第 33 卷 第 6 期 西安交通大学学报 Vol.33 №6
1999 年 6 月 JOURNAL OF XI'AN JIAOTONG UNIVERSITY Jun.1999

已知拓扑下的 4 度 Steiner 树算法

叶继昌, 徐寅峰
(西安交通大学, 710049, 西安)

摘要: 设 N 为平面上 $2n$ 个固定点的集合, M 为 n-2 个可动点的集合, E 为连接这些点的边的集合(也称作拓扑). 设 E 为点集 V 上的满 4 度 Steiner 拓扑(满 Steiner 拓扑也就是满足固定点的度为 1, 可动点的度为 4 的树的拓扑), $H(E)$ 为包含 E 在内的所有 E 的退化拓扑的集合. 文中构造了计算拓扑属于 $H(E)$ 的 4 度 Steiner 树算法, 并证明了算法的时间复杂性是 $O(n^2)$.

关键词: Steiner 树; 拓扑; 网络
中国图书资料分类法分类号: TN711; T733; O157.5

Algorithm for 4-Degree Steiner Tree Topology

Ye Jichang, Xu Yinfeng
(Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: Let N denote a set S of $2n$ fixed points and M a set of m moving points in the plane. A set E of edges connecting these points is called a topology. Let E be a full 4-degree Steiner topology (i.e. the topology of the tree which contains fixed points of degree 1 and moving points of degree 4) on the set V such that the set of topology $H(E)$ includes E and it's degenerate cases. An algorithm is presented for finding a 4-degree Steiner tree whose topology belongs to $H(E)$. The time complexity of the algorithm is $O(n^2)$.

Keywords: *steiner tree; topology; networks*

设 N 是 $2n$ 个固定点的集合, M 是 n-1 个可动点的集合, N 和 M 中的元素分别称为正则点和 Steiner 点, 互联 $V = N \cup M$ 的网络边的集合称为拓扑. 如果 V 上的拓扑满足正则点的度为 1, Steiner 点的度为 4, 则称它为满 4 度 Steiner 拓扑(F4DST). 设 D 为 V 上所有 F4DST 及其退化拓扑(即 F4DST 中某正则点与某 Steiner 点重合的拓扑)的集合, 记 t 为点集 V 上的树, $T(t)$ 表示树 t 的拓扑, $l(t)$ 表示树 t 的长度, 如果树 t 满足任何点处两相邻角之和都不小于π(称为 4 度 Steiner 树的角条件)且其拓扑 $T(t) \in D$, 则称 t 为 4 度 Steiner 树(简记作 4DST), $T(t)$ 称为 4 度 Steiner 拓扑. 如果 $T(t)$ 是 F4DST, 则称 t 为满 4 度 Steiner 树. 设 E 为 F4DST, 用 $H(E)$ 表示包含 E 在内的所有 E 的退化拓扑的集合. 本文构造了计算 $H(E)$ 中的 4DST 的 $O(n^2)$ 算法.

关于平面 Steiner 最小树问题[1,2], M.R. Garey[3] 证明了它是 NP-难的. Hwang 和 Weng[4] 讨

收稿日期: 1998-06-23. 作者简介: 叶继昌, 男, 1963 年 1 月 26 日生, 理学院科学计算与应用软件系, 副教授.
基金项目: 国家自然科学基金重点资助项目(19731001).

论了已知拓扑下的 Steiner 最小树算法, 本文将其推广到 4DST 问题上.

1　基本概念和定义

构造 $H(E)$ 中的 4DST 可以看成是调整 E 的边, 因为这些边的交点确定了 Steiner 点. 为方便起见, 将边看成是有向的, 当其处在每一可能调整的位置上时称其为射线. 射线 r 的起始点称为基, 记为 $b(r)$. 从水平线沿逆时针方向旋转到 r 的角记成 $\theta(r)$. 将 E 中边 e 对应的所有射线的集合称为发射体. 将与正则点相连的边称为终端边. 与终端边对应的发射体为从正则点出发沿各个方向的射线的集合, 这样的发射体称为星. 如果 E 中的两边与同一个 Steiner 点相连, 则称这两个发射体相邻. 如果分别属于两个发射体的两条射线是相向的, 则称这两条射线为相对的对子(简称相对对).

