$$\Delta = \frac{1}{n}(R_{max} - R_{min}) \tag{3}$$

где n – количество кругов регрессии;

R_{min} – минимальное значение радиуса круга регрессии;

R_{max} – максимальное значение радиуса круга регрессии;

Δ – шаг приращения.

По полученным результатам строиться таблица, в которой определяются диапазон радиусов и количество таких кругов.

Таблица 1.

Радиус	$r_1 = R_{min}$	$r_2 = r_1 + \Delta$	…	$r_n = R_{max}$
Частота	m_1	m_2	…	m_n

По таблице строится гистограмма, форма которой описывает объект или процесс, идентифицируя его.

4. СВОЙСТВА

Теорема: линейная регрессия проходит через центр круга регрессии.

Доказательство: по определению центр круга регрессии находится по формулам (1).

В общем случае любая линия на плоскости может быть представлена уравнением [2]:

$$y = a_0 + a_1 x \tag{4}$$

Коэффициенты уравнения (4) или коэффициенты линии регрессии определяются следующими выражениями:

$$a_1 = \left(\sum_{i=1}^{n} x_i y_i - \frac{1}{n}\sum_{i=1}^{n} x_i * \sum_{i=1}^{n} y_i\right) \Bigg/ \left(\sum_{i=1}^{n} x_i^2 - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)^2\right); \tag{5}$$

$$a_0 = \frac{1}{n}\left(\sum_{i=1}^{n} y_i - a_1 * \sum_{i=1}^{n} x_i\right). \tag{6}$$

По формулам (1) перейдем к x_C, y_C.

$$a_1 = \left(\sum_{i=1}^{n} x_i y_i - n x_C y_C\right) \Bigg/ \left(\sum_{i=1}^{n} x_i^2 - n x_C^{\ 2}\right) \tag{7}$$

$$a_0 = y_C - a_1 x_C \tag{8}$$

Формулы (7,8) подставим в уравнение линии (4):

$$y_C = y_C - \frac{\sum_{i=1}^{n} x_i y_i - n x_C y_C}{\sum_{i=1}^{n} x_i^2 - n x_C^2} x_C + \frac{\sum_{i=1}^{n} x_i y_i - n x_C y_C}{\sum_{i=1}^{n} x_i^2 - n x_C^2} x_C$$

$y_C = y_C$ - верно

Теорема доказана.

5. ВЫВОДЫ

Т.о. в работе показан совершенно новый подход к идентификации случайных объектов или процессов. Случайный объект или процесс можно описать не только с помощью линии регрессии, которая характеризуется углом наклона и отсекаемым отрезком на оси ординат, не только с помощью линий высшего порядка, таких как парабола, гипербола и т.д., но и с помощью кругов регрессии. Круги регрессии наиболее полно представляют случайный объект или процесс, т.к. имеют более сильную связь с объектом.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Основные математические формулы: Справочник/ В. Т. Воднев, А. Ф. Наумович, Н. Ф. Наумович; Под ред. Ю. С. Богданова. – 3-е изд., перераб. и доп. – Мн.: Выш. шк., 1995.-380 с.:ил.
2. Ефимова М.Р., Петрова Е.В., Румянцев В.Н. Общая теория статистики. – М.: ИНФРА-М, 1997.

УДК 681.3

Павлюкович С.В., Михалюк В.В., Шуть В.Н.

ПРИБЛИЖЕННЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧИ КОММИВОЯЖЕРА – МЕТОД КРИСТАЛЛИЗАЦИИ

Целью настоящей работы является обзор методов решения задачи коммивояжера и разработка нового метода приближенного решения этой задачи.

Существуют несколько формулировок задачи коммивояжера. Одна из них – комбинаторная формулировка.

