

*Серета А.А. — рецензент Рогоза В.С.*

*УНК “Институт прикладного системного анализа” НТУУ “КПИ”, Киев, Украина*

## Реализация операции скрещивания генетического алгоритма при решении задачи коммивояжера

Задача коммивояжера является одной из знаменитых задач теории комбинаторики. Ее суть формулируется следующим образом: коммивояжер (бродячий торговец) должен обойти все заданные города и вернуться в исходную позицию, при этом он не может посещать один город два или более раз. Главной целью задачи коммивояжера является построение оптимального, с точки зрения расстояния, затрат на топливо или времени, маршрута, при этом необходимые для оптимизации данные (расстояние между любыми двумя городами, расход топлива или время перемещения между ними) заданы либо численно, либо могут быть вычислены.

Для решения данной задачи используют множество различных алгоритмов, среди которых присутствует класс генетических алгоритмов.

Генетический алгоритм - это простая модель эволюции в природе, реализованная в виде компьютерной программы. В нем используются как аналог механизма генетического наследования, так и аналог естественного отбора. При этом сохраняется биологическая терминология в упрощенном виде.

Каждое возможное решение, которое дает алгоритм, называют «хромосомой». В процессе работы поддерживается так называемое «поколение» решений – набор хромосом. Работа алгоритма состоит в смене поколений – в их регенерации. Для выполнения такой смены используют операции «мутации» и «скрещивания». Операция мутации заключается в небольшом изменении хромосомы, а операция скрещивания - в слиянии двух или более хромосом в одну.

Пусть 2 хромосомы представлены в виде:  $X = \{x_1, x_2, \dots, x_n\}$  и  $Y = \{y_1, y_2, \dots, y_n\}$ , где  $\{x_1, \dots, x_n, y_1, \dots, y_n\}$  – факторы оптимизации,  $n$  – количество звеньев в хромосоме. В таком случае стандартная операция скрещивания даст следующих наследников:  $X_1 = \{x_1, \dots, x_K, y_{(K+1)}, \dots, y_n\}$  и  $Y_1 = \{y_1, \dots, y_K, x_{(K+1)}, \dots, x_n\}$ , где  $n$  – количество звеньев в хромосоме,  $K$  – “точка разрыва”,  $1 < K < n - 1$  [1]. При этом хромосомы  $X, Y$  принято называть родительскими,  $X_1, Y_1$  – дочерними.

Если решать задачу коммивояжера с помощью генетических алгоритмов, возникает несколько проблем. Две основных – представление данных и реализация операций мутации и скрещивания.

Пусть каждый город из списка тех, которые необходимо посетить, будет представлен порядковым числом. В таком случае, каждая хромосома будет содержать список чисел от 1 до  $N$ , расставленных в некотором порядке, где  $N$  – количество городов. Таким образом, переменные  $x_1 \dots x_n$  хромосомы  $X$  будут содержать порядковые числа, независимые друг от друга. В целом, хромосома будет содержать цепочки чисел, представляющие порядок обхода городов коммивояжером. Цель оптимизации - нахождение такой цепочки, при которой расстояние обхода всех городов (либо затраты на обход) минимально.

Однако, при таком представлении данных, реализация стандартных операций скрещивания и мутации невозможна, так как каждое число, представляющее город, может быть использовано только один раз и каждая хромосома содержит в себе список чисел от 1 до  $N$ .

При реализации операции скрещивания, предлагается подход «упорядоченного скрещивания», при котором города в левой части новой хромосомы повторяют порядок городов первой родительской хромосомы, а в правой части располагаются в порядке, в котором они расположены во второй родительской хромосоме. То есть первый наследник будет выглядеть следующим образом:  $X_1 = \{x_1, \dots, x_K, z_{(K+1)}, \dots, z_n\}$ , где каждая  $z$  из  $z_{(K+1)} - z_n$  принимает одно из значений значений  $x_{(K+1)} - x_n$ , в зависимости от того, в каком порядке расположены такие же значения в хромосоме  $Y$ . Второй наследник - аналогично.

**Литература.** 1. Neuroproject, <http://www.neuroproject.ru/>.