下面我们来解释 Steiner 合并、星合并、基合并等概念. 设 L_1、L_2、L_3 分别是边 e_1、e_2、e_3 所对应的星发射体, 边 e_1、e_2、e_3 与同一 Steiner 点 s 相连, 则 L_1、L_2、L_3 是相邻的. 设与 s 相连的第 4 条边为 e_4, 那么这 3 个发射体通过合并得到的新发射体 L_4 为边 e_4 所对应的发射体. 合并是指从 L_1、L_2、L_3 中分别取出一条射线. 根据 4DST 的角条件确定与 e_4 对应的一条射线, 首先求 L_1 和 L_2 的相对对, 设 $r_{12} \in L_1$ 和 $r_{21} \in L_2$, 在 L_3 中选取与线段 $b(r_{12})\,b(r_{21})$ 相交的射线 r_3, 由这 3 条射线确定的 L_4 中的射线 r_4 为与 r_3 平行且同向, r_4 的基 $b(r_4)$ 为线段 $b(r_{12})\,b(r_{21})$ 与射线 r_3 的交点, 如图 1 所示. 这种方式生成的射线称为 Steiner 合并. 如果 3 个相邻的发射体中含有星, 我们则要考虑所谓的基合并或星合并. 设 L_1 是星, 若 L_1 和 L_2, L_1 和 L_3 的相对对都存在, 分别为 r_{12} 和 r_{21}, r_{13} 和 r_{31}, 若 r_{21} 和 r_{31} 的夹角小于 π, 则由在射线 r_{21} 和 r_{31} 组成的角形域中, 以 r_{21} 和 r_{31} 的交点为基在这个角形域内的所有射线构成的集合称为角体, 它是 L_4 的子集. 将 r_{21} 和 r_{31} 称为角体的支撑柄, 并称以这种方式生成的射线集为基合并, 如图 2a 所示. 图 2b 为 3 个星的基合并. 若 r_{21} 和 r_{31} 的夹角等于 π, 则将 L_1、L_2 和 L_3 合并生成的 L_4 为 L_1 称为伪星, 将 r_{21} 和 r_{31} 称为伪星的轴. 这种合并称为星合并, 如图 3 所示.

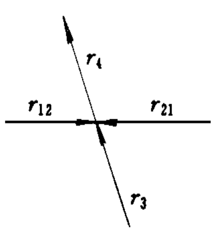
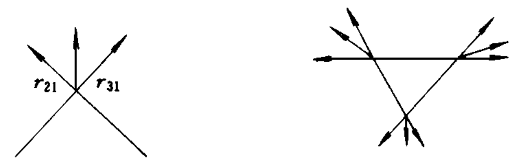


图 1　Steiner 合并



(a) 一个星的基合并　　(b) 三个星的基合并

图 2　基合并

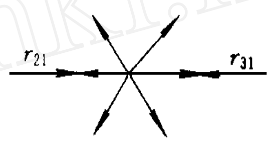


图 3　星合并

如果 B 是一条射线或是满足下列条件, 则称 B 为射线束:

(1) B 是由连续的射线构成的集合, 且这些射线所张成的角 $\theta(B) \le \pi$.

(2) 集合 $b(B) = \{ b(r) \mid r \in B\}$ 称为 B 的基, 它是一直线段或一个点.

