритма происходит отбор или селекция наиболее приспособленных особей, имеющих наиболее предпочтительные значения функции пригодности по сравнению с остальными особями. После чего к отобранным особям применяются операторы скрещивания и мутации.

В классическом генетическом алгоритме отбор наиболее приспособленных особей осуществляется случайным образом с помощью различных методов генерации дискретных случайных величин, имеющих различные законы распределения. Среди данных методов следует упомянуть отбор методом “колеса рулетки, пропорциональный отбор, равновероятный отбор и т.д. Каждый из перечисленных методов имеет как свои достоинства, так и недостатки, например, общим недостатком указанных методов является то, что при их использовании в некотором поколении наилучшие особи популяции могут быть потеряны. Одним из способов преодоления этого недостатка является использование элитного отбора, который предусматривает сохранение “наилучшей” особи в популяции (“лучшая” особь всегда переходит в следующее поколение).

На третьем этапе реализуется операция скрещивания особей. Смысл данной процедуры сводится к нахождению новых комбинаций генетического кода хромосом путем обмена случайным образом участков генетического кода у двух особей, прошедших отбор. Это способствует “получению” дополнительного числа новых особей, среди которых могут оказаться как более приспособленные, так и менее приспособленные особи. Это явление объясняется тем, что точка скрещивания (локус) выбирается случайным образом.

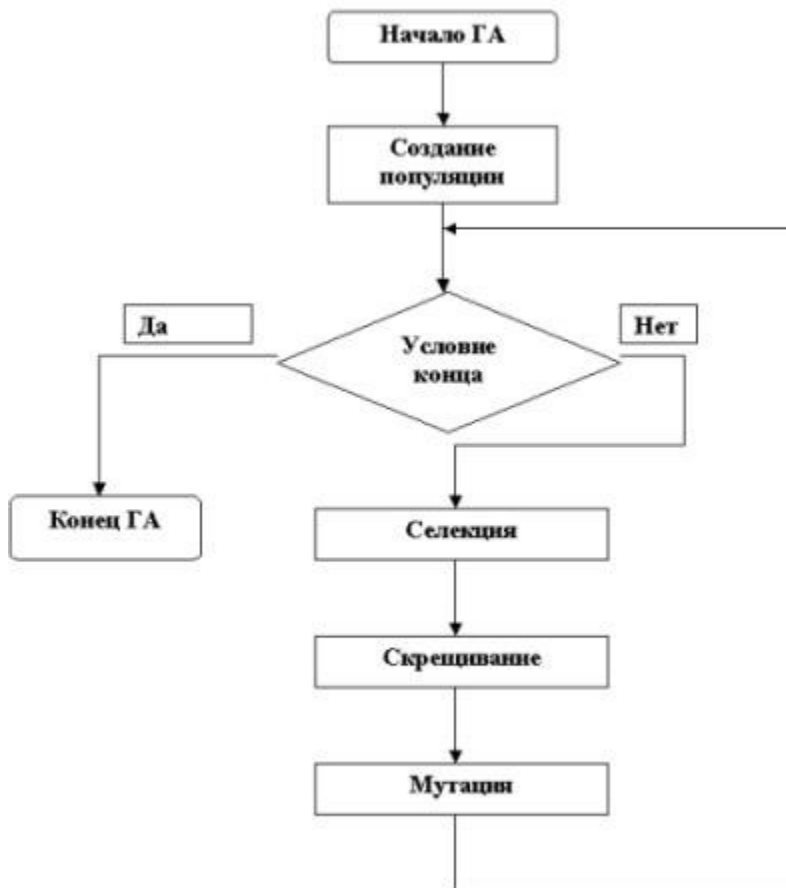
На четвертом этапе к особям популяции, полученным в результате скрещивания, применяется операция мутации. С помощью данной операции можно получать принципиально новые генотипы и фенотипы особи, что приводит к еще большему разнообразию особей в рассматриваемой популяции. Суть этого оператора заключается в следующем: в популяции случайным образом выбирается особь и так же случайно выбирается позиция гена, в которой значение изменяется на противоположное (или). Внесение таких случайных изменений позволяет существенно расширить пространство поиска приемлемых решений задачи целочисленной оптимизации.

В процессе работы алгоритма все указанные операторы применяются многократно

и ведут к постепенному изменению исходной популяции в направлении улучшения значения функции пригодности.

В качестве критериев останова работы генетического алгоритма принято рассматривать следующие условия:

- сформировано заданное число поколений,
- популяция достигла заданного уровня качества (например, 80% особей имеют одинаковую генетическую структуру или одинаковое значение функции пригодности),
- достигнут некоторый уровень сходимости, при котором улучшение популяции не происходит.



Простой ГА случайным образом генерирует начальную популяцию. Работа ГА представляет собой итерационный процесс, который продолжается до тех пор, пока не выполнятся заданное число поколений или какой-либо иной критерий останова. На каждом поколении ГА реализуется отбор пропорционально приспособленности, односточный кроссинговер и мутация. Сначала, пропорциональный отбор назначает каждой структуре вероятность $P_s(i)$ равную отношению ее приспособленности к суммарной приспособленности популяции:

$$P_s(i) = \frac{f(i)}{\sum_{i=1}^n f(i)} \quad (*)$$

Затем происходит отбор всех n особей для дальнейшей генетической обработки, согласно величине $P_s(i)$. Простейший пропорциональный отбор - рулетка - отбирает особей с помощью n "запусков" рулетки. Колесо рулетки содержит по одному сектору для каждого члена популяции. Размер i -ого сектора пропорционален соответствующей величине $P_s(i)$. При таком отборе члены популяции с более высокой приспособленностью с большей вероятностью будут чаще выбираться, чем особи с низкой приспособленностью.

