данный генетический алгоритм позволяет выходить из локальных оптимумов. Спецификой ГА является то, что он дает множество решений задачи, что не позволяет сделать ни один другой алгоритм. Получив такое множество решений можно выбрать наилучшее (решение, дающее минимальное значение целевой функции). Изменяя параметры ГА алгоритма, для каждой конкретной задачи можно улучшить полученный результат. При этом для данного ГА не имеет значения порядок задачи (размер исходной транспортной матрицы) с таким же успехом он может решать задачи больших порядков, ограничиваясь лишь оперативной памятью вычислительной машины. К тому же, хотя данный ГА предназначен для решения линейной ТЗ, при небольшой модификации (в процедуре вычисления целевой функции) его можно применить и для решения нелинейной ТЗ, т.к. он не накладывает никаких ограничений на целевую функцию. Все вышесказанное позволяет сделать вывод, что данный ГА перспективен для решения ТЗ большой размерности, а также ТЗ с нелинейной целевой функцией. При этом путем распараллеливания вычислений, позволяет получать множество независимых друг от друга решений.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Вентцель Е.С.* Исследование операций: задачи, принципы, методология. – М.: Наука. 1988. – 208 с.
2. *Вентцель Е.С.* Исследование операций. М., «Советское радио», 1972, 552 с.
3. *Хэмди А. Таха*. Введение в исследование операций. 6-е издание.: Пер. с англ. – М.: Издательский дом «Вильямс», 2001. – 912 с.
4. Genetic Algorithms + data structures = evolution programs Zbignew Michalewicz – 3rd ed. 1996. – 387 p.
5. Генетические алгоритмы : Учебное пособие. Под ред. В. М. Курейчика. – Ростов-на-Дону: ООО «Ростиздат», 2004. – 400с
6. *Курейчик В.М.* Генетические алгоритмы и их применение. – Таганрог: Изд-во ТРТУ, 2002. – 242 с.
7. *Д. Кнут*. Искусство программирования для ЭВМ т.3. – М.: Мир. 1978.

М.А. Бакало

ВАРИАНТЫ ПРЕДСТАВЛЕНИЯ РАЗЛИЧНЫХ ТИПОВ ДАННЫХ, ИХ КОДИРОВАНИЕ И ДЕКОДИРОВАНИЕ В ВИДЕ ХРОМОСОМ*

Введение. Разработка генетического алгоритма включает три основных компонента: разработка структуры, принципов кодирования и декодирования хромосомы; разработка основных генетических операторов; разработка общей структуры генетического поиска [1-4].

Очевидно, что решение вопроса о способе представления различных типов данных в виде кода хромосомы во многом определяет содержание отдельных операторов и генетического алгоритма в целом. Кроме того, правильный выбор представления, способов кодирования и декодирования обуславливает не только пространственную и временную сложность алгоритма, но и его эффективность.

Авторами проведено исследование способов представления различной информации в виде кода хромосом, применяемых авторами различных генетических алгоритмов. Результаты проведенной работы, обобщены при построении следующей классификации:

* Работа выполнена при частичной финансовой поддержке программы развития научного потенциала высшей школы 2006-2008 годы (проекты РНП.2.1.2.2238, РНП 2.1.2.3193).

1. Кодирование последовательности неповторяющихся чисел. При решении различных задач решение нередко может быть представлено в виде последовательности неповторяющихся чисел. В терминах дискретной математики такое решение является **перестановкой**. Допустим, что необходимо закодировать следующую последовательность чисел, которая может интерпретироваться, например, как последовательность обхода вершин графа.

1	**4**	**3**	**2**	**5**

♦ **"Прямое" кодирование порядка элементов**

В данном случае используется ***числовая негомологичная хромосома***, в которой **локус** – позиция элемента (вершины) в последовательности, **аллель** – значение элемента в данной позиции.

1	4	3	2	5

♦ **Кодирование позиций элементов в последовательности**

В данном случае используется ***числовая негомологичная хромосома***, в которой **локус** – элемент последовательности, **аллель** – позиция элемента (вершины) в последовательности.

1	4	3	2	5

Процесс декодирования данной хромосомы можно описать следующим образом: элемент со значением «1» ставится в первую позицию, поскольку в первом локусе стоит аллель со значением «1»; элемент «2» – в четверную позицию; элемент «3» – в третью; элемент «4» – во вторую; элемент «5» – в пятую.