Пусть $s(1)$, $s(2)$,…, $s(n)$ - некоторая перестановка чисел 1, 2, …, n. Если во множестве S_n подстановок вида

$$s = \begin{pmatrix} 1 & 2 & \dots & n \\ s(1) & s(2) & \dots & s(n) \end{pmatrix} \tag{1}$$

выделить множество S_n^1 всех полных циклов, тот задачу коммивояжера можно будет определить следующим образом. Пусть

$$C = \left\| C_{ij} \right\| \tag{2}$$

есть $n \times n$ матрица действительных чисел, называемая впредь матрицей расстояний. Каждой подстановке (1) сопоставим число

$$L_c(s) = \sum_{i=1}^{n} c_{is}(i), \tag{3}$$

называемое длиной подстановки (1). В частности, если подстановка (1) является циклом $\tau = (\tau_1, \tau_2, \dots, \tau_n)$, то ее длина $L_c(\tau) = c_{\tau_1\tau_2} + c_{\tau_2\tau_3} + \dots + c_{\tau_n\tau_1}$ (число n называется циклической длиной цикла τ). Задача коммивояжера заключается в нахождении S_n^1 подстановки (цикла) s_0, для которой $L_c(s_0)$ минимально на множестве длин (3), вычисленных для всех полных циклов

$$S_0 = \min_{s \in S_n^1} \{L_c(s)\}. \tag{4}$$

Павлюкович С.В. *Ассистент каф. вычислительной техники и прикладной математики Брестского государственного технического университета. .E-mail: pasw@tut.by.*
Михалюк В.В. *Аспирант каф. ЭВМиС Брестского государственного технического университета.*
Шуть Василий Николаевич. *К.т.н., доцент кафедры ЭВМиС Брестского государственного технического университета. Беларусь, БГТУ, 224017, г. Брест, ул. Московская, 267.*

Задача поиска подстановки s^0, такой, что

$$S^0 = \max_{s \in S_n^1} \{ L_c(s) \}, \qquad (5)$$

называется задачей коммивояжера на максимум. Определим M - длину подстановки s как

$$M_c(s) = \max_i \{ C_{is(i)} \mid i = \overline{1,n} \}. \qquad (6)$$

Задачу поиска в S_n^1 подстановки минимальной M - длины

$$s_M = \min_{s \in S_n^1} \{ M_c(s) \} \qquad (7)$$

назовем минимаксной задачей коммивояжера [1]. В дальнейшем будет рассматриваться задача на минимум.

Задача коммивояжера относится к классу дискретных экстремальных задач, т. е. задач отыскания экстремума некоторой функции на конечном множестве. Говоря об алгоритмах решения дискретных экстремальных задач, нельзя не касаться вопроса о качестве этих алгоритмов. В настоящее время, повидимому, нет общепринятой точки зрения на критерии сравнительной оценки разных алгоритмов. Однако, не вызывает особых сомнений то, что к числу важнейших характеристик алгоритмов могут быть отнесены такие величины как количество элементарных операций (трудоемкость) и объем памяти, необходимые для реализации алгоритма.

Любая дискретная экстремальная задача на конечном множестве, в принципе, может быть решена целенаправленным перебором конечного числа вариантов (допустимых решений). Однако, как показывает опыт, для большинства известных задач получение точного решения существующими методами можно гарантировать только после просмотра не менее c^m вариантов ($c \ge 2$), где m - длина записи входных данных задачи [2].

В частности, это замечание целиком относится к задаче коммивояжера. Так, в некоторых работах отмечается, что во всех известных точных методах решения этой задачи время счета растет экспоненциально с ростом числа городов, и высказывается предположение о принципиальной невозможности избавиться от этого «демона экспоненциальности».

В свете приведенных фактов поиски эффективных и точных методов решения для многих задач дискретной оптимизации, возникающих на практике, представляются малоперспективными. В этой ситуации мы вынуждены переходить либо к изучению более частных задач и поискам для них малотрудоемких алгоритмов, либо довольствоваться приближенными алгоритмами, т. е. алгоритмами, не гарантирующими точного решения. Ясно, что во втором случае, жертвуя свойством «точности» алгоритма, можно рассчитывать на существенный выигрыш в трудоемкости.

Если в случае точных алгоритмов упомянутых ранее характеристик – трудоемкости и объема памяти – на практике бывает достаточно для грубой оценки качества алгоритма, то для сравнения приближенных алгоритмов необходимо еще иметь определенное представление об отклонениях получаемых посредством данных алгоритмов решений от оптимума.

