# 一类实际网络中的最小截算法*

张宪超+, 万颖瑜, 陈国良

(中国科学技术大学 计算机科学与技术系,安徽 合肥 230027)

(国家高性能计算中心(合肥),安徽 合肥 230027)

# Approaches to the Minimum Cut Problem in a Class of Practical Networks

ZHANG Xian-Chao+, WAN Ying-Yu, CHEN Guo-Liang

(Department of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China)

(National High Performance Computing Center at Hefei, Hefei 230027, China)

+ Corresponding author: E-mail: xczhang081@sina.com

http://www.qqtechnology.com

Received 2002-04-12; Accepted 2002-09-11

**Zhang XC, Wan YY, Chen GL. Approaches to the minimum cut problem in a class of practical networks. *Journal of Software*, 2003,14(5):885~890.**

http://www.jos.org.cn/1000-9825/14/885.htm

**Abstract**: The minimum cut problem between two distinguished nodes in a undirected planar network with limited capacities on both nodes and edges is discussed. The traditional method reduces the node-edge-capacity problem to an only-edge-capacity problem. But this method does not maintain the planarity of a planar network, so the special qualities of planar networks can not be used. The traditional algorithm runs in time $O(n^2\log n)$. In this paper, approaches that fully use the planarity of planar networks are presented. For an $s$-$t$ network in which the source and the sink are on a same face, the minimum cut problem to the shortest path problem in a planar graph is reduced, thus an algorithm that runs in time $O(n)$ is obtained. For a common planar network, another method that reduces the node-edge-capacity problem to an only-edge-capacity problem is presented. This reduction method does not destroy the planarity of a planar network, so that the problem can be solved in time $O(n\log n)$.

**Key words**: combinatorial optimization; network optimization; maximum flow; minimum cut; planar graph

**摘 要**: 讨论了节点和边都有容量限制的无向平面网络中的两点间的最小截问题.传统方法是把节点和边都有容量的网络中的最小截问题转化为只有边有容量的问题,但该方法用在平面网络时不能保持网络的平面性,因此网络的平面性不能得到利用.使用传统方法的计算时间为 $O(n^2\log n)$(其中 $n$ 为网络的节点数).给出了可以充分利用网络平面性的方法.对源和汇共面的 $s$-$t$ 平面网络,把最小截问题转化为平面图上两点间的最短路径问题,从而可以得到 $O(n)$时间的算法;对一般的平面网络,给出了新的将节点和边都有容量的问题转化为仅边有容量问题的方法,这种转化方法不破坏网络的平面性,从而可以利用平面网络中仅边有容量问题的计算方法,使原问

* Supported by the National Grand Fundamental Research 973 Program of China under Grant No.G1999032700 (国家重点基础研究发展规划(973))

**第一作者简介**: 张宪超(1971－),男,辽宁朝阳人,博士,主要研究领域为网络优化,组合优化,算法设计与分析,并行分布式计算.

题在 $O(n\log n)$时间内获得解决.

**关键词**: 组合优化;网络优化;最大流;最小截;平面图

**中图法分类号**: TP301 **文献标识码**: A

最小截问题是一个经典的组合优化问题.它和其对偶问题——最大流问题是计算机科学和运筹学等领域的重要内容,在电力、交通、通信、计算机网络等领域有着广泛的应用.文献[1]对此作了很好的总结.对一般网络中的最小截,目前最好算法的时间复杂度为 $O(nm\log(n^2/m))$,其中 $n$ 为网络的节点数,$m$ 为边数[2~4].

由于在交通、通信、VLSI 等许多应用领域中存在大量平面网络[5],平面网络中的问题被广泛地进行了研究[5~12].通过开发平面网络的特殊性质,可以得到比一般算法更快的算法.对源和汇共面的平面网络(称为 $s$-$t$ 网络),可以在 $O(n)$ 时间内获得最小截[12].对一般的平面网络,现在最好的时间复杂度为 $O(n\log n)$ [5].

目前,在最小截和最大流问题的研究中只考虑了网络的边有容量的情况.在实际的网络中,边和节点都会有容量的限制.由于可以通过把网络中有容量的节点分裂成一条边的简单方法,把节点和边都有容量的网络中的最小截问题转化为仅边有容量的问题[1,13,14],因此对一般网络来说,只需考虑仅边有容量就可以了.