♦ **Кодирование с последующим упорядочиванием**

В данном случае используется ***числовая гомологичная хромосома***, в которой **локус** – элемент последовательности, **аллель** – некоторое значение в соответствии с которым элементы упорядочиваются при декодировании. При этом диапазон допустимых значений аллелей может быть много шире числа элементов последовательности. Например, при упорядочивании по возрастанию, приведенная выше последовательность может быть закодирована следующим образом:

1	8	5	2	9

Процесс декодирования данной хромосомы можно описать следующим образом: локусу «1» соответствует аллель «1»; локусу «2» – аллель «8»; локусу «3» – аллель «5»; локусу «4» – аллель «2»; локусу «5» – аллель «9». Аллели упорядочивается по возрастанию, а затем заменяются номерами соответствующих им локусов:

Хромосома:	1	8	5	2	9
Упорядоченная последовательность аллелей:	1	2	5	8	9
Декодированная последовательность:	**1**	**4**	**3**	**2**	**5**

А при упорядочивании по убыванию та же последовательность может быть представлена следующей хромосомой:

11	3	6	7	2

Следует отметить, что подобный способ кодирования не устанавливает взаимнооднозначного соответствия фенотипа и генотипа особи, т.е. одна и та же последовательность может быть представлена несколькими разными хромосомами, равно как и одна хромосома может быть декодирована несколькими разными способами, в случае совпадения значений разных генов.

2. Кодирование множества чисел, допускающего многократное вхождение одинаковых элементов. Достаточно распространены задачи, решение которых представляет собой набор некоторых параметров. При этом области допустимых значений указанных параметров могут перекрываться или совпадать. Примером такой задачи будет являться аппроксимация сложной функции полиномом, решение которой – набор коэффициентов перед слагаемыми соответствующего полинома.

Допустим, что необходимо закодировать следующее множество с повторениями:

1	1	2	4	6	1	2	5	4

♦ **Кодирование без учета порядка элементов во множестве**

Данный подход предполагает, что порядок следования элементов во множестве не учитывается, кодируемая информация – количество повторений того, или иного элемента. При таком подходе можно использовать ***числовую гомологичную хромосому***, в которой длина хромосомы совпадает с числом допустимых элементов, **локус** – элемент множества, **аллель** – число вхождений данного элемента в кодируемое множество. Следует обратить внимание, что сумма аллелей хромосомы должна совпадать с мощностью кодируемого множества. Данное ограничение потребует разработки модифицированных ГО. Например, если элементы [1,2, ... 9], то приведенное выше множество будет кодироваться следующей хромосомой:

3	2	0	2	1	1	0	0	0

Следует отметить чрезмерный расход памяти при данном способе кодирования множеств, элементы которых могут принимать значение в широком диапазоне значений. Например, приведенная выше последовательность в случае широкого диапазона допустимых значений [1,2, … 100], будет закодирована вектором из 100 элементов, при этом лишь 5 их них будут иметь ненулевые значения. Устранить этот недостаток можно, применяя ***векторные хромосомы переменной длины***, в которых **аллель** – кортеж длиной два, первый элемент которого соответствует «элементу последовательности», а второй – кратности его вхождения в последовательность. Векторное представление приведенного выше примера будет следующим:

<1; 3>	<2; 2>	<4; 2>	<5; 1>	<6; 1>

Существенным недостатком второго подхода является невозможность применения стандартных генетических операторов ввиду непостоянной длины хромосомы.

♦ **“Прямое ” кодирование множества (с учетом порядка элементов)**

В данном случае используется ***числовая гомологичная хромосома***, в которой **локус** – позиция элемента во множестве, **аллель** – элемент из допустимого диапазона в данной позиции:

1	1	2	4	6	1	2	5	4

3. Кодирование многомерной последовательности чисел. Широко распространены задачи, данные которых организованы в виде матрицы. Такова, например, задача размещения элементов на подложке кристалла, либо задача покрытия схемы стандартными ячейками. Рассмотрим подходы к кодированию многомерной последовательности на примере плоской (двумерной) матрицы. Отдельно будут оговариваться случаи, когда предложенный метод неприменим, для кодирования последовательности большей размерности, или имеет какие-либо особенности. Пусть мы имеем следующее распределение чисел на плоскости:

1	6	3
4	8	2
5	9	7

♦ **Линеаризация n-мерного пространства (преобразование в одномерную последовательность)**

Основная идея состоит в том, чтобы пронумеровать все ячейки в соответствии с ранее заданным правилом (по строкам, по столбцам, вдоль главной/побочной диагонали, по спирали и т.п.). Например, нумерация позиций по правой спирали от центра к краям:

7	8	9
6	1	2
5	4	3

Предложенный подход позволяет развернуть приведенное выше размещение (двумерную последовательность) в одномерную последовательность:

8	2	7	9	5	4	1	6	3

После подобного преобразования можно использовать любой метод кодирования одномерной последовательности предложенный в **п.1** для последовательности неповторяющихся чисел, и в **п.2** – для множества с повторениями.

