

近乎最佳的 Manhattan 型 Steiner 树近似算法

马军 杨波 马绍汉

(山东大学计算机科学系 济南 250100)

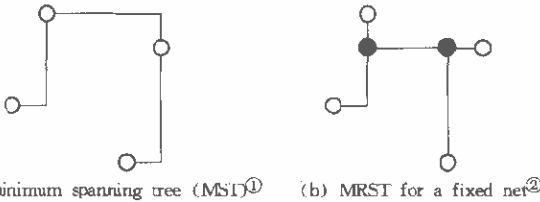
E-mail: majun@sdu.edu.cn

摘要 求解最佳的 Manhattan 型 Steiner 树问题(minimum rectilinear Steiner tree, 简记为 MRST 问题)是在 VLSI 布线、网络通信中所遇到的组合优化问题, 同时也是一个 NP-难解问题. 该文给出对该问题的 $O(n^2)$ 时间复杂性的近似算法. 该算法在最坏情况下的近似比严格小于 $3/2$. 计算机实验结果表明, 所求得的支撑树的平均费用与最佳算法的平均费用仅相差 0.8%. 该算法稍加修改, 可应用到三维或多维的 Manhattan 空间对 Steiner 问题求解, 且易于在并行与分布式环境下编程实现.

关键词 Steiner 树, 算法复杂性, 组合优化, 电路布线.

中图分类号 O157

MRST(minimum rectilinear Steiner tree)问题可简述为: 设 Z 为平面上给定的 n 个顶点集合, 对 Z 中任意两个顶点 $p_1=(x_1, y_1), p_2=(x_2, y_2)$, 其连线只限于水平与垂直直线. 其连接长度定义为 $dist(p_1, p_2)=|x_1-x_2|+|y_1-y_2|$ (称为 Manhattan 空间距离). 问题是如何添加平面上的一个点集 S , 以得到平面上的—棵树 T . T 为支撑平面顶点集合 $Z \cup S$ 的最小生成树, 这里, 支撑树 T 的费用 $COST(T)$ 被定义为所有 T 中的边的长度之和, T 被称为 Steiner 树; S 中的顶点被称为 Steiner 顶点. 若 S 为空集, T 被称为最小生成树(minimum spanning tree, 简称 MST). 图 1 给出了连接 4 个顶点的 MST 与其所对应的 Steiner 树.



Hollow dots represent the original point set Z , while solid dots represent the set S of added Steiner points^③.

①最小生成树, ②MRST的解,
③空心圆为 Z 的顶点, 实心圆为 S 集中的 Steiner 顶点.

Fig. 1
图1

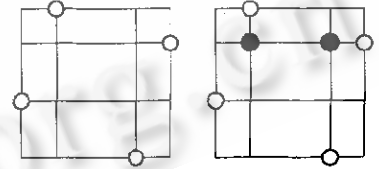


Fig. 2 Hanan's theorem
图2 关于Hanan定理的说明

MRST 问题是一个经典的组合优化问题, 在布线等领域得到了较为广泛的应用^[1-6]. 由于 MRST 的精确求解属于 NP-完全问题^[1], 故研究对 MRST 具有快速、高近似比的近似算法, 一直得到人们的重视. 首先, Hanan 证明了 S 为通过 Z 中 n 个顶点的水平与垂直直线所形成的交叉点集的子集, 如图 2 所示^[3]; Snyder 证明了该结论在更高维的 Manhattan 空间也成立^[4]; 其次, 人们证明了 MST 为 MRST 的一个相当好的近似解, 满足 $COST(MST)/COST(MRST) \leq 3/2$ ^[1]. 因此, 人们提出不少基于 MST 算法思想的求解 MRST 的近似算法^[1], 使

• 本文研究得到国家 863 高科技项目基金(No. 863-306-ZT06-01-4)资助. 作者马军, 1956 年生, 博士, 教授, 主要研究领域为算法, 人工智能, 并行计算. 杨波, 1975 年生, 硕士生, 主要研究领域为数据库, 算法. 马绍汉, 1938 年生, 教授, 博士生导师, 主要研究领域为算法, 人工智能.