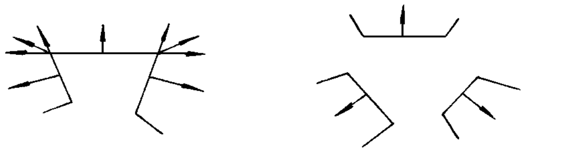
如果两个射线束接触处既不重迭也没有裂缝且它们的基线构成一直线段, 则称它们是相邻的. 多射线束 $M(B_1, \cdots, B_m)$ 或是一个射线束 $(m = 1)$ 或是多个相邻的射线束 $(m > 1)$, 满足 $\theta(M) = \sum_{i=1}^{m} \theta(B_i) \le \pi$, 如图 4 所示.


(a) 一个射线束　　(b) 多射线束

图 4　射线束示意图

根据 4DST 的角条件, 4DST 的所有顶点的度都不超过 4. 因此, 一条能被收缩而不引入 5 度及 5 度以上顶点的边的充要条件是被收缩的边是终端边. 如果发射体 L 上有沿各个方向的射线, 且其基 $b(L) = \{ b(r) \mid r \in L\}$ 是一个点, 则 L 要么是星要么是伪星. 如果 L 的基 $b(L)$ 是三角形, 则称 L 为三角形. 如果 L 由多个多射线束的并组成, 则称 L 为残缺三角形, 如图 5.


(a) 星, 伪星　　(b) 三角形



(c) 有角体的残缺三角形　(d) 无角体的残缺三角形

图 5　各种发射体

2　算法

本算法由 2 部分组成:

(1) 合并阶段. 合并阶段的每一步都在某一点(如 s)处相邻的发射体 L_1、L_2 和 L_3 被合并成一个新的发射体 L_4(若 L_4 为一个空发射体,则 $H(E)$ 中的 4DST 不存在),这个 L_4 与和 s 相连的第 4 个边对应. 合并阶段开始时,有 $2n$ 个终端边,对应 $2n$ 个星. 每个合并步都减少 2 个发射体,在 $n-1$ 个合并步后,仅剩下 2 个发射体,此时,合并阶段结束. 为方便起见,我们要求剩下的 2 个发射体都不是星,这是容易做到的. 在第 $n-2$ 个合并步后,我们有 4 个发射体,这 4 个发射体中至少有一个不是星,我们可以合并另外 3 个发射体,从而使得到的 2 个发射体都不是星.

(2) 调整阶段:由于 E 中有 $3n-2$ 个边,而每个合并步都去掉 3 个边,所以合并结束时仅剩下一个边 e 对应剩下的 2 个发射体. 寻找这 2 个发射体的相对对,若不存在,则 $H(E)$ 中的 4DST 不存在;否则,相对对的 2 个基之间的线段就确定了边 e. 从 e 开始,我们连续地回溯追踪,最后 $3n-2$ 个边被确定,得到一个 4DST.

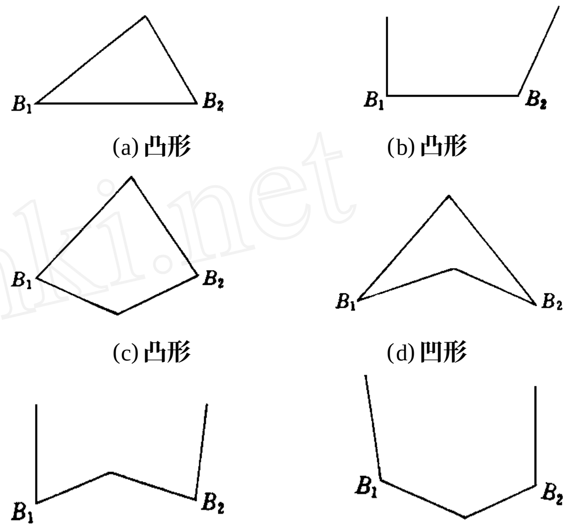
定理 1　算法输出一个 4DST 的充要条件是 $H(E)$ 在 V 中存在 4 度 Steiner 树.

