

*О.В. СЕРАЯ*, канд. техн. наук, НТУ "ХПИ"

## **ПРИМЕНЕНИЕ ПРОЦЕДУРЫ КЛАСТЕРИЗАЦИИ ПРИ РЕШЕНИИ ЗАДАЧИ КОММИВОЯЖЕРА ВЫСОКОЙ РАЗМЕРНОСТИ С ИСПОЛЬЗОВАНИЕМ ГЕНЕТИЧЕСКОГО АЛГОРИТМА**

Проведено аналіз відомих методів розв'язання задачі комівояжера. Для ефективного вирішення цієї проблеми великої розмірності запропоновано використання генетичного алгоритму з попередньою кластеризацією пунктів призначення.

The analysis of the known methods decision of traveling salesman task is conducted. For the effective decision of this largeness task is suggested to use a genetic algorithm with preliminary procedure of cluster analysis of setting points.

**Введение. Анализ методов решения задачи коммивояжера.** Одной из важных задач транспортной логистики является так называемая задача коммивояжера. Суть задачи проста. Бродячий торговец должен посетить  $n$  пунктов назначения, переезжая из одного пункта назначения в другой. Маршрут должен быть составлен так, чтобы он охватывал все пункты назначения, но не проходил через один и тот же пункт дважды. Расстояния между пунктами считаются известными. Задача состоит в отыскании кратчайшего маршрута.

Проблема коммивояжера может быть успешно решена непосредственно, если количество пунктов назначения невелико. Принципиальной особенностью задачи коммивояжера является то, что при увеличении числа пунктов вычислительная сложность ее решения существенно и быстро возрастает, и для некоторой размерности задачи она уже не может быть решена за приемлемое время (или будет решена, но не оптимально). Известные методы решения: комбинаторный перебор, использование теории графов, метод ветвей и границ, аналитический.

Комбинаторный способ решения состоит в переборе всех возможных вариантов решения и выборе лучшего. Количество всех просматриваемых решений равно  $n!$ , где  $n$  – количество пунктов назначения. В реальной задаче число объектов, которые необходимо посетить, достаточно велико (порядка 100). В этом случае объем пространства поиска (число вариантов решений) приближается по порядку величины к числу атомов в видимой части Вселенной. Поэтому этот способ принципиально неэффективен с точки зрения вычислительной сложности. Достоинство этого метода в том, что полученное с его помощью решение однозначно является оптимальным.

Задача коммивояжера в терминах теории графов формулируется следующим образом [1]. Пунктам назначения соответствуют вершины графа, а его дугам приписываются расстояния между соответствующими вершинами.

Задача состоит в отыскании кратчайшего замкнутого контура, проходящего через все вершины графа, причем каждая вершина используется один раз.

Эта задача была впервые исследована Део и Хакими, давшими ее формулировку на языке линейного программирования. Для полного графа с  $n$  вершинами эта формулировка содержит  $n(n+1)$  переменных и  $n(n+3)/2+1$  ограничений. Кроме того, вводится большое число ограничений, обеспечивающих связность маршрута, которые затруднительно сформулировать явно. Этот метод, безусловно, более эффективен, чем полный перебор, однако и его применение реально лишь для задач очень небольшой размерности ( $n < 10-12$ ).

Другой подход, использующий идеи линейного программирования [2, 3], реализуется следующим образом. Введем набор булевых переменных

$$x_{ij} = \begin{cases} 1, & \text{если коммивояжер из пункта } i \text{ переезжает в пункт } j, \\ 0 & \text{в противном случае.} \end{cases}$$