本文通讯联系人: 马军, 济南 250100, 山东大学计算机科学系(新校)

本文 1998-08-31 收到原稿, 1999-03-03 收到修改稿

得所生成的支撑树的平均解费用可比 $COST(MST)$ 减少 7~9%^[1]. 然而, 这些算法在最坏情况下的近似比仍为 3/2. 因此, 是否存在最坏情况下近似比严格小于 3/2 的算法, 就成为一个公开的问题^[6].

本文给出一个新的 MRST 近似算法. 在最坏情况下, $COST(T)/COST(MRST) < 3/2$. 计算机上的实验结果表明, 新算法所产生的支撑树, 其平均费用可比 MST 改进 10.2%, 与文献[6]所给出的平均最佳算法的理论值相比, 仅平均相差 0.8%.

1 算法与实验结果

在下面的讨论中, 用 p, e 分别表示平面上的一个顶点和一条边. p 到 e 的距离 $dist(p, e)$ 定义为 $dist(p, p')$ 的最小值, 其中 p' 为 e 上的任意顶点. 由于边 e 可能为一条折线, 故用三元组 (p_1, p_2, p_3) 表示 e , 其中 p_1, p_3 分别表示 e 的两个端点, p_2 表示中间的拐弯点. 根据几何意义, 边 e, e' 之间的距离定义为 $\min\{dist(u, e'), dist(v, e) \mid u \text{ 为 } e \text{ 的三元组表示中的三点之一, 而 } v \text{ 则是 } e' \text{ 的三元组表示中的三点之一}\}$.

ST 算法. (Steiner 树近似求解算法)

输入: 给定平面上顶点集合 Z 的顶点的二维坐标 $(x_i, y_i), 1 \leq i \leq n$.

输出: 支撑 $Z \cup S$ 顶点的树 T 的边及 S 中的 Steiner 顶点.

// 使用的变量为: 对顶点 $i = (x_i, y_i)$, 用 $V(i)(1) = x_i; V(i)(2) = y_i$ 表示, 二维数组元素 $W(i, j)$ 为顶点 i, j 之间的 Manhattan 距离. 在三元组 (i, p_1, p_2) 中, P_1 和 P_2 分别表示 $Z \cup S - \{i\}$ 中距顶点 i 最近和次最近的顶点 ($p_1 = p_2$, 若 Z 中只有一个未被访问顶点). 函数 $length(i, p_1, p_2)$ 表示连接 3 个顶点的 MRST 子树的费用.

步骤 1.

(1.1) $S := \emptyset; L := \emptyset$. 对任意两个顶点 i, j , 标记未被访问. 根据任意两点 i, j 的二维坐标, 计算 $W(i, j)$. 根据数组 W , 计算三元组 (i, p_1, p_2) . 在如图 3 所示的 3 种连接方法中, 找出最短连线, 其长度记为 $length(i, p_1, p_2)$.

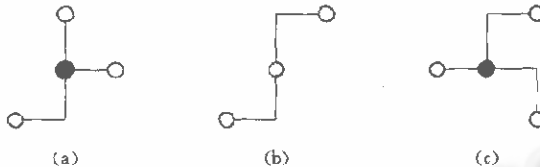


Fig. 3 The optimal connections for three vertices based on their positions
图3 基于三点相对位置的最佳连接方法

(1.2) 若 Z 中已无未被访问顶点, 转向步骤 2;

(1.3) 在未被访问的顶点中, 寻找顶点 i , i 具有目前最小的 $length(i, p_1, p_2)$ 值, 并按以下两种情况进行连接:

情况 1. 3 点均未被访问过, 根据它们的相互位置, 按图 3 所示的连接方法产生 3 点最佳连接; 标记 3 个顶点被访问过和属于子树 T_i ; 若按图 3(a) 或 (c) 进行连接, 则把产生的 Steiner 顶点送 S , 把产生的树边送 L .

