

多参数最小支撑树问题的复杂性和算法*

李帮义 姚恩瑜

(浙江大学应用数学系 杭州 310027)

摘要 建立了多参数最小支撑树问题(RMST)的模型,并证明该问题是NP-完全的。利用经典Greedy算法,给出了该问题的一个近似算法,并分析了该近似算法的性能比,证明了所给出的界是紧的。

关键词 最小支撑树, NP-完全, 算法

分类号 O 122.4

Complexity and Algorithm of the Minimum Spanning Tree Problem with Multiple Parameter

Li Bangyi, Yao Enyu

(Zhejiang University)

Abstract The model of minimum spanning tree with multiple parameter is established. It is proved that the problem is NP-complete. Using the classical Greedy algorithm, a simple heuristic algorithm is given. Its bound is proved to be tight.

Key words minimum spanning tree, NP-complete, algorithm

1 引言

最小支撑树问题(MST)是一个经典的组合优化问题,在实际中有着广泛的应用。在MST中,每条边上只有一个参数,而实际网络中,每条边上可能有多个参数,如长度、成本、时间和可靠性等。在许多决策问题中,一个问题可有多个方案,不同的方案有不同的参数。因此建立如下支撑树问题的模型:

设 $G = (V, E, L^1, L^2, \dots, L^k)$ 是一个网络,对每条边 $v_i v_j$, 有非负长度参数 $l_{i,j}^r, 1 \leq r \leq k, k$ 是决策的方案数。设 T 是 G 的一棵支撑树,则

$$L^r(T) = \sum_{v_i v_j \in T} l_{i,j}^r, \quad 1 \leq r \leq k \quad (1)$$

作为 T 的第 r 个方案的长度。 T 的长度定义为 $L(T) = \min_{1 \leq r \leq k} L^r(T)$ 。RMST 为求 G 的一棵支撑树 T , 使 $L(T)$ 达到最小, 即

$$\min_T \max_{1 \leq r \leq k} \{L^r(T) : T \text{ 是 } G \text{ 的一棵支撑树}\} \quad (2)$$

RMST 的判定形式为: 对给定的实数 l , 是否存在 G 的一棵支撑树 T , 使

$$L^1(T) \leq l, L^2(T) \leq l, \dots, L^k(T) \leq l \quad (3)$$

当 $k = 1$ 时, 该问题即为一般意义下的最小支撑树。这类模型一般称为 Robust 模型, 许多专家对其进行了研究^[1-3]。

本文首先证明 RMST 是 NP-完全的, 然后利用 Greedy 算法给出一个近似算法, 并分析了该近似算法的性能比, 证明了所给出的界是紧的。

2 复杂性证明

RMST 复杂性的证明将用到划分问题的复杂性^[4]。划分问题为: 给定集合

$$A = \{a_1, a_2, \dots, a_n\}, \quad a_i \in R^+ \\ a_i = 2l, \quad 1 \leq i \leq n \quad (4)$$

问 A 能否划分为 A_1 和 A_2 , 使

* 国家重点基础研究专项经费资助项目

1999-09-27 收稿, 2000-03-01 修回

$$\begin{cases} A = A_1 \cup A_2, & A_1 \cap A_2 = \emptyset \\ a_i \in A_1, & a_i \in A_2 \end{cases} \quad (5)$$

定理1 RMST 问题是 NP-完全的。

证明 RMST 问题的判定形式显然是多项式时间内可验证的, 即 RMST 问题是 NP 的。下面把划分问题多项式时间归约到一个特殊的 RMST 问题。

现设划分问题的一个实例

$$A = \{a_1, a_2, \dots, a_n\}, \quad a_i = 2l \quad (6)$$

其中 a_i 为一个正实数。构造一个网络 G 如图 1 所示, 图中 $v_0 v_i$ 的两个长度参数为 $\{a_i, 0\}$, $v_i v_{n+2}$ 的长度参数为 $\{0, a_i\}$, $1 \leq i \leq n, k = 2$; $v_0 v_{n+1}$ 和 $v_{n+1} v_{n+2}$ 的两个长度参数为 $\{0, 0\}$ 。

