УДК 519.95

ОБ ОДНОМ РЕШЕНИИ ЗАДАЧИ КОММИВОЯЖЕРА

Л. А. ПЧЕЛИНЦЕВ

(Москва)

В работе излагается метод, который дает возможность найти оптимальную стратегию коммивояжера посредством итераций, а не путем решения рекуррентных соотношений (см. [1], [2]). Указанный метод позволяет учитывать возможность посылки сразу нескольких, например l, коммивояжеров. По объему требуемой памяти ЭВМ предлагаемый метод при $l=1$ равноценен методу динамического программирования. Логика итерационного процесса нахождения оптимальной стратегии коммивояжера не требует исследования сложной задачи целочисленного линейного программирования (см., например, [3]). В данной работе задача формулируется для случая l коммивояжеров.

1. Постановка задачи. Дано n городов с матрицей $C=\{c_{\lambda r}\}$ $(\lambda,\ r=1,\ldots,n)$ стоимостей проезда между ними. Из первого города выезжает l, $1\leqslant l\leqslant n-1$, коммивояжеров с задачей объехать все остальные города и вернуться в первый город. В каждом городе должен побывать только один (любой) коммивояжер.

Таким образом, каждый k-й коммивояжер, $k=1,\ \ldots,\ l$, объедет совершенно определенное подмножество городов S_k в определенной последовательности, понеся при этом потери, равные сумме стоимостей переездов между городами данного подмножества. Причем подмножества S_k, $k=1,\ \ldots,\ l$, не пересекаются между собой. После того как все l коммивояжеров вернутся в первый город (отправной пункт), подсчитываются суммарные потери как сумма потерь отдельных коммивояжеров. Задача состоит в том, чтобы найти такое разбиение всего множества $n-1$ городов (без учета отправного пункта) на l подмножеств S_k, $k=1,\ \ldots,\ l$, и такой порядок объезда городов в каждом S_k, чтобы суммарные потери были минимальны.

2. Основные определения. Введем пространство состояний $X=\{x_i\}$, $i=1,\ \ldots,\ 1+w$, где x_i есть $(n-1)$-мерный вектор (без учета отправного пункта), μ-я координата которого равна нулю, если какой-либо из коммивояжеров уже побывал в μ-м городе; единице, если μ-й город еще не посетил ни один коммивояжер; 0^+, если в μ-м городе находится какой-либо коммивояжер. Состояние $x_1=(111\ldots\ldots11)$ назовем начальным, а состояние $x_{1+w}=(000\ldots00)$ конечным. Некоторое промежуточное (переходное) состояние будет иметь, например, такой вид: $x_1=(10^+0000^+110^+11\ldots110)$.

Заменим понятие l коммивояжеров понятием одного условного коммивояжера. Процесс путешествия условного коммивояжера начинается из начального состояния x_1, проходит некоторые переходные состояния и заканчивается всегда в конечном состоянии x_{1+w}.

Переход процесса из одного состояния в другое будем называть поездкой условного коммивояжера и обозначать через π. Очевидно, что каждой поездке π соответствуют l переездов действительных коммивояжеров. Поскольку действительные коммивояжеры не должны заезжать в города, которые уже были посещены кем-либо из них, то в каждом определенном состоянии x_i, $i=1,\ \ldots,\ w$, условный коммивояжер может совершить одну поездку из вполне определенного подмножества $\Pi(x_i)=\{\pi\}$ допустимых поездок. Стоимость q этой поездки будет складываться из стоимостей соответствующих переездов, совершенных действительными коммивояжерами. Очевидно, что $\Pi=\Pi(x_1)\supset\Pi(x_i)$, $i=2,\ldots,w$, $\Pi(x_{1+w})=\varnothing$.

Назовем стратегией условного коммивояжера однозначное отображение δ: $X\to\Pi$, где каждому состоянию x_i стратегия δ назначает определенную поездку из числа допустимых, т. е. из $\Pi(x_i)$.

Фиксированная стратегия δ определяет детерминированную переходную матрицу $P^\delta=\{p(x_j\mid x_i,\ \pi)\}$ порядка $(1+w)\times(1+w)$ такую, что все члены i-й строки

p_{ij} равны нулю, за исключением одного, равного единице. Пусть B^δ — блок матрицы P^δ за вычетом строки и колонки для конечного состояния x_{1+w}.

Матрица B^δ для некоторой фиксированной стратегии δ однозначно определяет процесс путешествия условного коммивояжера (а следовательно, и процесс путешествия действительных коммивояжеров).

Пусть v_i^δ, $i = 1, \ldots, w$,— те суммарные потери, которые несет условный коммивояжер, если процесс его путешествия начинается с состояния x_i как с начального, совершается согласно стратегии δ и заканчивается в конечном состоянии x_{1+w}. Введем векторное обозначение $V^\delta = \{v_i^\delta\}$, $i = 1, \ldots, w$, где V^δ — вектор-столбец.

Пусть также $q_i^{\pi(\delta)}$ — это стоимость поездки условного коммивояжера, которая назначается стратегией δ состоянию x_i. Аналогично, $Q^\delta = \{q_i^{\pi(\delta)}\}$, $i = 1, \ldots, w$, где Q^δ — вектор-столбец.

3. Алгоритм решения задачи. В приведенных выше обозначениях нашей задачей является минимизация величины v_1^δ. Но очевидно, что оптимальная стратегия δ_*, которая будет минимизировать вектор V^δ в целом, т. е. $V^{\delta_*} = \min\limits_\delta V^\delta$, обеспечит и минимизацию его первой координаты v_1^δ.