但是对于平面网络来说,这种转化不能保持网络的平面性[14].这样,对于平面网络中节点和边都有容量的问题,目前最好算法的时间复杂度为 $O(n^2\log n)$ (即 $O(nm\log(n^2/m))$ [3],注意到平面网络中 $m=O(n)$ ).

本文在仅边有容量的平面网络中的最小截问题算法的启发下,得到了节点和边都有容量的无向平面网络中的最小截算法.在 $s$-$t$ 网络中,把最小截问题转化为平面图中的两点间的最短路径问题,这样,利用文献[12]中 $O(n)$ 时间的平面图最短路径算法,可以在 $O(n)$ 内计算出最小截.对一般的平面网络,得到了一个新的把节点和边都有容量的问题转化为仅边有容量的问题的方法,新的转化方法保持了网络的平面性.这样,利用文献[5]中仅边有容量的问题的算法,可以使问题在 $O(n\log n)$ 时间内获得解决.

## 1 问题描述和预备知识

**定义 1**. 本文所指的网络是一个无向加权图 $G=(V,E)$,其中 $V$ 为 $G$ 的节点集合, $E$ 为边集合.在 $G$ 中指定两个节点 $s$ 和 $t$,分别称为源和汇.其他节点称为中转节点.中转节点 $v$ 上的权 $c(v)\geq 0$ 称为 $v$ 的容量.边 $e=(i,j)\in E$ 上的权 $c(i,j)$ 称为该边的容量.$G$ 称作网络的底图.在不混淆的情况下,直接用底图代表该网络.

**定义 2**. 可以嵌入平面且使任意两条边互不交叉的图称为平面图.底图为平面图的网络称为平面网络.一个平面图 $G$ 嵌入到另一个平面中可以把平面划分成若干个区域,每个区域称为 $G$ 的一个面,其中有一个无限的面,称为外面.若平面网络 $G$ 中的源 $s$ 和汇 $t$ 在同一个面上,则称它为 $s$-$t$ 网络.

**定义 3**. 令 $A$ 为有向边 $\{\langle i,j\rangle\in A\,|\,(i,j)\in E\}$ 的集合.称从集合 $A$ 到实数集 $R$ 上的函数 $f$:$A\rightarrow R$ 为网络 $G$ 上的一个流,若以下条件成立:

$$\forall\langle i,j\rangle\in A,\ 0\leq f(\langle i,j\rangle)\leq c(i,j), \tag{1}$$

$$\forall i\in V-\{s,t\},\ \sum_{\langle k,i\rangle\in A} f(\langle k,i\rangle)=\sum_{\langle i,j\rangle\in A} f(\langle i,j\rangle)\leq c(i), \tag{2}$$

则称相应的 $\sum_{\langle i,t\rangle\in A} f(\langle i,t\rangle)$ 为流 $f$ 的流量.最大流是指具有最大流量的流.

**定义 4**. 集合 $C\subset E\cup V-\{s,t\}$ 称为网络 $G$ 的一个截,若 $G$ 中任意一条从 $s$ 到 $t$ 的路径($s$-$t$ 路径)经过 $C$ 的一个元素.截 $C$ 中元素容量的和称为该截的容量.容量最小的截称为最小截.仅由边构成的截称为边截.

**定理 1(最大流-最小截定理)**[1,13,15].网络 $G$ 中的最大流量等于它的最小截的容量.

最大流-最小截定理说明了最大流和最小截问题是一对对偶问题.

## 2 用带链对偶图中的截环表示原图中的截

网络 $G$ 中的一个截截断了 $G$ 中任意一条 $s$-$t$ 路径.注意到:

**命题 1**. 如果在平面中用一个环把两个点 $s$ 和 $t$ 分开,则它可以截断平面上任何一条从 $s$ 到 $t$ 的路径(如图 1

所示).

命题 1 是拓扑学中的一个基本结论.我们希望利用这一性质,找到可以代表平面网络中截的环.

**定理 2**[16]. 设 $v$ 是平面图 $G$ 的任意一个节点,可以把 $G$ 嵌入到一个平面,使 $v$ 在外面上.

根据定理 2,总可以把平面网络 $G$ 嵌入到一个平面,使源 $s$ 在外面上.以下总考虑网络的这种平面嵌入.