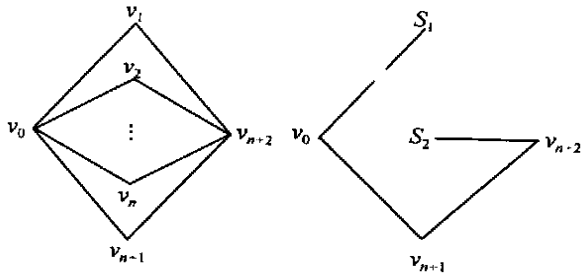


图 1 网络 G 图 2 G 的一棵支撑树 T

由图 1 知, 划分问题有解的充分必要条件为网络 G 中存在一棵支撑树 T , 使

$$L^1(T) = l, \quad L^2(T) = l \quad (7)$$

设划分问题有解, $A = A_1 \cup A_2$, 使

$$a_i \in A_1, \quad a_i \in A_2 \quad (8)$$

则定义顶点集合

$$S_1 = \{v_i: a_i \in A_1\}, \quad S_2 = \{v_i: a_i \in A_2\}$$

$$S_1 \cap S_2 = \{v_1, v_2, \dots, v_n\}$$

构造网络 G 的一棵支撑树 T 如图 2 所示。

由 S_1 和 S_2 的定义知, T 满足

$$L^1(T) = \sum_{a_i \in A_1} a_i = l, \quad L^2(T) = \sum_{a_i \in A_2} a_i = l \quad (9)$$

设 G 有一棵支撑树 T 满足式 (7), 则 T 有 $n+2$ 条边。对于顶点 $v_1, v_2, \dots, v_n, v_{n+1}$, 至少每个顶点与一条边关联。根据 G 的构造, $v_1, v_2, \dots, v_n, v_{n+1}$ 这 $n+1$ 个顶点中, 有 n 个顶点与一条边关联, 有一个顶点与两条边关联。

下面证明与两条边关联的顶点必为 v_{n+1} 。用反证法, 不妨假设与两条边关联的顶点为 v_1 , 则 T 中顶点集合 $\{v_2, v_3, \dots, v_n\}$ 分成两部分, 一部分与 v_0 相

邻, 设为 V_1 ; 另一部分与 v_{n+2} 相邻, 设为 V_2 。与 v_{n+1} 关联的顶点可以是 v_0 , 也可以是 v_{n+2} , 分别示于图 3 和图 4。

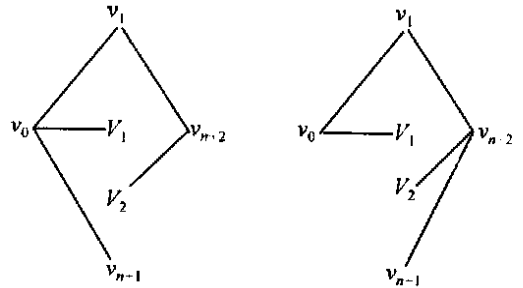


图 3 与 v_{n+1} 关联的顶点是 v_0 时树 T 的结构 图 4 与 v_{n+1} 关联的顶点是 v_{n+2} 时树 T 的结构
于是有

$$L^1(T) = a^1 + \sum_{a_i \in V_1} a_i, \quad L^2(T) = a^1 + \sum_{a_i \in V_2} a_i$$

$$L^1(T) + L^2(T) = 2a^1 + \sum_{a_i \in V_1 \cup V_2} a_i = a^1 + 2l$$

这与式 (7) 矛盾。

由以上分析知, 在树 T 中, v_{n+1} 与两条边关联, v_1, v_2, \dots, v_n 各与一条边关联, 故 $\{v_1, v_2, \dots, v_n\}$ 分成两个集合 S_1 和 S_2 , 一个与 v_0 相邻接, 一个与 v_{n+2} 相邻接。即

$$\begin{cases} S_1 = \{v_i: v_0 v_i \in T, 1 \leq i \leq n\} \\ S_2 = \{v_i: v_i v_{n+2} \in T, 1 \leq i \leq n\} \end{cases} \quad (10)$$

树 T 的结构如图 5 所示。

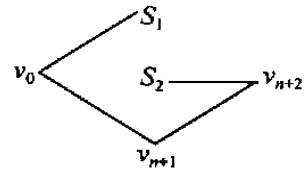


图 5 树 T 的结构

由假设知式 (7) 成立, 又

$$L^1(T) + L^2(T) = \sum_{v_i \in S_1} a_i + \sum_{v_i \in S_2} a_i = 2l \quad (11)$$

故 $L^1(T) = l, \quad L^2(T) = l \quad (12)$

由 S_1 和 S_2 的定义, 集合 A 的一个划分为

$$A_1 = \{a_i: v_i \in S_1\}, \quad A_2 = \{a_i: v_i \in S_2\} \quad (13)$$

则 $a_i \in S_1, \quad a_i \in S_2 \quad (14)$

A_1, A_2 即为所求的一个划分。

上述网络的构造显然是多项式时间内可完成的, 故 RMST 问题是 NP-完全的。(证毕)