Приведем классификацию основных методов точного и приближенного решения задачи коммивояжера.

Точные методы решения задачи коммивояжера можно разделить на методы:

1). полного перебора;
2). динамического программирования;
3). ветвей и границ;
4). множителей Лагранжа;
5). отсекающих плоскостей;
6). композитные.

Эффективные алгоритмы точного решения задачи коммивояжера, разработанные в последние годы, как правило, принадлежат к разряду композитных, т. е. содержат в себе элементы нескольких методов[3].

Основным инструментом решения практических задач являются эвристические (приближенные) алгоритмы. Появление эвристик обусловлено, в первую очередь, излишней чувствительностью точных алгоритмов по отношению к специфике задачи и наличию дополнительных условий. Эвристики представляют собой попытку учесть специфику задачи простыми средствами, создать прием, эффективный для решения задач с определенной особенностью [4].

Эвристические алгоритмы решения задачи коммивояжера можно условно разбить на следующие классы:

1). алгоритмы построения тура;
2). монотонные алгоритмы улучшения тура;
3). немонотонные алгоритмы улучшения тура;
4). комбинированные алгоритмы;
5). человеко-машинные и гибридные алгоритмы;
6). алгоритмы декомпозиции и агрегирования решения больших и сверхбольших задач.

Алгоритмы построения тура последовательно включают в тур вершины и дуги (ребра) в соответствии с некоторыми правилами.

Алгоритмы улучшения тура (монотонные и немонотонные) уменьшают длину тура, переставляя вершины в начальном туре.

Комбинированные алгоритмы содержат элементы первых трех классов: улучшение тура, полученного алгоритмом первого класса, алгоритмами классов 1 и 2, улучшение промежуточных подтуров в процессе построения и т. д.

Под гибридной процедурой понимается программный комплекс, в котором на различных этапах вычислительного процесса имеется возможность управляемого выбора на программном уровне одного из нескольких алгоритмов. Если при этом возможно участие человека в диалоговом режиме, то такая процедура называется человеко-машинной.

В алгоритмах декомпозиции и агрегирования наиболее существенными элементами являются разбиение множества вершин на подмножества, решение задачи коммивояжера для подмножеств и затем объединение этих решений в одно.

Разработанный алгоритм кристаллизации для приближенного решения задачи коммивояжера представляет собой комбинированный алгоритм, воплотивший некоторые из вышеописанных идей. Рассмотрим его подробно.

Пусть имеется n городов, расстояние между которыми задается при помощи матрицы расстояний (симметричной или несимметричной) размерностью $n \times n$. Расстояния между городами обозначим как l_{ij} $(i, j = \overline{1,n})$. Для симметричной матрицы расстояний $l_{ij} = l_{ji}$ $(i, j = \overline{1,n})$. Длина пути из города i в город i равна нулю, т. е. $l_{ii} = 0$ ($i = \overline{i,n}$). Рассмотрим алгоритм кристаллизации для симметричной матрицы расстояний. Для несимметричной алгоритм аналогичен.

1). Для удобства вначале все n вершин графа (города) представляются в виде n контуров (петли). Таким образом, на первом шаге алгоритма мы имеем n контуров A_x $(x = \overline{1,n})$. Каждый контур хранит информацию об образующих его ребрах.
2). Выбираются два склеиваемых (объединяемых) контура A_j и A_k.

Таблица 1 – Теоретическое сравнение методов решения задачи коммивояжера.

Название метода	Трудоемкость	Используемая память	Качество решения
Полного перебора	$n!$	n^2	точный
Ветвей и границ	$O(\sim 7n^2)$	$\sim (2n^3)$	точный
Генетический	$O(k \times n)$	$\sim n^2$	Приближенный
Включения ближайшего соседа	$O(n \ln n)$	$\sim n^2$	Приближенный
Кристаллизации	$\sim n^2$	$\sim n^2$	Приближенный

Таблица 2 – Использование памяти различными алгоритмами решения задачи коммивояжера.