情况 2. 否则, 寻找连接 i 到 P_1 或 L 中的某一边上的点的边 e , e 满足其长度为 $\min\{dist(i, P_1), dist(i, e'), dist(i, e'') \mid e' \text{ 为 } L \text{ 中与顶点相关联的边, } e'' \text{ 则是颠倒 } e' \text{ 的垂直、水平连接次序之后所得到的边}\}$. 把 e 送 L , 标记 i 与 P_1 属于相同的子树和被访问过标记; 若 e 的一个端点为 Steiner 顶点, 则把它送 S .

(1.4) 对每个新产生的 Steiner 顶点 u , 计算 u 与所有尚未访问顶点 j 的距离 $d(u, j)$;

if $dist(u, j) < dist(j, p_2)$ then {

if $dist(u, j) < dist(j, p_1)$ then $\{p_2 = p_1; p_1 = u\}$

else $p_2 = u$; endif;

修改对应的三元组 (j, p_1, p_2) 及函数 $length(j, p_1, p_2)$ 的值. } endif;

(1.5) 转向执行步骤 (1.2).

步骤 2. 设在步骤 1 后,形成了 k 棵子树 T_1, T_2, \dots, T_k . 计算边 e , 满足 e 可产生对平面两棵子树 T_i 与 $T_j, j \neq i$ 边集之间的最小连接边, $1 \leq i < j \leq k$. 通过 e 归并两棵子树. 送 e 到 L , 若 e 的一个端点为 Steiner 顶点, 送其到 S . 重复上述过程, 直到平面上只有一棵树 T .

步骤 3. 输出 L 中的边及 $COST(T)$. 结束.

根据传统的对 MRST 算法近似度的评价方法^[1,5,6], 我们对我们的算法与 MST 算法在同一输入下支撑树的费用进行对比, 见表 1. 其中 n 个顶点是由计算机随机产生在一个正方形平面内. a 表示正方形的边长, $r = COST(T)/COST(MST)$, s 表示算法使用的 Steiner 顶点个数.

Table 1 The performance of algorithm ST against MST algorithm
表 1 最小支撑树算法费用与本算法费用比较

n	a=100				a=200			
	COST(MST)	COST(T)	s	r	COST(MST)	COST(T)	s	r
10	244	219	4	0.897 541	600	516	6	0.860 000
20	365	322	10	0.882 192	829	744	10	0.897 467
30	478	414	14	0.866 109	1 093	934	14	0.854 529
40	541	489	17	0.903 882	1 032	960	18	0.930 233
50	628	567	17	0.902 866	1 221	1 082	17	0.886 159
60	631	594	23	0.941 363	1 298	1 178	26	0.907 550
70	720	654	28	0.908 333	1 478	1 335	25	0.903 248
80	762	688	32	0.902 887	1 537	1 400	34	0.910 865
90	797	724	32	0.908 407	1 614	1 467	33	0.908 922
100	872	793	45	0.909 404	1 726	1 577	43	0.913 673

表 1 表明 ST 算法所产生的支撑树的平均费用, 可使用 MST 进行近似求解改进 10.2%.

2 算法的理论分析

显然, 算法 ST 得到了一棵关于 $Z \cup S$ 的支撑树 T . 下面分析算法 ST 所产生的支撑树对 MST 的改进以及 ST 算法的时间复杂性.

引理 2.1. 当 $|Z| \leq 4$ 时, $COST(T) = COST(MRST)$.

证明: 显然, 当 $|Z| < 3$ 时, 引理成立; 当 $|Z| = 3$ 时, 去掉对称性, 只有如图 3 所示的 3 种关于 3 个顶点的相互位置. 设 C 表示围住 3 个顶点的长方形 R 的周长, 显然, 从图 3 所给出的连接方法中所选出的最短连接的长度恰为 $C/2$, 为构成了连接这 3 个点的最小费用连接, 故为一个 MRST.

当 $|Z| = 4$ 时, MRST 总为一个连接 3 点费用最少的子 MRST, 加上剩余一点与该子 MRST 边的水平或垂直连线组成, 这与我们的算法执行相同. 故引理也成立.