Зададим матрицу  $(C_{ij})$  расстояний между пунктами. Тогда

$$L(X) = \sum_{i=1}^n \sum_{j=1}^n C_{ij} x_{ij} \quad (1)$$

есть длина маршрута для выбранного плана переездов  $X$ . На матрицу  $X = (x_{ij})$  должны быть наложены ограничения

$$\sum_{i=1}^n x_{ij} = 1, \quad j = 1, 2, \dots, n; \quad (2)$$

$$\sum_{j=1}^n x_{ij} = 1, \quad i = 1, 2, \dots, n. \quad (3)$$

Система ограничений (2) обеспечивает построение маршрута, в котором въезд в каждый пункт осуществляется один раз. Система ограничений (3), в свою очередь, обеспечивает построение маршрута, в котором выезд из каждого пункта осуществляется один раз. К сожалению, эти ограничения оказываются недостаточными, так как среди допускаемых ими решений имеются маршруты, не образующие полный цикл, включающий все пункта. Можно показать, что устранение всех подциклов обеспечивается при добавлении системы ограничений

$$u_i - u_j + nx_{ij} \leq n - 1, \quad i = 2, 3, \dots, n, \quad j = 2, 3, \dots, n.$$

Таким образом, исходная задача сведена к задаче линейного целочисленного программирования. Для решения линейных целочисленных задач существуют специальные алгоритмы, в частности, алгоритм Гомори [4].

Применение методов целочисленного линейного программирования обеспечивает принципиальную возможность решения задачи коммивояжера. Однако следует заметить, что и здесь количество переменных в задаче очень быстро растет с увеличением размерности исходной задачи, что очень резко ограничивает возможности практического применения этого метода.

Одним из наиболее эффективных методов решения задачи коммивояжера является метод «ветвей и границ» [5 – 7]. Метод основан на итерационном отсечении бесперспективных вариантов построения маршрута. Вычислительную трудоемкость (продолжительность) решения задачи коммивояжера этим методом можно оценить в условных единицах (одна единица соответствует перебору  $10^6$  вариантов). Приведем экспериментально полученную таблицу трудоемкости решения задачи от ее размерности.

Таблица 1

Зависимость продолжительности решения от размерности задачи  
(в условных единицах)

Кол-во пунктов	3	4	5	6	7	8
Продолжительность, у.е.	0.14	0.19	0.26	0.34	0.73	1.41
Количество пунктов	9	10	11	12	13	
Продолжительность, у.е.	6.67	24.24	167.88	774.25	2125.5	

Экспоненциальный характер зависимости продолжительности решения задачи от ее размерности позволяет использовать для ее описания модель

$$T = a_0 e^{a_1 n} \quad (4)$$

Параметры модели  $(a_0, a_1)$  легко отыскиваются. Логарифмируя (4) и переходя к новым переменным, получим

$$y = \alpha_0 + \alpha_1 n; \quad y = \ln T; \quad \alpha_0 = \ln a_0; \quad \alpha_1 = a_1. \quad (5)$$

Теперь формируем функционал наименьших квадратов

$$J = (HA - Y)^T (HA - Y);$$

$$H = \begin{pmatrix} 1 & n_1 \\ 1 & n_2 \\ \dots & \dots \\ 1 & n_m \end{pmatrix}; \quad A = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix}; \quad Y = \begin{pmatrix} \ln T_1 \\ \ln T_2 \\ \dots \\ \ln T_m \end{pmatrix},$$

минимизация которого дает вектор  $\hat{A}$  оценок параметров уравнения регрессии (5):

$$\hat{A} = (H^T H)^{-1} H^T Y.$$

Для исходных данных, помещенных в табл. 1, имеем  $\hat{A} = \begin{pmatrix} -6.76 \\ 1.07 \end{pmatrix}$ , откуда

$$T(n) = 0.0012 e^{1.07n}.$$

Ясно, что и этот метод не может быть использован для задач реальной размерности.

**Цель работы:** исследовать возможность решения задачи коммивояжера с помощью интенсивно развивающейся технологии генетических алгоритмов [8 – 10].

**Основные результаты.** Как известно, эффективность генетического алгоритма решения конкретной задачи определяется, прежде всего, качеством процедуры формирования популяции. Для многих задач, например, задач безусловной оптимизации, разработаны простейшие и быстрые способы формирования жизнеспособных особей. К сожалению, в задаче коммивояжера дело обстоит не так.

