С.Сейфуллиннің 120 жылдығына арналған **«Сейфуллин оқулары–10: Мемлекеттің индустриалды–инновациялық саясатын құрудағы бәсекеге қабілетті кадрларды дайындау келешегі мен ғылымның рөлі»** атты Халықаралық ғылыми-теориялық конференциясының материалдары = Материалы Международной научно-теоретической конференции **«Сейфуллинские чтения–10: Новые перспективы подготовки конкурентоспособных кадров и роль науки в формировании индустриально-инновационной политики страны»,** посвященная 120-летию со дня рождения С.Сейфуллина**.** – 2014. – Т.I, ч.3. – С.34-37

АЛГОРИТМИЧЕСКИЕ АСПЕКТЫ ЭВОЛЮЦИОННОГО ИЛИ ГЕНЕТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Иксымбаева Ж.С.

Интеллектом называется способность мозга решать (интеллектуальные) задачи путем приобретения, запоминания и целенаправленного преобразования знаний в процессе обучения на опыте и адаптации к разнообразным обстоятельствам.

В этом определении под термином «знания» подразумевается не только ту информацию, которая поступает в мозг через органы чувств. Такого типа знания чрезвычайно важны, но недостаточны для интеллектуальной деятельности. Дело в том, что объекты окружающей нас среды обладают свойством не только воздействовать на органы чувств, но и находиться друг с другом в определенных отношениях. Ясно, что для того, чтобы осуществлять в окружающей среде интеллектуальную деятельность (или хотя бы просто существовать), необходимо иметь в системе знаний модель этого мира. В этой информационной модели окружающей среды реальные объекты, их свойства и отношения между ними не только отображаются и запоминаются, но и, как это отмечено в данном определении интеллекта, могут мысленно «целенаправленно преобразовываться». При этом существенно то, что формирование модели внешней среды происходит «в процессе обучения на опыте и адаптации к разнообразным обстоятельствам».

Для того, чтобы пояснить, чем отличается интеллектуальная задача от просто задачи, необходимо ввести термин «алгоритм» — один из краеугольных терминов кибернетики.

Под алгоритмом понимают точное предписание о выполнении в определенном порядке системы операций для решения любой задачи из некоторого данного класса (множества) задач. Термин «алгоритм» происходит от имени узбекского математика Аль-Хорезми, который еще в IX веке предложил простейшие арифметические алгоритмы. В математике и кибернетике класс задач определенного типа считается решенным, когда для ее решения установлен алгоритм. Нахождение алгоритмов является естественной целью человека при решении им разнообразных классов задач.

Отыскание алгоритма для задач некоторого данного типа связано с тонкими и сложными рассуждениями, требующими большой изобретательности и высокой квалификации. Принято считать, что подобного рода деятельность требует участия интеллекта человека. Задачи, связанные с отысканием алгоритма решения класса задач определенного типа, будем называть интеллектуальными.

Генетический алгоритм является самым известным на данный момент представителем эволюционных алгоритмов, и по своей сути является алгоритмом для нахождения глобального экстремума многоэкстремальной функции. Генетический алгоритм представляет собой модель размножения живых организмов.

Для начала представим себе целевую функцию от многих переменных, у которой необходимо найти глобальных максимум или минимум: $f(x_1, x_2, x_3, \ldots, x_N)$.

Для того чтобы заработал генетический алгоритм, нам необходимо представить независимые переменные в виде хромосом. Как это делается? Как создать хромосомы?

Первым шагом будет преобразование независимых переменных в хромосомы, которые будут содержать всю необходимую информацию о каждой создаваемой особи. Имеется два варианта кодирования параметров:

- в двоичном формате;
- в формате с плавающей запятой.

В случае если мы используем двоичное кодирование, мы используем N бит для каждого параметра, причем N может быть различным для каждого параметра. Если параметр может изменяться между минимальным значением MIN и максимальным MAX, используем следующие формулы для преобразования:

r = g*(MAX – MIN) / (2^N – 1) + MIN.

g = (r – MIN) / (MAX – MIN) * (2^N – 1)

где g – целочисленные двоичные гены,

r – эквивалент генов в формате с плавающей запятой.

Хромосомы в формате с плавающей запятой, создаются при помощи размещения закодированных параметров один за другим.

Если сравнивать эти два способа представления, то более хорошие результаты дает вариант представления в двоичном формате (особенно, при использовании кодов Грея). Правда, в этом случае приходиться мириться с постоянным кодированием/декодированием параметров.

В общем, генетический алгоритм работает следующим образом. В первом поколении все хромосомы генерируются случайно. Определяется их «полезность». Начиная с этой точки, генетический алгоритм может начинать генерировать новую популяцию. Обычно, размер популяции постоянен.

Репродукция состоит из четырех шагов:

- селекции

и трех генетических операторов (порядок применения не важен)

- кроссовер
- мутация
- инверсия

Кроссовер является наиболее важным генетическим оператором. Он генерирует новую хромосому, объединяя генетический материал двух родительских. Существует несколько вариантов кроссовера. Наиболее простым является одноточечный. В этом варианте просто берутся две хромосомы, и перерезаются в случайно выбранной точке. Результирующая

хромосома получается из начала одной и конца другой родительских хромосом.

001100101110010|11000 ***110101101101000|11100***	-------->	001100101110010***11100***

Мутация представляет собой случайное изменение хромосомы (обычно простым изменением состояния одного из битов на противоположное).

Данный оператор позволяет более быстро находить генетический алгоритм локальные экстремумы с одной стороны, и позволяет «перескочить» на другой локальный экстремум с другой.

00110010111001011000	-------->	00110010111001111000

Инверсия инвертирует (изменяет) порядок бит в хромосоме путем циклической перестановки (случайное количество раз). Многие модификации генетический алгоритм обходятся без данного генетического оператора.

00110010111001011000	-------->	11000001100101110010

Очень важно понять, за счет чего генетический алгоритм на несколько порядков превосходит по быстроте случайный поиск во многих задачах? Дело здесь видимо в том, что большинство систем имеют довольно независимые подсистемы. Вследствие этого, при обмене генетическим материалом часто может встретиться ситуация, когда от каждого из родителей берутся гены, соответствующие наиболее удачному варианту определенной подсистемы (остальные «уродцы» постепенно вымирают).

Другими словами, генетический алгоритм позволяет накапливать удачные решения для систем, состоящих из относительно независимых подсистем (большинство современных сложных технических систем, и все известные живые организмы). Соответственно можно предсказать и когда генетический алгоритм скорее всего даст сбой (или, по крайней мере, не покажет особых преимуществ перед методом Монте-Карло) — системы, которые сложно разбить на подсистемы (узлы, модули), а так же в случае неудачного порядка расположения генов (рядом расположены параметры, относящиеся к различным подсистемам), при котором преимущества обмена генетическим материалом сводятся к нулю. Последнее замечание несколько ослабляется в системах с диплоидным (двойным) генетическим набором.

Данные, которые закодированы в генотипе, могут представлять собой команды какой-либо виртуальной машины. В таком случае мы можем говорить об эволюционном или генетическом программировании.

Список литературы

1. Шукович Г. Применение генетических алгоритмов и систем генерирующих графов для создания модулярных нейросетей // Программирование, 2002, № 1, с. 13-20.

2. Эшби У.Р. Введение в кибернетику. 3-е изд. М. URSS. КомКнига, 2006.