

Авторы:

Завгородний Станислав Дмитриевич

студент

Швейкин Владислав Витальевич

студент

Танаев Иван Владимирович

студент

Дмитриев Егор Андреевич

студент

ФГАОУ ВО «Самарский национальный исследовательский
университет им. академика С.П. Королева»

г. Самара, Самарская область

ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ И ИХ ПРИМЕНЕНИЕ

Аннотация: в данной статье рассмотрен эвристический поисковый алгоритм, который используется для моделирования и решения оптимизационных задач, называемый генетическим алгоритмом, и его практическое применение.

Ключевые слова: хромосомы, генотип, мутация, селекция, популяция, родители, потомки, кроссовер.

Описание алгоритма

Задачи формализуется так, чтобы решение этой задачи можно было бы закодировать вектором генов (генотипа), в котором в качестве гена могут быть биты, числа и другие объекты. В классических реализациях длина генотипа фиксирована, но также существуют разновидности ГА, в которых такого ограничения нет.

Как правило генотипы начальной популяции создаются случайным образом. Затем происходит их оценка с использованием «функции приспособленности», поэтому с каждым генотипом сопоставляется специальное значение (приспособленность), определяющее эффективность решения поставленной задачи.

Важно следить за «гладкостью рельефа» функции приспособленности при ее выборе.

С учетом значения приспособленности из полученного множества решений выбираются те, к которым применяются «генетические операторы»: скрещивание и мутация. В результате получают новые решения, для которых также вычисляется значение приспособленности и проводится селекция лучших решений для последующего поколения.

Эти действия проводятся итеративно. Таким образом моделируется «эволюционный процесс», который продолжается несколько поколений. Процесс останавливается при выполнении определенного критерия. Таким критерием может быть:

- отыскание глобального решения;
- израсходование времени, отпущенного на эволюцию;
- израсходование числа поколений, отпущенных на эволюцию.

Генетические Алгоритмы обычно используются для поиска решений в многомерных пространствах поиска.

Этапы генетического алгоритма:

1. Задание функции приспособленности для популяционных особей
2. Создание начальной популяции

Начало цикла:

1. Скрещивание.
2. Мутирование.
3. Вычисление значений функции приспособленности для всех особей.
4. Селекция.

5. Конец цикла при выполнении условия остановки, иначе переход к началу цикла.



Рис. 1. Схема работы генетического алгоритма

Перспективы применения генетических алгоритмов

Генетические алгоритмы оптимизации, являясь алгоритмами случайно-направленного поиска, применяются в тех случаях, когда формулировка задачи в виде, пригодном для быстрых алгоритмов локальной оптимизации, очень трудна или невозможна, а также при задаче оптимизации недифференцируемой функции или задаче глобальной многоэкстремальной оптимизации. Такие алгоритмы нашли применение в решении Диофантовых уравнений.

Пусть имеется уравнение: $a + 2b + 3c + 4d = 30$. Его корни расположены на отрезке от 1 до 30. В качестве первого поколения возьмем 5 случайных значений a, b, c, d .

(a,b,c,d)
(1,28,15,3)
(14,9,2,4)
(13,5,7,3)
(23,8,16,19)
(9,13,5,2)

Рис. 2. 1-е поколение и его содержимое

Для вычисления коэффициентов выживаемости подставляется каждое решение в исходное выражение. Этим значением будет расстояние от вычисленного значения до 30.

Коэффициент выживаемости
$ 114-30 =84$
$ 54-30 =24$
$ 56-30 =26$
$ 163-30 =133$
$ 58-30 =28$

Рис. 3. Коэффициенты выживаемости первого поколения

Меньшие значения наиболее близкие к 30 более будут востребованными. Большие значения будут иметь меньший коэффициент выживаемости. Теперь вычисляются значения вероятностей выбора «хромосомы».

Подходящестъ
$(1/84)/0.135266 = 8.80\%$
$(1/24)/0.135266 = 30.8\%$
$(1/26)/0.135266 = 28.4\%$
$(1/133)/0.135266 = 5.56\%$
$(1/28)/0.135266 = 26.4\%$

Рис. 4. Вероятность оказаться родителем

Далее выбираются пять пар родителей, порождающих одного потомка. Шанс стать родителем каждый раз будет одинаковым и равным шансу на выживание.

Хромосома отца	Хромосома матери
3	1
5	2
3	5
2	5
5	3

Рис. 5. Симуляция выбора родителей

Каждый потомок содержит информацию о генах родителей. Это можно обеспечить разными способами. В данном случае используется «кроссовер». Если у матери набор решений a_1, b_1, c_1, d_1 и у отца a_2, b_2, c_2, d_2 , то возможны 6 различных кроссоверов.

Хромосома-отец	Хромосома-мать	Хромосома-потомок
$a_1 b_1, c_1, d_1$	$a_2 b_2, c_2, d_2$	a_1, b_2, c_2, d_2 or a_2, b_1, c_1, d_1
$a_1, b_1 c_1, d_1$	$a_2, b_2 c_2, d_2$	a_1, b_1, c_2, d_2 or a_2, b_2, c_1, d_1
$a_1, b_1, c_1 d_1$	$a_2, b_2, c_2 d_2$	a_1, b_1, c_1, d_2 or a_2, b_2, c_2, d_1

Рис. 6. Кроссоверы между родителями

Теперь это проделывается с родителями.

Хромосома-отец	Хромосома-мать	Хромосома-потомок
(13 5,7,3)	(1 28,15,3)	(13,28,15,3)
(9,13 5,2)	(14,9 2,4)	(9,13,2,4)
(13,5,7 3)	(9,13,5 2)	(13,5,7,2)
(14 9,2,4)	(9 13,5,2)	(14,13,5,2)
(13,5 7, 3)	(9,13 5, 2)	(13,5,5,2)

Рис. 7. Симуляция кроссоверов хромосом родителей

Теперь вычисляются коэффициенты выживаемости потомков.

Хромосома-потомок	Коэффициент выживаемости
(13,28,15,3)	$ 126-30 =96$
(9,13,2,4)	$ 57-30 =27$
(13,5,7,2)	$ 57-30 =22$
(14,13,5,2)	$ 63-30 =33$
(13,5,5,2)	$ 46-30 =16$

Рис. 8. Коэффициенты выживаемости потомков

Средняя приспособленность родителей была 59.4, а у потомков стала 38.8. Последующие поколения могут мутировать. Продолжая так дальше одна из хромосом достигнет выживаемости 0 и станет решением уравнения. Системы с большой популяцией сходятся к 0 более быстро и стабильно.

Так же генетические алгоритмы применяются и в реальной жизни. Например, команда роботов для разминирования территории, управляемых нейронной сетью оптимизированной генетическим алгоритмом, показывала лучшие результаты перед обычной командой роботов.

Список литературы

1. Божич В.И. Нейросетевое управление в мультиагентной системе с самоорганизующейся коммуникацией / В.И. Божич, Р.Н. Кононенко, А.А. Абияка // Материалы Всеросс. конф. «Нейроинформатика-99». Ч. 3. – М.: МИФИ, 1999. – С. 239–246.

2. Николенко С. Генетические алгоритмы [Электронный ресурс]. – Режим доступа: <http://logic.pdmi.ras.ru/~sergey/teaching/ml/04-genetic.pdf>