Известные процедуры в результате применения стандартных операторов (рекомбинация, мутация и т.д.) приводят к получению особей, значительная часть которых – «инвалиды», соответствующие маршрутам, не удовлетворяют требованиям. В связи с этим существенная часть времени работы алгоритма расходуется непроизводительно. Тем не менее, генетические алгоритмы реализуют, по-видимому, наиболее эффективный подход к решению комбинаторных задач. В табл. 2 приведены экспериментально полученные данные относительно продолжительности решения задачи коммивояжера разной размерности.

Таблица 2

Зависимость продолжительности решения от размерности задачи  
(в условных единицах)

Количество пунктов	10	18	26	34
Продолжительность, у.е.	1,02	11,1	31,2	56,3
Количество пунктов	42	50	58	66
Продолжительность, у.е.	126,7	205,3	275,6	396,8

Для описания соответствующей зависимости введем степенную модель

$$T(n) = b_0 n^{\beta_1}. \quad (6)$$

Логарифмируя (6) и переходя к новым переменным, получим

$$y = \beta_0 + \beta_1 z; \quad y = \ln T; \quad z = \ln n; \quad \beta_0 = \ln b_0; \quad \beta_1 = b_1. \quad (7)$$

Формируем функционал наименьших квадратов

$$J_1 = (H_1 B - Y)^T (H_1 B - Y); \quad (8)$$

$$H_1 = \begin{pmatrix} 1 & \ln(n_1) \\ 1 & \ln(n_2) \\ \dots & \dots \\ 1 & \ln(n_m) \end{pmatrix}; B = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}; Y = \begin{pmatrix} \ln T_1 \\ \ln T_2 \\ \dots \\ \ln T_m \end{pmatrix}.$$

Минимизация (8) дает вектор  $\hat{B}$  оценок параметров регрессии (7):

$$\hat{B} = (H_1^T H_1)^{-1} H_1^T Y.$$

Для исходных данных табл. 2 имеем

$$T(n) = 0.0014x^{3.025}.$$

Таким образом, генетический алгоритм решения задачи коммивояжера на много порядков более эффективен, нежели любой другой известный алгоритм решения *NP*-полных задач. Предельная размерность задачи для этого алгоритма 60–70 пунктов назначения. Вместе с тем, приближенное решение этой задачи очень хорошего качества возможно получить даже для случая существенно более высокой размерности в результате применения комбинации «кластеризация – генетический алгоритм». Предлагаемая при этом технология решения задачи коммивояжера является четырехэтапной.

На первом этапе все множество пунктов назначения с использованием какого-либо алгоритма кластеризации разделяется на совокупность кластеров  $E_1 = \{i_{11}, i_{12}, \dots, i_{1n_1}\}$ ,  $E_2 = \{i_{21}, i_{22}, \dots, i_{2n_2}\}$ , ...,  $E_m = \{i_{m1}, i_{m2}, \dots, i_{mn_m}\}$ . Для каждого кластера отыскиваются координаты центров тяжести, определяющие в совокупности набор точек  $M_1 = (X_{ц1}, Y_{ц1})$ ,  $M_2 = (X_{ц2}, Y_{ц2})$ , ...,  $M_m = (X_{цm}, Y_{цm})$ .

На втором этапе для точек  $M_1, M_2, \dots, M_m$  с использованием генетического алгоритма строится кратчайший маршрут их обхода  $\{M_{j_1}, M_{j_2}, \dots, M_{j_m}\}$ .

На следующем этапе для каждой пары «соседних» кластеров отыскивается пункт выхода из предыдущего кластера и пункт входа в последующий. Пусть, например, номер предыдущего кластера –  $k_1$ , а номер следующего за ним в порядке обхода –  $k_2$ . Тогда пара «номер пункта  $I_{k_1}^{(вых)}$  выхода их кластера  $k_1$  и номер пункта  $I_{k_2}^{(вх)}$  входа в кластер  $k_2$ » определяется из соотношения

$$(I_{k_1}^{(вых)}, I_{k_2}^{(вх)}) = \arg \min_{\substack{s_1 \in M_{k_1} \\ s_2 \in M_{k_2}}} \{R(i_{k_1 s_1}, i_{k_2 s_2})\},$$

где  $R(i_{k_1s_1}, i_{k_2s_2})$  – расстояние между  $s_1$ -м пунктом из кластера  $k_1$  и  $s_2$ -м пунктом из кластера  $k_2$ .