Ниже мы приводим итерационную вычислительную схему для нахождения оптимальной стратегии δ_*.

1. Для данного числа городов n, данного числа коммивояжеров l и данных стоимостей переездов $\{c_{\lambda r}\}$, λ, $r = 1, \ldots, n$, составить список возможных состояний $X = \{x_i\}$, $i = 1, \ldots, 1 + w$, которые могут встретиться при путешествии условного коммивояжера.

2. Каждому состоянию x_i назначить некоторую поездку условного коммивояжера из числа допустимых, которая должна указывать всем l действительным коммивояжерам, куда они должны дальше ехать. В этом заключается задание некоторой начальной стратегии δ_0.

3. Исходя из возможных переходных состояний $x_1, \ldots, x_w$ и принятой стратегии δ_0, составить переходную матрицу B^{δ_0} порядка $(w \times w)$.

4. Определить составляющие вектора Q^{δ_0}.

5. Определить составляющие вектора V^{δ_0}, решив систему алгебраических уравнений вида

$$\{v_i^{\delta_0} = q_i^{\pi(\delta_0)} + b_{ij}^{\pi(\delta_0)} v_j^{\delta_0}\}, \qquad i, j = 1, \ldots, w.$$

Т. е.

$$V^{\delta_0} = (I - B^{\delta_0})^{-1} Q^{\delta_0},$$

где I — единичная матрица порядка $(w \times w)$.

6. Сделать первую итерацию по улучшению начального приближения δ_0, т. е. отыскать такую стратегию δ_1, что $V^{\delta_1} < V^{\delta_0}$. Для этого в каждом состоянии x_i надо отыскать такую поездку π из числа допустимых, которая минимизирует величину $v_i = q_i^\pi + b_{ij}^\pi v_j^{\delta_0}$, и сделать ее новой поездкой в состоянии x_i. Если такой поездки не найдется, то надо в состоянии x_i оставить старую поездку $\pi(\delta_0)$. После того как эти вычисления будут проведены для всех состояний x_i, $i = 1, \ldots, w$, определится новая стратегия δ_1 условного коммивояжера. Причем $V^{\delta_1} \leqslant V^{\delta_0}$.

7. Составить матрицу B^{δ_1}.

8. Определить составляющие вектора Q^{δ_1}.

9. Определить составляющие вектора V^{δ_1}, решив систему алгебраических уравнений вида, указанного в п. 5.

10. Совершить итерацию по улучшению приближения δ_1. Для этого вычислить стратегию δ_2, руководствуясь п. 6.

11. Составить матрицу B^{δ_2}.

12. Определить составляющие вектора Q^{δ_2}.

13. Определить составляющие вектора V^{δ_2}.

14. Вычислить δ_3 и т. д.

Легко показать, что данный алгоритм всегда сходится к оптимальной стратегии δ_* (см. [4]). Последовательные приближения следует закончить, когда $\delta_N = \delta_{N-1}$. Это значит, что $\delta_{N-1} = \delta_*$. Число приближений N зависит от того, насколько удачно выбрано начальное приближение δ_0. Стратегию δ_0 следует выбирать исходя из особенностей каждого конкретного варианта матрицы $\{c_{\lambda r}\}$. Иногда, например, сразу видно, что совершение в каждом состоянии наиболее дешевой поездки является очень хорошим начальным приближением. В других примерах это может и не оправдаться. Но из интуитивных соображений всегда можно выбрать некоторое начальное приближение δ_0.

В [1] указано, что при использовании рекуррентного метода динамического программирования для решения задачи коммивояжера для n городов при $l = 1$ требуется $M_d^1 = (n-1)2^{n-2}$ ячеек памяти ЭВМ. Имеется в виду число ячеек, которые обеспечивают запоминание достаточной статистики потерь, используемой в рекуррентных соотношениях. В нашем случае аналогичная память M_h^1 будет определяться числом возможных переходных состояний, т. е. числом w. Из комбинационных соображений можно написать, что при $l \neq 1$ имеет место соотношение

$$w = M_h^l = \sum_{m=l}^{n-1} C_{n-1}^m A_m^l,$$

где C_{n-1}^m — число сочетаний из $n-1$ по m, A_m^l — число размещений из m по l. Отсюда при $l = 1$ имеем

$$M_h^1 = \sum_{m=1}^{n-1} m C_{n-1}^m = M_d^1 = (n-1)2^{n-2}.$$

Т. е. для задачи с одним коммивояжером предлагаемый метод и метод динамического программирования, приведенный в [1], являются равноценными в смысле необходимого объема памяти.

Вывод. В работе предлагается итерационный метод решения задачи коммивояжера, расширенный на случай нескольких коммивояжеров.

В заключение автор выражает глубокую благодарность П. И. Кузнецову, В. Д. Поддерюгину и А. Н. Ширяеву за постоянное внимание и помощь в работе.

Поступила в редакцию
17.12.1964

Цитированная литература

1. М. Хелд, Р. М. Карп. Применение динамического программирования к задачам упорядочения. В «Кибернетич. сб.», № 9. М., «Мир», 1964, 202—218.
2. Р. Беллман. Применение динамического программирования к задаче о коммивояжере. В «Кибернетич. сб.», № 9. М., «Мир», 1964, 219—222.
3. В. И. Мудров. Один способ решения задачи коммивояжера с помощью целочисленного линейного программирования. Ж. вычисл. матем. и матем. физ., 1963, 3, № 6, 1137—1139.
4. Р. А. Ховард. Динамическое программирование и марковские процессы. М., Изд-во ин. лит., 1964.