首先考虑节点没有容量的网络.在这种网络中,最小截只在边截中出现.

**定义 5**. 对仅边有容量的平面网络 $G$ ,构造 $G$ 的对偶图 $G'$ .对任意边 $e\in G$ ,定义 $e$ 的对偶边 $e'$ 的权值 $c(e')=c(e)$ ,作为 $e'$ 的长度.称 $G'$ 中包围原图 $G$ 的汇 $t$ 的环为边截环,如图 2 所示(加粗部分代表一个边截和相应的边截环).

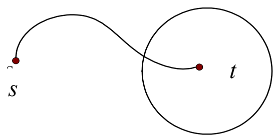


Fig.1 A circle in the plane cut any $s$-$t$ path

图 1 平面上任何一个环截断任意一条 $s$-$t$ 路径

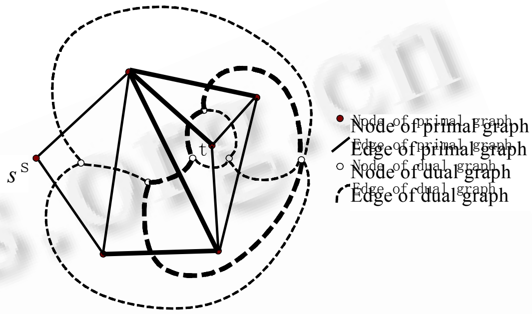


Fig.2 Graph $G$ and its dual graph $G'$

图 2 图 $G$ 和它的对偶图 $G'$

**定理 3**[8]. $G$ 中的边截和 $G'$ 中的边截环是一一对应的,且每个边截环的长度等于相应边截的容量.因此,$G$ 中的最小截的容量等于 $G'$ 中最短的边截环的长度.

定理 3 是许多仅边有容量平面网络中的最小(边)截算法的基础.

我们希望在节点和边都有容量的平面网络中得到类似的结论.对节点和边都有容量的网络，如果希望用平面中的“环”来代表网络中的截,则这些“环”不仅有可能从边上截断网络中所有从 $s$ 到 $t$ 的路径,还有可能从节点处截断这些路径.为此,对节点和边都有容量的网络 $G$,先构造它的对偶图的 $G'$ .在 $G'$ 的基础上定义一个新图 $G''$ :在 $G'$ 中加入 $G$ 的所有中转节点;对任一中转节点 $v\in V-\{s,t\}$ ,它包含在 $G'$ 的一个面 $f_v$ 中,在 $v$ 和 $f_v$ 的所有边界节点之间连一条边,定义这些边的长度为 $\frac{c(v)}{2}$(如图 3 所示).

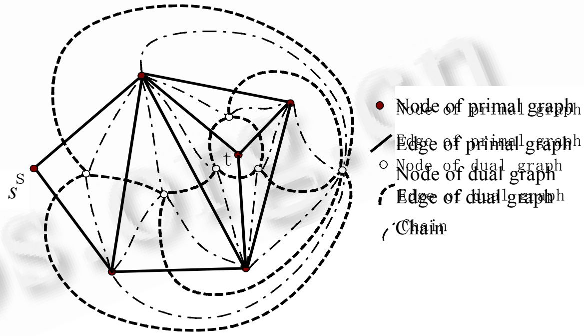


Fig.3 Primal graph $G$ and its chained-dual graph $G''$

图 3 原图 $G$ 和它的带链对偶图 $G''$

**定义 6**. 按照上述规则得到的图 $G''$ 仍为平面图,称它为 $G$ 的带链对偶图,新加入的边称为链.类似边截环的定义,称 $G''$ 中包围原图汇 $t$ 的环为截环.

**定理 4.** $G$ 中最小截的容量等于 $G''$ 中最短的截环的长度.

证明:一方面,对 $G''$ 中的任一截环 $C''$ 上的边,建立如下的对应关系:若该边是一条普通的边,取它在 $G$ 中的对偶边和它对应.若该边是一条链,则环 $C''$ 中必还有一条链和它相连于 $G$ 中的一个节点.取该节点和这两条链对应.这样,截环 $C''$ 就和 $G$ 中边集合和节点集合的并集的一个子集 $C\subset V\cup E$ 取得了一个对应关系.注意到 $G$ 中的任一条 $s$-$t$ 路径必经过 $C$ 中的一个元素,故 $C$ 是 $G$ 中的一个截,它的容量是截环 $C''$ 的长度.