证明　对于算法中的 3 种合并类型,显然合并阶段保留了所有 4DST,而并没有引入任何非 4DST. 合并阶段结束时,对应同一个边 e 有两个发射体 L 和 L'. 由先前的假设剩下的两个发射体都不是星,那么边就不是终端边. 因而,如果边 e 收缩,则 4DST 不存在. 如果发射体 L 和 L 中不存在相对对,那么 $H(E)$ 中也不存在 4DST.

如前所述,一旦找到相对对,接下去的调整将会自动地完成并生成一个 4DST. 定理 1 证毕.

3　算法的复杂性

设 B_1 和 B_2 是两个射线束,如果 B_1 和 B_2 的边界射线构成如图 6 中的几种形状,则称 B_1 和 B_2 是可构形的,并称图 6a、b、c、f 为凸形,d、e 为凹形. 否则称 B_1 和 B_2 是不可构形的.



(a) 凸形　(b) 凸形　(c) 凸形　(d) 凹形　(e) 凹形　(f) 凸形

图 6　B_1 和 B_2 的构形图

下面的引理 1 是显然的.

引理 1　两射线束 B_1 和 B_2 存在相对对的充要条件是 B_1 和 B_2 是可构形的且其构形图是凸形.

首先讨论确定相对对的时间复杂性. 我们给出如下引理.

引理 2　确定两多射线束 $M(B_1, \cdots, B_p)$ 和 $M(B'_1, \cdots, B'_q)$ 的相对对,或说明其相对对不存在所需时间均为 $O(p+q)$.

证明　设 $M(B_1, \cdots, B_p)$ 和 $M(B'_1, \cdots, B'_q)$ 如图 7 所示. 如果 B_1 和 B'_1 是可构形的,那么其构形图若是凸形,由引理 1 知在 B_1 和 B'_1 中存在相对对,连接 $t(B_1)$ 和 $t(B'_1)$ 即得相对对. 若是凹形,相对对应在 $M(B_1, \cdots, B_p)$ 和 $M(B'_2, \cdots, B'_q)$ 中. 如果 B_1 和 B'_1 是不可构形的,不妨设 B_1 和 B_2 的公共边界与 B'_1 的两条边界不相交但与 $M(B'_2, \cdots, B'_q)$ 相交,如图 8,显然相对对只能在 $M(B_1, \cdots, B_p)$ 和 $M(B'_2, \cdots, B'_q)$ 中. 如此下去,至多经过 $p+q$ 步就可得到相对对或说明不存在. 若相对对存在,则两个射线束的顶的连线就确定了相对对,因而可用常数时间完成. 引理 2 证毕.