♦ **Разложение по векторам**

Зачастую, подгруппы данных многомерной последовательности могут быть логически объедены. Например, каждая «строка» матрицы может интерпретироваться как набор характеристик элементов системы.

В этом случае такие логически связанные группы данных удобно представить как единичный элемент хромосомы – ген. При таком подходе многомерная последовательность кодируется с использование ***векторной хромосомы***, в которой **локус** соответствует логически связанной группе данных (строка, столбец матрицы; плоскость и т.п.), а **аллель** – кортежу вида $<x_1; x_2; \dots x_n>$, соответствующего кодируемому подмножеству данных.

Разложение по строкам приведенной выше матрицы позволит получить следующую хромосому:

<1; 6; 3>	<4; 8; 2>	<5; 9; 7>

♦ **Координатное представление**

Данный подход предполагает, что элементы, размещаемые в n-мерном пространстве, не повторяются. Предлагается использовать ***векторную хромосому***, в которой **локус** соответствует номеру элемента, а **аллель** – кортежу вида $<x_1; x_2; \dots x_n>$ - определяющему “координаты” элемента.

<1; 3>	<3; 2>	<3; 3>	<1; 2>	<1; 1>	<2; 3>	<3; 1>	<2; 2>	<2; 1>

4. Кодирование действительных чисел. Существует класс задач, решение которых представляется действительными числами или их последовательностью. Например, моделирование функции, имеющей поверхность сложной формы, или подбор коэффициентов связей нейронной сети. Основная сложность при решении таких задач заключается в том, что существующие генетические операторы ориентированны на данные, имеющие дискретную область допустимых значений, тогда как действительные числа определены на непрерывном интервале.

Пусть необходимо закодировать действительное число ***d***, определенное на интервале **[*a, b*]**, с заданной точностью Δ.

♦ **Дискретизации диапазона допустимых значений**

Данный подход состоит в том, что диапазон допустимых значений **[*a, b*]** разбивается, на отрезки длиной Δ, которые нумеруются, начиная с нуля. Обозначим за ***k*** – номер отрезка, в который попадает число ***d***, $\boldsymbol{k_{max}}$ – максимальное число отрезков:

$$k_{\max} = \frac{b-a}{\Delta}.$$

Тогда преобразование ***d*** → ***k*** будет производиться по следующей формуле:

$$k = INT\left(\frac{d-a}{\Delta}\right).$$

При декодировании ***k*** → ***d*** возможно несколько вариантов: ***d*** соответствует левый (правый) конец отрезка ***k***; ***d*** соответствует середине отрезка ***k***; ***d*** генерируется случайным образом в диапазоне $\left[k \cdot \Delta; (k+1) \cdot \Delta\right]$.

♦ **Разложение в дробь**

При таком подходе действительное число ***d*** представляется как отношение двух целых чисел ***A, R***

$$d \to \frac{A}{R}, \text{ при этом } \left|d - \frac{A}{R}\right| \le \Delta.$$

Например, ***d = 46,168*** может быть представлено в виде ***<5771; 125>***.

♦ **Представление в виде числа с плавающей запятой**

Данный подход широко применяется в компьютерной технике и состоит в том, что действительное число ***d***, записывается в виде

$$F = p_1 C + p_2 D,\ где\ \ p_1, p_2 \in [0,1],\ p_1 = 1 - p_2$$

где ***m*** – мантисса, ***N*** – основание, ***p*** – порядок, являющиеся целыми числами. Таким образом, действительное число ***d*** может быть представлено кортежем **<*m, p*>**.

Например, ***d = 46,168*** при основании ***N = 10*** будет представлено в виде $\boldsymbol{46168 \cdot 10^{-3}}$ и будет кодироваться парой чисел ***<46168; -3>***.

Заключение. Результаты работы по обобщению и классификации способов представления информации в виде генетического кода, опубликованные в настоящей статье, являются частью исследований автора, проведенных при подготовке диссертации. Приведенную классификацию нельзя считать всеобъемлющей, однако, с помощью предложенных подходов можно охватить широкий класс задач.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Holland John H., Adaptation in Natural and Artificial Systems: An Introductory Analysis with Application to Biology, Control, and Artificial Intelligence. USA: University of Michigan, 1975.
2. Goldberg David E. Genetic Algorithms in Search, Optimization and Machine Learning. USA: Addison-Wesley Publishing Company, Inc., 1989.
3. Handbook of Genetic Algorithms. Edited by Lawrence Davis. USA: Van Nostrand Reinhold, New York, 1991.
4. Гладков Л.А, Курейчик В.В., Курейчик В.М. Генетические алгоритмы. – М.: Физматлит, 2006.