综上所述, 引理得证. □

由引理的证明过程不难看出, 对应于 3 个点的 MST, 均有 $COST(MST) \geq C/2$.

引理 2.2. $COST(T) \leq COST(MST)$; $COST(T) = COST(MST)$ 当且仅当 MST 是一个 MRST.

证明: 首先证明 $COST(T) \leq COST(MST)$. 根据引理 2.1, 当 $|Z| \leq 4$ 时, 引理成立. 故可在下面的讨论中, 假设 $|Z| > 4$.

由于 MRST 问题是几何平面上基于 Manhattan 距离的 Steiner 问题, MRST 实质上是用平行于 x 轴和 y 轴的最短线段连成. 根据 Hanan 的定理, MRST 可以通过对平面进行适当划分, 在每个小平面上求出局部的 MRST, 然后通过适当连接, 得到全局的 MRST. 在本算法中, 首先利用具有最短周长的可围住目前集合 $Z \cup S$ 中尚未访问过的任意 3 个顶点 v_1, v_2, v_3 的长方形 R 对平面分区, 在 R 内生成子树 T_i . 由引理 2.1 可知, T_i 为 R 内的 MRST. 若 $e = (v_4, v_5)$, 表示在算法结束时连接 T_i 与其他子树的一条边, 其中 v_4 为 T_i 的顶点; v_5 为其他子树上的顶点. 由 ST 算法步骤 (1.3) 中的情况 2 和步骤 2 的执行可知, e 为连接 T_i 与其他子树的最短边. 设 R' 表示围住 4 个顶点 v_1, v_2, v_3, v_5 的长方形, 由 R 的周长最短性和 e 为连接 v_5 到 T_i 的最短边以及 Kruskal 算法对边的

选取^[7]推知,在 R' 上生成的局部 MST 必为全局 MST 的一部分. 由于我们的算法是基于 Hanan 定理的执行,不难穷举出基于图 3 的所有 e 与 T_i 可能连接的情况,总有算法在 R' 上生成的子树费用 $\leq R'$ 上对应的局部 MST 的费用. 由于 R 的任意性以及 ST 算法步骤 2 中,对森林 F 的归并过程中,总是选择连接两棵子树的最短边进行归并,可推出 $COST(T) \leq COST(MST)$.

根据引理 2.1 和算法 ST 的执行,若所产生的支撑树 T 为一个 MST,当且仅当在算法结束后,未产生任何 Steiner 顶点;而只要 MST 不是 MRST 的解,算法 ST 必然会找到一个 Steiner 顶点,使得 $COST(T) < COST(MST)$.

基于上述讨论,引理得证. □

定理 2.1. 设 $r = COST(T)/COST(MST)$, 则 $2/3 \leq r \leq 1$.

证明:根据 MRST 的定义, $COST(T) \geq COST(MRST)$. 因已知 $COST(MST) \leq 3/2 COST(MRST)$, 故有

$$r = \frac{COST(T)}{COST(MST)} \geq \frac{2COST(MRST)}{3COST(MRST)} = 2/3. \quad \square$$

定理 2.2. $COST(T)/COST(MRST) < 3/2$.

证明:若对给定的 n 个顶点, $COST(T) < COST(MST)$, 则

$$\frac{COST(T)}{COST(MRST)} = \frac{COST(T)}{COST(MST)} \cdot \frac{COST(MST)}{COST(MRST)} < 3/2;$$

否则,基于引理 2.2, $COST(T)/COST(MRST) = 1 < 3/2$. □

定理 2.3. ST 算法的时间复杂性为 $O(n^3)$.

证明:显然,在 ST 算法中,步骤(1.1)的操作数为 $O(n^2)$,并且步骤(1.2)~(1.4)的操作数为 $O(n)$. 为减少步骤 2 的时间复杂性,我们还在步骤 1 的开始部分,对 Z 中的每一顶点 v ,计算 v 与 $Z - \{v\}$ 顶点的最短连接边. 每当一棵新子树生成时,立即计算该子树与其他子树相连时的最短连接边. 即使加上上述计算,步骤 1 的时间复杂性仍为 $O(n^2)$.