另一方面,对 $G$ 中的任意一个截 $C$,若 $C$ 是一个边截,则由定理 3,可以在 $G'$ 中找到一个边截环,也就是 $G''$ 中

的一个截环,它的长度为 $C$ 的容量.若 $C$ 是一个包含节点的截,对于 $C$ 中的每个元素,若它是一条边,则它对应于 $G'$ 中的一条边;若它是一个节点,则它对应于 $G'$ 中的一个面.下面说明所有这些边和面在 $G'$ 中构成一个包围汇 $t$ 的“环”.

由截的定义可知,$C$ 截断了所有 $s$-$t$ 路径.每条 $s$-$t$ 路径总是先从源 $s$ 出发,经过截 $C$,再到达汇 $t$.我们把从 $s$ 到 $C$ 那条子路径上,和 $C$ 中的某个节点或边相关联的边称为 $C$ 的入边;把在从 $C$ 到 $t$ 的那条子路径上,与 $C$ 中的某个节点或边相关联的边称为 $C$ 的出边.考虑 $C$ 中某节点 $v$,如果用所有与 $v$ 相关联的出边代替 $v$,则这些出边和 $C$ 中其他元素一起构成一个新的截.依次类推,把 $C$ 中所有的节点都用相应的出边代替,则可以得到一个全部由边构成的边截.由定理 3 可知,这个边截对应于 $G'$ 中的一个边截环,称它为 $C$ 的出边截环.在这个出边截环中,与截 $C$ 中的一条边 $e$ 对应的部分就是该边在 $G'$ 的对偶边 $e'$;与 $C$ 中某节点 $v$ 对应的部分为 $v$ 的所有出边的对偶边,它们是 $G'$ 中和节点 $v$ 对应的面 $f_v$ 的边界的一部分.因此,在 $G'$ 中,与原图的截 $C$ 中的所有元素对应的边和面构成了一个包围汇 $t$ 的“环”,称这个“环”为 $G'$ 中的一个超截环,如图 4 所示(加粗部分代表相应的出边截环).

对于超截环中的一个面,它通过一个节点(或一条边)和超截环中的另外一个元素相连,不妨称该节点(或该边的两个端点)为键点.对每个这样的面 $f_v$,在它的每个键点和它所对应的原图节点 $v$ 之间连上一条链.这样可以得到 $G''$ 中的一组截环(如图 5 所示).不难看出,这些截环的长度和相应的截的容量相等.

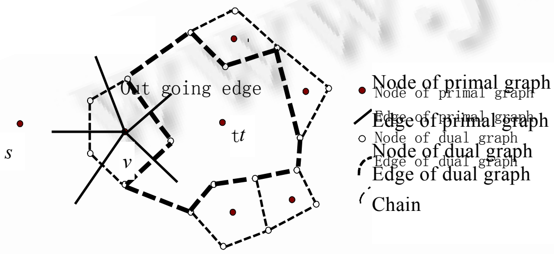


Fig.4 A super-cut-circle

图 4 一个超截环

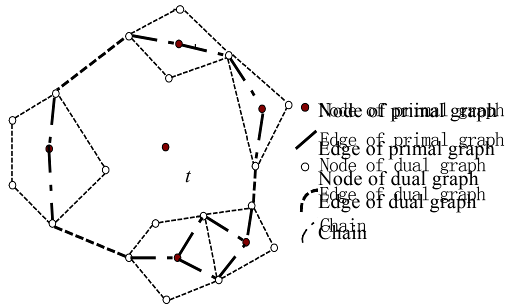


Fig.5 The super-cut-circle in Fig.4 determines two cut-circles

图 5 图 4 的超截环确定了两个截环

这样,我们说明了对 $G''$ 中的任一截环,可以在 $G$ 中找到一个截,它的容量等于该截环的长度;而对于 $G$ 中的任一截,可以找到 $G''$ 中的一组截环,它们的长度为该截的容量.于是得出结论:$G$ 的最小截的容量等于 $G''$ 中最短的截环长度. □

定理 4 说明了原图 $G$ 最小截问题可以转化为它的带链对偶图中最短截环问题.