其次讨论 3 个相邻发射体 L_1、L_2、L_3 经合并生

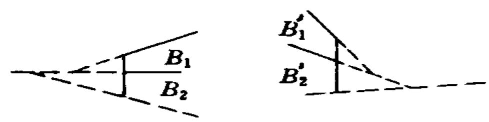


图 7　$M(B_1, \cdots, B_p)$ 和 $M(B'_1, \cdots, B'_q)$

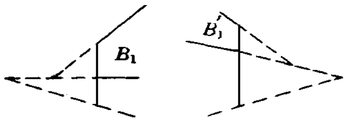


图 8　B_1 和 B_1' 是不可构形的

成 L_4 所需时间及 L_4 中射线束的个数. 为了讨论方便,我们将星和伪星看成是由两个射线束组成. 设 L_1、L_2 和 L_3 分别含有 l_1、l_2 和 l_3 个射线束. L_1、L_2、L_3 进行的合并无非是基合并、Steiner 合并、星合并. 无论进行哪种合并都要确定相对对. 由引理 2,确定相对对需 $O(l_1+l_2+l_3)$ 时间. 由此,我们在已知相对对的前提下讨论这 3 种合并. 由星合并和基合并的过程知它们可在常数时间内完成. 先看看射线束、星或伪星的 Steiner 合并. 由于射线束是由基和它的顶所确定,当基为点时,这个射线束是角体,这时它由它的两个支撑柄即可确定,星和伪星由一点也可确定. 所以由 Steiner 合并的定义,这个合并可用常数时间完成;由上述可知 L_1, L_2, L_3 的 Steiner 合并至多用 $O(l_1+l_2+l_3)$ 时间;由于星或伪星都被看成是由两个射线束组成,而星或伪星的 Steiner 合并至多生成一个射线束,所以若 L_1, L_2, L_3 中有 $k(0\leq k\leq 3)$ 个星或伪星,则至多生成 $l_1+l_2+l_3-k$ 个射线束. 由此得 L_1、L_2、L_3 合并至多需 $O(l_1+l_2+l_3)$ 时间. 注意到若进行星合并,就不能进行 Steiner 合并和基合并,我们由 3 个发射体的合并生成的新发射体的射线束数不超过原来 3 个发射体的射线束总数.

最后,我们讨论算法的复杂性. 算法开始时我们有 $2n$ 个星,因而共有 $4n$ 个射线束. 我们已经说明了每个合并步并不增加射线束总数. 因此,发射体 L 至多包含 $4n$ 个射线束. 故每个合并步至多需 $O(n)$ 时间. 由于有 $n-1$ 个合并步,因此合并阶段共需时间 $O(n^2)$.

在调整阶段开始,我们需要寻找相对对,由上述讨论知需 $O(n)$ 时间. 在接下去的调整过程中,已知一条射线 r,要找出生成射线 r 的 3 条射线 r_1、r_2、r_3,这个过程至多经过 $n-1$ 步. 下面说明每一步仅需常数时间:若射线 r 的基 $b(r)$ 落在正则点上,那么射线 r 所在的射线束 B 要么是角体,要么是伪星,不可能是星(因为此时这条边的调整已经结束),角体的两个支撑柄或伪星的两个轴就是生成 r 的 3 条射线中的 2 条,第 3 条相对对应的边退化为零. 若射线 r 的基 $b(r)$ 不落在正则点上,那它就是 Steiner 点,射线束 B 的基线 $b(B)$ 对应两个正则点 x 和 y,而 $[x, b(r)]$ 和 $[y, b(r)]$ 是生成 r 的 2 条射线. 不妨设第 3 条射线对应的射线束为 B_3,则 $[t(B_3), b(r)]$ 为生成 r 的第 3 条射线. 每个射线束的顶及射线束的基线对应的两个正则点可在构造 B 时储存起来. 由此得出调整阶段所需时间为 $O(n)$.

综上所述,有如下定理:

定理 2　算法的时间复杂性为 $O(n^2)$.

4　结束语

本文给出的计算已知拓扑下 4 度 Steiner 树的算法在油田集输站选址[5,6]等许多问题中有很好的应用价值. k 度 Steiner 树的算法设计值得进一步深入研究,它有着十分重要的理论意义和非常广泛的应用前景.

参考文献:

[1]　GilbertEN,PollakHO.Steinerminimaltrees.SIAMJ ApplMath,1968,16 (1):1～29.

[2]　TrietschD,Hwan gFK.Anim provedal gorithmfor Steinertrees.SIAMJA pplMath,1990,50 (2):244～263.

[3]　GareyMR,GrahamRL.Thecom plexit yofcom putin g Steinerminimaltrees.SIAMJA pplMath,1977,37 (4):835～859.

[4]　HwangFK,Wen gJ.F.Theshortestnetworkundera givento pology.JAl g,1992,13:468～488.

[5]　陈德泉. 原油集输系统规划优选研究. 优选与管理科学.1987,3 (3):1～8.

[6]　徐寅峰,刘自成. 关于最大权 k-子集分析问题. 高校应用数学学报,1994,9 (4):453～457.

(编辑　狄怀春　赵大良)