在步骤 2 中,基于上述已在步骤 1 中所进行的计算,我们可在 $O(n)$ 时间找出连接两棵子树的最短边. 显然,可在 $O(1)$ 时间内完成两棵子树的归并. 因为对每条边可用 3 个顶点表示,每个顶点均保留与其他子树边相连的最短边和 L 中的边数为 $O(n)$,故对归并后的新子树计算与其他树边相连的最短边时,也可在 $O(n)$ 时间内完成. 因此,步骤 2 的一次循环可在 $O(n)$ 时间内完成. 由于最多有 $O(n)$ 次循环,故步骤 2 的时间复杂性为 $O(n^2)$.

因产生的 Steiner 顶点的个数不会超过 n ,故生成树的边数仍为 $O(n)$,所以步骤 3 的时间复杂性为 $O(n)$.

由上述讨论,定理得证. □

本算法较文献[6]的算法复杂性减少了 $O(n)$ 因子.

3 结束语

本文给出一个求解 MRST 的新的近似算法,与以往的结果相比,该算法在最坏情况下的近似比严格小于 $3/2$,并且具有较好的时间复杂性. 计算机运行结果表明,所产生的支撑树的费用,仅比平均最佳算法的理论值相差 0.8%,不失为一种实用算法. 基于 Snyder 定理,对新算法稍作修改,便可适合三维和多维的 Manhattan 空间对 MRST 问题的求解. 由于新算法中的主要计算为取最小值操作,该算法也非常便于在并行与分布式环境下实现^[8].

致谢 本文作者对初审与复审老师提出的建设性的修改意见深表感谢.

参考文献

- 1 Hwang F K, Richards D S, Winter P. The Steiner Tree Problem. The Netherlands, North-Holland, 1992. 20~200
- 2 Maxemchuk N F. Video distribution on multicast networks. IEEE Journal on Selected Areas in Communications, 1987, 15 (3):357~372

- 3 Hanan M. On Steiner's problem with rectilinear distance. *Journal on Applied Mathematics*, 1966,14(6):255~265
- 4 Synder T L. A simple and faster algorithm for the rectilinear Steiner problem in general dimension. In: Kavi R ed. *Proceedings of the ACM Symposium on Computational Geometry*. New York: ACM Press, 1990. 1340~1345
- 5 Kahng A B, Robins G. A new class of iterative Steiner tree heuristics with good performance. *IEEE Transactions on Computer Aided Design*, 1992,11(7):893~902
- 6 Salowe J S, Warme D M. An exact rectilinear Steiner tree algorithm. In: Yao Y Y ed. *Proceedings of the IEEE International Conference on Computer Design*. Cambridge, MA: IEEE Press, 1993. 472~475
- 7 Aho A V, Hopcroft J E, Ullman J D. *The Design and Analysis of Computer Algorithms*. Reading, MA: Addison-Wesley Publishing Company, 1975. 172~176
- 8 Ma Jun, Takaoka T, Ma Shao-han. Parallel algorithms for a class of graph theoretic problems. *Transactions of Information Processing Society of Japan*, 1994,35(7):1235~1240

A Near-Optimal Approximation Algorithm for Manhattan Steiner Tree

MA Jun YANG Bo MA Shao-han

(Department of Computer Science Shandong University Ji'nan 250100)

Abstract The minimum rectilinear Steiner tree (MRST) problem is an NP-complete problem which arises in VLSI wiring, network routing and many combinatorial optimization problems. In this paper, an $O(n^2)$ time complexity approximation algorithm for MRST is proposed. The approximation ratio of the algorithm is strictly less than $3/2$. The computer verification of the algorithm shows that the costs of the produced spanning trees are only 0.8% away from the optimal. In addition, this algorithm can be revised for multidimensional Manhattan space and implemented in parallel/distributed environments easily.

Key words Steiner tree, complexity theory, combinatorial optimization, ci © 中国科学院软件研究所 <http://www.jos.org.cn>