## 3 $s$-$t$ 网络中的最小截算法

$s$-$t$ 网络有一个特殊的性质,即它允许边 $(s,t)$ 的存在而不破坏网络的平面性.下面总假设边 $(s,t)$ 是存在的.若 $(s,t)$ 不存在,可以在网络中加上一条边 $(s,t)$ 并令它的容量为 0,这样不会影响网络中的最小截的容量.

下面考虑 $s$-$t$ 网络 $G$ 的带链对偶图 $G''$ 中的截环.注意到一个截环要截断 $G$ 中所有的 $s$-$t$ 路径,因此每个截环都包含了边 $(s,t)$ 的对偶边,记为 $(s',t')$,它截断边 $(s,t)$ (如图 6 所示).

由于 $(s',t')$ 是所有截环的公共边.在这些截环中去掉边 $(s',t')$,则每个截环的剩余部分在

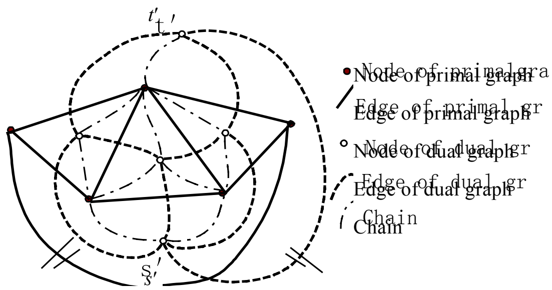


Fig.6 The minimum cut problem in $s$-$t$ network can be reduced to the shortest path problem

图 6 $s$-$t$ 网络中的最小截问题可以转化为最短路径问题

$G''$ 中构成一条从 $s'$ 到 $t'$ 的路径.因此,最短截环问题就可以转化为求从 $s'$ 到 $t'$ 的最短路径问题.

构造一个平面图 $G$ 的对偶图 $G'$ 需要 $O(n)$ 时间.在对偶图 $G'$ 的基础上构造带链对偶图 $G''$ 需要 $O(n)$ 的时间.利用文献[12]中的算法,可以在 $O(n)$ 的时间内计算平面图中两点间的最短路径.

**定理 5**. 可以在 $O(n)$ 时间内求出节点和边都有容量限制的 $s$-$t$ 网络中的最小截.

## 4 在一般平面网络中求最小截

对源和汇不在同一面上的平面网络,$s$ 和 $t$ 之间是不存在边的.在这样的网络的带链对偶图中直接计算长度最小的截环是比较困难的.幸运的是,目前已经有了很有效的求解仅边有容量的网络中的最小截(最大流)算法,因此这里希望把节点和边均有容量的网络中的最小截问题转化为仅边有容量的网络中的问题.

定理 3 指出,一个网络中的边截和它的对偶图中的边截环是一一对应的.因此,不但可以用长度最小的边截环来求得最小边截,反过来同样可以用最小边截来求长度最小的边截环.

考虑网络 $G$ 的带链对偶图 $G''$,如果再构造它的对偶图 $G_1$,则相对于 $G_1$ 来说,$G''$ 中的所有截环都是边截环.只要求出 $G''$ 的对偶图 $G_1$ 的最小边截,就可以求得 $G''$ 中长度最小的截环,也就求出了原图 $G$ 的最小截.

到此为止,我们已经找到了在节点和边都有容量的一般平面网络 $G$ 中求最小截的方法,即它可以转化为另一个仅边有容量的平面网络 $G_1$ 中的最小边截问题.我们希望找到 $G_1$ 和 $G$ 的关系,从而得到更简单的转化方法.

**引理 1**. 任意给定平面图中的一个节点,与它相关联的所有边的对偶边在对偶图中构成一个环.

证明:与一个节点关联的所有边把平面分割成若干个区域,这些区域都是对偶图的节点.注意到在对偶图中,这些区域中所有两个相邻的区域之间都存在一条边,因此这些边构成一个环. □

**定义 7**. $G''$ 中所有与原图 $G$ 的中转节点 $v$ 相关联的链将 $v$ 所对应的面 $f_v$ 又分成若干个区域,不妨称为 $f_v$ 的子面.由引理 1 可知,这些链的对偶边在 $G_1$ 中构成一个环,称为链环,每个子面构成了链环的一个节点.