Наконец, на последнем, четвертом этапе решается последовательность задач коммивояжера для каждого из кластеров  $E_1, E_2, \dots, E_m$  с учетом полученных на предыдущем этапе начального и конечного пунктов. В результате будет получена последовательность маршрутов

$$\{i_{j_1,1}^{(BX)}, i_{j_1,1}, \dots, i_{j_1,1}^{(BIX)}\}, \{i_{j_2,1}^{(BX)}, i_{j_2,2}, \dots, i_{j_2,2}^{(BIX)}\}, \dots, \{i_{j_m,1}^{(BX)}, i_{j_m,2}, \dots, i_{j_m,m}^{(BIX)}\},$$

образующих искомый замкнутый маршрут. Эффективность этого приближенного алгоритма поразительно высока. Общая продолжительность его работы определяется выражением

$$\tilde{T}(n) = T(m) + \sum_{k=1}^m T(n_k).$$

Оценку  $\eta$  порядка выигрыша в продолжительности работы, получаемого при использовании приближенного алгоритма, можно определить, если положить, что  $n$  пунктов назначения разбиты на  $\sqrt{n}$  кластеров по  $\sqrt{n}$  пунктов в каждом. Тогда  $\tilde{T}(n) = (\sqrt{n} + 1)T(\sqrt{n})$ .

$$\text{При этом } \eta = \frac{T(n)}{\tilde{T}(n)} = \frac{n^{3.025}}{(\sqrt{n} + 1)(\sqrt{n})^{3.025}} = \frac{(\sqrt{n})^{3.025}}{(\sqrt{n} + 1)} \approx n.$$

**Выводы.** Таким образом, для решения задачи коммивояжера высокой размерности предложена многоэтапная процедура, использующая предварительную кластеризацию множества пунктов назначения. Проведена оценка выигрыша в продолжительности решения задачи. Показано, что разработанная технология может быть эффективно использована при решении реальных задач коммивояжера.

**Список литературы:** 1. Берж К. Теория графов и ее применение. – М.: ИИЛ, 1962. – 319 с. 2. Корбут А.А., Финкельштейн Ю.Ю. Дискретное программирование. – М.: Наука, 1969. – 392 с. 3. Ковалев М.М. Дискретная оптимизация. – Минск: Изд. Бел. ун., 1977. – 238 с. 4. Гомори Р.Е. Бомоль У. Дж. Целочисленное программирование и оценки. – Новосибирск: СО АН СССР, 1962. – 198 с. 5. Литтл Дж., Мурти К., Суинни Д., Кэрел К. Алгоритм для решения задачи о коммивояжере // Экономика и математические методы. – 1965. – Т. 1. – № 1. – С. 34 – 48. 6. Корбут А.А., Сигал И.Х., Финкельштейн Ю.Ю. Метод ветвей и границ // Math. Operat. and Statist. Set. Optim. – 1977. – Т. 8. – № 2. – С. 283 – 290. 7. Михалевич В.С., Трубин В.А., Шор Н.З. Оптимизационные задачи производственно-транспортного планирования. – М.: Наука, 1986. – 259 с. 8. Holland J.H. Adaptation in Natural and Artificial Systems. – University of Michigan Press, 1975. – 98 p. 9. Лысенко Ю.Г., Иванов Н.Н., Минц А.Ю. Нейронные сети и генетические алгоритмы. – Донецк: ООО «Юго-Восток», 2003. – 265 с. 10. Koza J.R. Genetic Programming. – MIT Press, 1994. – 134 p.

Поступила в редакцию 3.04.2006