Количество точек	Перебора	Ветвей и границ	Кристаллизация	Ветвей и границ приближенный	Включение ближайшего соседа
5	100	1900,26	380	204	124
7	132	5340,1	588	324	228
10	180	32136,5	960	564	444
15	-	-	1740	1124	964
20	-	-	2720	1884	1684
50	-	-	12800	10644	10204
100	-	-	45600	41244	40404
250	-	-	264000	253044	251004
500	-	-	1028000	1006044	1002004
750	-	-	2292000	2259044	2253004
1000	-	-	4056000	4012044	4004004

Таблица 3 – Время работы различных алгоритмов решения задачи коммивояжера.

Количество точек	Перебора	Ветвей и границ	Кристаллизация	Ветвей и границ приближенный	Включение ближайшего соседа
5	0	0,0006	0	0	0
7	0,0012	0,0016	0,0004	0	0
10	1,29122	0,0136	0	0,0004	0
15	-	-	0,0012	0,001	0
20	-	-	0,0014	0,003	0,0004
50	-	-	0,0048	0,04264	0,001
100	-	-	0,03748	0,28518	0,0034
250	-	-	0,41058	4,66732	0,02302
500	-	-	3,05814	47,6364	0,11816
750	-	-	9,5506	181,303	0,2702
1000	-	-	23,1604	502,899	0,5428

Таблица 4 – Среднее отклонение длины пути от эталона для различных методов решения задачи коммивояжера.

Количество точек	Перебора	Ветвей и границ	Ветвей и границ приближенный	Включение ближайшего соседа
5	-0,5374	-0,5374	-0,4442	4,30961
7	-2,0879	-2,0879	-1,2663	3,35281
10	-5,2279	-5,2279	-1,8503	8,52554
15	-	-	-4,9621	6,23281
20	-	-	-0,4979	1,28686
50	-	-	4,51109	4,99681
100	-	-	8,17604	6,19559
250	-	-	11,3477	5,29257
500	-	-	11,6889	4,77628
750	-	-	13,1288	4,5987
1000	-	-	12,2792	4,23648
Среднее	-	-	**4,73735**	**4,89128**

3). Происходит склейка контуров A_j и A_k.

4). $n = n - 1$.

5). Если $n = 1$, т.е. после операции склейки остался один контур, алгоритм завершает свою работу, иначе – переход на шаг 2.

Операция склейки представляет собой процедуру объединения двух контуров, при которой из каждого контура удаляется по одному ребру, а затем добавляются два ребра так, чтобы получился замкнутый контур.

Главной особенностью данного алгоритма является метод выбора склеиваемых контуров. Для каждой пары контуров

A_j и A_k ($j,k=\overline{1,n}$) ищутся такие ребра (d,c) и (f,g), чтобы $-l_{de}-l_{fg}+l_{df}+l_{eg}$, где l_{dc} - длина ребра (d,c), принадлежащего контуру A_j, l_{fg} - длина ребра (f,g), принадлежащего контуру A_k, l_{df}, l_{cg} - длины добавляемых ребер, было минимальным. Получившее значение составляет оценку C_{jk}, которая представляет собой критерий для выбора склеиваемых контуров A_j и A_k.

Далее, при помощи просмотра всех получившихся оценок, выбирается минимальная оценка C_{jk}, которая говорит о том, что в результате склейки контуров A_j и A_k получится контур минимальной длины из всех возможных.

Таким образом, для того, чтобы решить задачу коммивояжера для n точек, потребуется $n-1$ операций склейки (объединения) контуров.

Сразу же можно предложить существенное улучшение этого алгоритма. На каждом шаге кристаллизации не гарантировано, что контур, полученный в результате склейки, будет оптимальным. Поэтому после каждой склейки для получившегося контура можно выполнять процедуру k - оптимизации. Это улучшит качественные показатели решения, но ухудшит временные характеристики алгоритма.

Главным преимуществом алгоритма кристаллизации является то, что он предназначен для решения как симметричной задачи коммивояжера, так и несимметричной.

Для того, чтобы сравнивать различные методы решения некоторой задачи, необходимо провести как теоретическое сравнение, так и практическое сравнение, т. е. пользуясь данными, полученными в результате эксперимента.