注 1 由于 RMST 问题是 NP-完全的, 故该问

题不存在多项式时间算法,除非 $P = NP$ 。

3 近似算法及性能比

在 RMST 中,每条边上有多个参数,因此求解最小支撑树的经典算法对 RMST 是无效的。

尽管 Greedy 算法求解 RMST 无效,但可以求出每条边的平均长度,用平均长度代替原长度向量,然后用 Greedy 算法求解 MST,用所求出的支撑树作为原问题的一个近似解。

算法 1 步骤如下(近似算法):

Step1: 对每条边求平均长度

$$\bar{l}_{i,j} = \frac{1}{k} \sum_{r=1}^k l_{i,j}^r \quad (15)$$

Step2: 调用 Greedy 算法,在长度参数 $\bar{l}_{i,j}$ 下求最小支撑树,设结果为 T , 长度为 L ;

Step3: 计算近似解

$$L_{opt}^A = \max_{1 \leq r \leq k} L^r(T) \quad (16)$$

下面分析近似算法的性能比,为此首先给出一个定义和一个引理。

定义 1 $\theta = \frac{\max_{1 \leq r \leq k} L^r(T)}{\min_{1 \leq r \leq k} L^r(T)}$

显然 $\theta \geq 1$, 它反映了对于近似解 T , 其最大长度与最小长度的比值。

引理 1 设最优解的长度为 L_{opt} , 则 $L_{opt} \leq L$ 。

证明 设 T^* 是一棵最优树, 则

$$L_{opt} = \min_{1 \leq r \leq k} L^r(T^*) \leq \frac{1}{k} \sum_{1 \leq r \leq k} L^r(T^*) \leq L$$

定理 2 对于算法 1 所求出的近似解 T , 有

$$\frac{L_{opt}^A}{L_{opt}} \leq \frac{\theta k}{\theta + k - 1} \quad (17)$$

证明

$$\begin{aligned} L_{opt}^A &= \max_{1 \leq r \leq k} L^r(T) \\ &= \frac{1}{k} [\max_{1 \leq r \leq k} L^r(T) + (k-1) \max_{1 \leq r \leq k} L^r(T)] \\ &= \frac{1}{k} [\max_{1 \leq r \leq k} L^r(T) + (k-1) \theta \min_{1 \leq r \leq k} L^r(T)] \\ &= \frac{1}{k} \max_{1 \leq r \leq k} L^r(T) + \frac{\theta}{k} [\max_{1 \leq r \leq k} L^r(T) - \min_{1 \leq r \leq k} L^r(T)] \\ &= \frac{1-\theta}{k} L_{opt}^A + \theta L_{opt} \end{aligned}$$

上式经整理即得式(17)。(证毕)

定理 2 给出了近似算法的性能比, 该性能比能否继续改进? 下面的推论回答了这一问题。

推论 1 近似算法的性能比 $\frac{\theta k}{\theta + k - 1}$ 是紧

的。

实际上, 由图 6 所示的网络(其中 $\theta \geq k$, k 是一个大于 1 的自然数, 每条边旁的 k 重参数是长度向量)恰好达到了该比值。由本文算法得到的近似解见图 7(a), 实际最优解见图 7(b), 其性能比为

$$\frac{L_{opt}^A}{L_{opt}} = \frac{\theta}{(\theta + k - 1)/k} = \frac{k\theta}{\theta + k - 1}$$

恰好达到了定理 2 中的界。

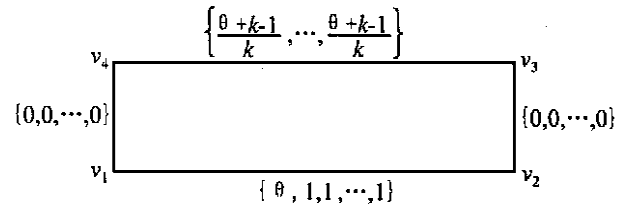


图 6 网络



图 7 结果比较

(a) 近似解 (b) 实际最优解

经实例计算, 本文算法的实际性能比远小于 $\frac{\theta k}{\theta + k - 1}$, 基本能达到实际决策精度的要求。

参考文献

- Gupta S K, Rosenhead J. Robustness in sequential investment decisions. Management Science, 1972, 15 (2): 18~29
- Sengupta J K. Robust decisions in economic models. Computers and Operations Research, 1991, 18(2): 221~232
- Yu G. On the max-min 0-1 knapsack problem with robust optimization application. Operation Research, 1996, 44(2): 407~415
- Garey M R, Johnson D S. Computers and intractability. San Francisco: H Freeman, 1979

作者简介

李帮义 男, 1963 年生。1988 年于山东大学获运筹学与控制论专业硕士学位, 现为浙江大学应用数学系副教授, 博士研究生。主要研究领域为决策最优化, 网络最优化等。

姚恩瑜 女, 1941 年生。浙江大学数学系教授, 博士生导师。主要研究领域为算法分析, 数学规划等。