ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Январь—март 1998. Серия 1. Том 5, № 1, 60–63

УДК 519.8

ПРИБЛИЖЕННЫЙ АЛГОРИТМ ДЛЯ РЕШЕНИЯ ЗАДАЧИ О p-ЦЕНТРЕ С НЕРАВЕНСТВОМ ТРЕУГОЛЬНИКА*)

М. И. Свириденко

Рассматривается взвешенная задача о p-центре, когда матрица расстояний удовлетворяет неравенству треугольника, т. е. для всех i, j, k выполняется неравенство $d_{ij}+d_{jk} \geqslant d_{ik}$. В [2] для приближенного решения этой задачи предлагается алгоритм с относительной погрешностью $\rho = 3$, временная сложность которого не превосходит величины $O(n^2 \log_2 n)$. В настоящей работе предлагается алгоритм с той же погрешностью и временной сложностью $O(n^2)$.

Введение

В задаче о p-центре в полном неориентированном взвешенном графе с множеством вершин V требуется найти такое множество $S \subseteq V$, минимизирующее расстояние от $V \setminus S$ до S, что $|S| \leqslant p$. Расстояние от множества $V \setminus S$ до множества S определяется как $\max\limits_{i \in V\setminus S} \min\limits_{j \in S} d_{ij}$.

В настоящей работе рассматривается обобщение задачи о p-центре. Приведем формальную постановку задачи: в полном неориентированном графе с множеством вершин V найти

$$\min_{S \subseteq V} \left\{ \max_{i \in V\setminus S} \min_{j \in S} d_{ij} \mid \sum_{i \in S} c_i \leqslant C \right\}, \tag{1}$$

где $d_{ij} \geqslant 0$ — вес ребра, инцидентного i-й и j-й вершинам, а $c_i \geqslant 0$ — вес i-й вершины. В дальнейшем будем предполагать, что матрица (d_{ij}) удовлетворяет неравенству треугольника, т. е. для всех i, j, k выполняется неравенство $d_{ij} + d_{jk} \geqslant d_{ik}$.

В [2] был предложен алгоритм для задачи (1) с относительной оценкой погрешности $\rho = 3$ и временной сложностью $O(n^2 \log_2 n)$, где $n = |V|$. В [3] было доказано, что для задачи о p-центре не существует алгоритма

*) Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (код проекта 97–01–00890).

с относительной оценкой погрешности $\rho < 2$ при условии, что $P \neq NP$. В настоящей статье с использованием алгоритма из [1] строится алгоритм с оценкой $\rho = 3$ и временной сложностью, не превосходящей величины $O(n^2)$.

1. Описание и трудоемкость алгоритма

Введем следующие обозначения:

$r(S)$ — значение целевой функции на множестве $S \subseteq V$;

$H(R, i) = \{j \mid d_{ij} \leqslant R\}$ — окрестность вершины i радиуса R;

$v(R, i)$ — вершина минимального веса в окрестности $H(R, i)$;

$T(R, S) = \{v(R, i) \mid i \in S\}$ — набор вершин минимального веса, поставленных в соответствие вершинам из множества S;

S^* — оптимальное решение задачи (1);

R^* — значение целевой функции на оптимальном решении, т. е. $R^* = r(S^*)$.

Алгоритм решения задачи (1) состоит из двух этапов. На первом этапе формируются два множества S и S_l. На каждом шаге первого этапа в множество S добавляется наиболее удаленная вершина. Если при этом изменяется расстояние от S до $V \setminus S$, то множество S_l изменяется. На втором этапе происходит улучшение уже полученного решения.

Алгоритм

Первый этап

Шаг 1. Выбирается произвольная вершина $i_1 \in V$ такая, что $c_{i_1} \leqslant C$, и полагается $S = \{i_1\}, S_l = \{i_1\}$. Для всех вершин $j \in V \setminus S$ находится $r_j = d_{i_1 j}$ (расстояние от j-й вершины до i_1-й вершины) и полагается $D_1 = \max\limits_{j \in V \setminus S} r_j$, $p = 1$, $l = 1$.

Шаг k. Находится вершина $i_k \in V \setminus S$ такая, что $r_{i_k} = D_{k-1}$ (т. е. самая удаленная от множества S вершина). Полагается $S = S \cup \{i_k\}$. Для всех $j \in V \setminus S$ находится $r_j = \min\{r_j, d_{j i_k}\}$ (расстояние от j-й вершины до множества S) и полагается $D_k = \max\limits_{j \in V \setminus S} r_j$, $p = k$.

а) Если $D_k = D_{k-1}$, то алгоритм выполняет $(k+1)$-й шаг.

б) Если $D_k < D_{k-1}$ и $\sum\limits_{i \in T(D_k/2, S)} c_i \leqslant C$, то полагается $l = k$, $S_l = T(D_k/2, S)$ и алгоритм выполняет $(k+1)$-й шаг.

в) Если $D_k < D_{k-1}$ и $\sum\limits_{i \in T(D_k/2, S)} c_i > C$, то выполняется второй этап.