Теоретическое сравнение по объему используемой памяти и трудоемкости для некоторых алгоритмов решения задачи коммивояжера приведено в таблице 1.

Из таблицы видно, что самыми трудоемкими методами являются точные методы – метод полного перебора и ветвей и границ. Причем, метод ветвей и границ менее трудоемок за счет того, что количество используемой им памяти гораздо больше, нежели у метода полного перебора.

Приближенные методы решения имеют примерно одинаковые трудоемкость и количество используемой памяти. Разница заключается лишь в оценке качества решения, которую можно получить лишь экспериментальным методом.

Экспериментальный анализ методов будем проводить по следующим критериям:

1). Средний объем используемой памяти.
2). Среднее время работы алгоритма.
3). Среднее отклонение длины пути от эталона. В качестве эталона выберем результат, получившийся в результате работы алгоритма кристаллизации. Отклонение будем выражать в процентном отношении, т.е. получившееся положительное значение x будет означать, что длина пути у исследуемого метода больше на $x\%$, чем у пути, получившего в результате применения алгоритма кристаллизации.

В результате эксперимента для количества точек 5, 7, 10, 15, 20, 50, 100, 250, 500, 750, 1000, получены результаты, показанные в таблицах 2 – 4. Указанное время – в секундах, объем памяти – в байтах, отклонение – в процентах.

Из приведенных выше таблиц можно сделать вывод о применимости того или иного метода. Если нужно точное решение, необходимо применять либо метод ветвей и границ, либо метод полного перебора. Метод полного перебора самый медленный, однако, в отличие от метода ветвей и границ, использует мало памяти. Эксперимент показал, что максимальное количество точек, для которого можно решить задачу коммивояжера за приемлемое время для метода перебора – 12, для метода ветвей и границ – 20.

Если точность не играет большой роли или количество точек больше 20, необходимо использовать приближенные методы решения. Так как приближенные методы решения используют примерно одинаковый объем памяти, есть смысл сравнивать их только по быстродействию и по качеству решения. Из таблицы 3 видно, что самым быстрым является метод включения ближайшего соседа. Самым медленным – приближенный алгоритм ветвей и границ. Однако, метод кристаллизации – самый точный.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Меламед И. И., Сергеев С. И., Сигал И. Х., Задача коммивояжера. Вопросы теории // Автоматика и телемеханика. – М.: Наука, 1989. №9. с.3-33.
2. Гимади Э. Х., Глебов Н. И., Перепелица В. А., Алгоритмы с оценками для задач дискретной оптимизации, Сб. «Управляемые системы», Новосибирск, вып. 11, 1974, 35-41.
3. Меламед И. И., Сергеев С. И., Сигал И. Х., Задача коммивояжера. Точные методы // Автоматика и телемеханика. – М.: Наука, 1989. №10. с.3-29.
4. Меламед И. И., Сергеев С. И., Сигал И. Х., Задача коммивояжера. Приближенные алгоритмы // Автоматика и телемеханика. – М.: Наука, 1989. №11. с.3-26.

УДК 681.51

Ревотюк М.П., Тихомирова Е.В.

АЛГОРИТМИЧЕСКАЯ ИНТЕРПРЕТАЦИЯ ПРОЦЕССОВ НА ВРЕМЕННЫХ СЕТЯХ ПЕТРИ

Известно, что наиболее привлекательным видом моделирования дискретных процессов с регулярной структурой являются сети Петри и их расширения [1]. Ориентация сетей Петри на отражение свойства восприимчивости реальных систем к локальным изменениям переменных состояния весьма удобна как при формализации дискретных процессов со сложными асинхронными взаимодействиями, так и реализации технологий объектно-ориентированного проектирования и программирования [2]. Цель исследования здесь - построение эффективных алгоритмов интерпретации процессов

Ревотюк Михаил Павлович. *Доцент каф. ИТАС Белорусского государственного университета информатики и радиоэлектроники.*
Тихомирова Екатерина Валерьевна. *Аспирант Белорусского государственного университета информатики и радиоэлектроники.*
Беларусь, г. Минск.