После отбора, n выбранных особей подвергаются кроссинговеру (иногда называемому рекомбинацией) с заданной вероятностью P_s .

n строк случайным образом разбиваются на $n/2$ пары. Для каждой пары с вероятностью P_s может применяться кроссинговер. Соответственно с вероятностью $1-P_s$ кроссинговер не происходит и неизменные особи переходят на стадию мутации. Если кроссинговер происходит, полученные потомки заменяют собой родителей и переходят к мутации.

Одноточечный кроссинговер работает следующим образом. Сначала, случайным образом выбирается одна из 1-1 точек разрыва. (Точка разрыва - участок между соседними битами в строке.) Обе родительские структуры разрываются на два сегмента по этой точке. Затем, соответствующие сегменты различных родителей склеиваются и получаются два генотипа потомков.

После того, как закончится стадия кроссинговера, выполняются операторы мутации. В каждой строке, которая подвергается мутации, каждый бит с вероятностью P_m изменяется на противоположный. Популяция, полученная после мутации записывает поверх старой и этим цикл одного поколения завершается. Последующие поколения обрабатываются таким же образом: отбор, кроссинговер и мутация.

Пример

Найти максимум функции

$$f(x) = 2 \cdot x^2 + 1$$

для целочисленной переменной x , принимающей значения от 0 до 31.

Решение:

Используя двоичную систему счисления, закодируем числа от 0 до 31. Тогда хромосомы приобретут вид двоичных последовательностей, состоящих из 5 битов (0 как 00000, 31 как 11111)

В роли функции приспособленности будет выступать функция

$$f(x) = 2 \cdot x^2 + 1$$

Тогда приспособленность хромосомы будет определяться значением функции $f(x)$

Выберем случайным образом исходную популяцию, состоящую из 6 кодовых последовательностей ($N=6$). Пусть выбраны хромосомы:

$ch_1=[10011]$
 $ch_2=[00011]$
 $ch_3=[00111]$
 $ch_4=[10101]$
 $ch_5=[01000]$
 $ch_6=[11101]$

Соответствующие им фенотипы – это числа от 0 до 31:

$ch_1^*=19$, $ch_2^*=3$, $ch_3^*=7$, $ch_4^*=21$, $ch_5^*=8$, $ch_6^*=29$

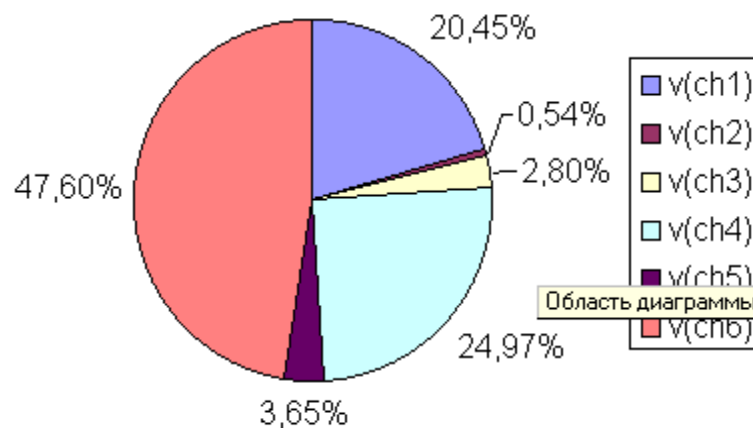
По формуле (1) рассчитаем значения функции приспособленности для каждой хромосомы в популяции, получим:

$f(ch_1)=723$, $f(ch_2)=19$, $f(ch_3)=99$, $f(ch_4)=883$, $f(ch_5)=129$,
 $f(ch_6)=1683$

Селекция хромосом осуществляется методом рулетки.

Вероятности выбора определяем по формуле (*):

$V(ch_1)=20.45$, $V(ch_2)=0.54$, $V(ch_3)=2.8$, $V(ch_4)=24.97$,
 $V(ch_5)=3.65$, $V(ch_6)=47.60$



Розыгрыш с помощью колеса рулетки сводится к случайному выбору числа из интервала $[0, 100]$, указывающего на соответствующий сектор на колесе, т.е. на конкретную хромосому.

Допустим, что выбраны числа: 97, 26, 54, 13, 31, 88. Это означает выбор хромосом. Это означает выбор хромосом

$$ch_6, ch_4, ch_6, ch_1, ch_4, ch_6$$

Допустим, что для скрещивания сформированы пары:

ch1 и ch4

ch4 и ch6

ch6 и ch6.

Для первой пары случайным образом выбрана точка скрещивания, равная 3. Для второй пары — 2.

При условии, что вероятность мутации $P_m = 0$, в новую популяцию включаются хромосомы:

от первой пары: ch1 = [10001], ch2 = [10111]

от второй пары: ch3 = [10101], ch4 = [11101]

от третьей пары: ch5 = [11101], ch6 = [11101]

Декодировав полученные последовательности, получим фенотипы:

$$ch_1^* = 17 \quad ch_2^* = 23 \quad ch_3^* = 21 \quad ch_4^* = 29 \quad ch_5^* = 29 \quad ch_6^* = 29$$

Как мы видим количество наиболее приспособленных особей увеличивается.

Продолжая данный процесс, уже на следующей итерации может быть получена хромосома [11111] с максимальным значением функции принадлежности $2 * 31^2 + 1 = 1923$

Генетический алгоритм используется для решения многих задач оптимизации:

- составление расписаний (производственных, учебных и т.д.)
- задачи раскроя-упаковки
- аппроксимации и т.д.