Второй этап

Пусть R — минимальное число среди r_j таких, что $D_p/2 < r_j < D_l/2$ и $\sum\limits_{i\in T(r_j,S)} c_i \leqslant C$. Если такого R не существует, то полагается $\widetilde{S} = S_l$. В противном случае полагается $\widetilde{S} = T(R, S)$.

Конец

Заметим, что в конце работы первого этапа выполняется неравенство $D_p > D_l$. Докажем вспомогательную лемму для оценки временной сложности алгоритма.

Лемма 1. *Пусть дано множество $S \subseteq V$ такое, что $\min\limits_{i,j\in S} d_{ij} > D$. Тогда множество $T(D/2, S)$ можно вычислить за $O(n)$ действий.*

Доказательство. Так как $H(D/2, i) \cap H(D/2, j) = \varnothing$ при $j,\ i \in S$ и $j \neq i$, то множества $H(D/2, i)$ при $i \in S$ вычисляются за один просмотр множества V, т. е. за $O(n)$ действий. Множество $T(D/2, S)$ вычисляется не более чем за $O\Big(\sum\limits_{i\in S} |H(D/2, i)|\Big)$ действий, что не превосходит величины $O(n)$. Лемма 1 доказана.

Число шагов на первом этапе не превосходит n. Используя лемму 1, получаем, что все вычисления, производимые на k-м шаге, выполняются не более чем за $O(n)$ действий. Следовательно, общая временная сложность первого этапа не превосходит величины $O(n^2)$. На втором этапе для каждого r_j такого, что $D_p/2 < r_j < D_l/2$, вычисляется $T(r_j, S)$. Так как число различных r_j не превосходит n, то временная сложность алгоритма не превосходит величины $O(n^2)$.

2. Оценка относительной погрешности

Лемма 2. *Если множество S таково, что $\min\limits_{i,j\in S} d_{ij} > 2R^*$, то множество $T(R^*, S)$ является допустимым решением задачи* (1).

Доказательство. Для любого $i \in S$ найдется $k(i) \in S^*$ такой, что $d_{ik(i)} \leqslant R^*$. Кроме того, для всех $j \in S \setminus \{i\}$ выполняется неравенство $d_{jk(i)} > R^*$, так как в противном случае из неравенства треугольника следует, что $d_{ij} \leqslant d_{ik(i)} + d_{k(i)j} \leqslant 2R^*$. Это противоречит условию леммы. Следовательно, $k(i) \neq k(j)$ для всех $i,\ j \in S,\ i \neq j$, и выполняются неравенства

$$\sum_{i\in T(R^*,S)} c_i = \sum_{i\in S} c_{v(R^*,i)} \leqslant \sum_{i\in S} c_{k(i)} \leqslant \sum_{i\in S^*} c_i \leqslant C.$$

Лемма 2 доказана.

Теорема 1. *Алгоритм находит такое множество* $\widetilde{S}$, *что* $r(\widetilde{S})/r(S^*) \leqslant 3$.

Доказательство. Рассмотрим три случая.

1. Пусть $D_l/2 \leqslant R^*$. Тогда если $\widetilde{S} = S_l$, то из неравенства треугольника следует, что $r(\widetilde{S}) \leqslant D_l/2 + D_l \leqslant 3R^*$. Если $\widetilde{S} = T(R, S)$, где $D_p/2 < R < D_l/2$, то $r(\widetilde{S}) \leqslant R + D_p < D_l/2 + D_l \leqslant 3R^*$.

2. Пусть $R^* \leqslant D_p/2$. Так как $\min\limits_{i,j \in S} d_{ij} = D_l > D_p \geqslant 2R^*$, то по лемме 2 имеем

$$\sum_{i \in T(D_p/2, S)} c_i \leqslant \sum_{i \in T(R^*, S)} c_i \leqslant C.$$

Это противоречит условию перехода с первого этапа на второй.

3. Пусть $D_p/2 < R^* < D_l/2$. Так как по лемме 2 множество $T(R^*, S)$ является допустимым решением задачи (1), то число R, выбираемое на втором этапе алгоритма, существует и удовлетворяет следующим условиям: $D_p/2 < R \leqslant \max\limits_{i \in S} d_{ik(i)} \leqslant R^*$ и $\sum\limits_{i \in T(R,S)} c_i \leqslant C$. Из неравенства треугольника следует, что $r(\widetilde{S}) = r(T(R, S)) \leqslant R^* + D_p < 3R^*$. Теорема 1 доказана.

ЛИТЕРАТУРА

1. **Dyer M. E., Frieze A. M.** A simple heuristic for the p-centre problem // Oper. Res. Lett. 1985. V. 3, N 6. P. 285–288.
2. **Hochbaum D. S., Shmoys D. B.** A unified approach to approximation algorithms for bottleneck problems // J. Assoc. Comput. Mach. 1986. V. 33, N 3. P. 533–550.
3. **Hsu W. L., Nemhauser G. L.** Easy and hard bottleneck location problems // Discrete Appl. Math. 1979. V. 1, N 3. P. 209–216.

Адрес автора:

Институт математики
им. С. Л. Соболева СО РАН,
пр. Академика Коптюга, 4,
630090 Новосибирск,
Россия

Статья поступила
23 апреля 1997 г.,
переработанный вариант —
11 февраля 1998 г.