注意到每个子面都以 $G''$ 中一个非链边为边界,因此在 $G_1$ 中,链环的每个节点和一条非链边的对偶边相关联.在 $G''$ 中,原图 $G$ 的源 $s$ 和汇 $t$ 所对应的面 $f_s$ 和 $f_t$ 没有被分割,它们分别对应于 $G_1$ 中的两个节点,$s_1$ 和 $t_1$.这样,我们就建立了 $G$ 和 $G_1$ 的对应关系:$G$ 中的每条边对应于 $G_1$ 中的一条边,它们的容量相等;$G$ 的每个中转节点对应于 $G_1$ 中的一个链环,链环上每条边的容量为该节点容量的一半;$G$ 的源和汇分别对应于 $G_1$ 中的两个节点,它们分别作为 $G_1$ 的源和汇(如图 7 所示).

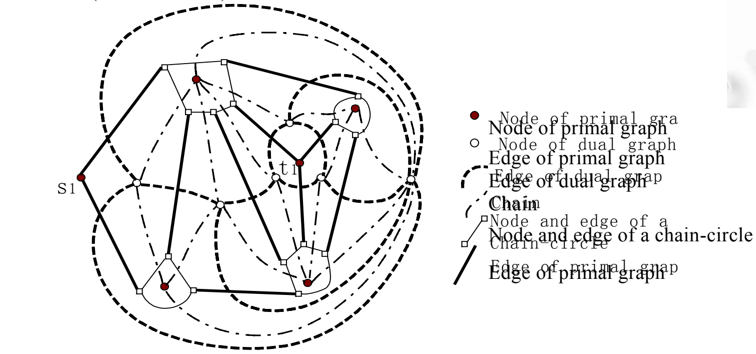


Fig.7 Chained-Dual graph $G''$ and its dual graph $G_1$

图 7 带链对偶图 $G''$及其对偶图 $G_1$

**定义 8**. 上述与原网络 $G$ 有相同最大流量的网络 $G_1$ 称为原网络的等量网络.

若原网络 $G$ 的节点数为 $n$,则它的等量网络 $G_1$ 的节点数为 $O(n)$,且可以在 $O(n)$ 时间内构造 $G_1$.注意到原网络中截的元素和相应等量网络中边截的元素有明确的对应关系,结合前面的讨论,有:

**定理 6**. 可以在 $O(n\log n)$ 时间内求得一个节点和边都有容量的一般平面网络中的最小截.

## 5 结 论

对平面网络来说,传统的解决节点和边都有容量的最小截问题的方法不能保持其平面性,计算复杂度很高,为 $O(n^2\log n)$.本文给出了可以利用网络平面性的方法,把 $s$-$t$ 网络和一般平面网络上节点及边都有容量的最小截问题的计算时间复杂度分别降低到 $O(n)$ 和 $O(n\log n)$.本文的方法可以计算网络中的最小截或最大流的流量.在一些应用中还需要计算网络中的最大流函数.如何把等量网络中的最大流函数转化为原网络中的最大流函数,是一个十分有意义的问题.对此,文献[17]中提供的伪流算法具有一定的启发性.

**References**:

[1] Ahujia RK, Magnanti TL, Orlin JB. Network Flows: Theory, Algorithms and Applications. Prentice-Hall, 1993.

[2] Goldberg AV. Recent developments in maximum flow algorithms. In: Proceedings of the 6th Scandinavian Workshop on Algorithm Theory. Stockholm, 1998.

[3] Goldberg AV, Tarjan RE. A new approach to the maximum flow problem. Journal of the Association for Computing Machinery, 1988,35(4):921~940.

[4] Zhang XC. Study on maximum flow problem algorithms [Ph.D.Thesis]. Hefei: University of Science and Technology of China, 2000 (in Chinese with English abstract).

[5] Weihe K. Maximum $(s,t)$-flows in planar network in $O(|V|\log|V|)$ time. Journal of Computer and System Sciences, 1997,55(3): 454~475.

[6] Itai A, Shiloach Y. Maximum flows in planar networks in planar networks. SIAM Journal of Computing, 1979,8(1):135~150.

[7] Hassion R. Maximum Flow in $(s,t)$ planar networks. Information Processing Letters, 1981,13(3):107.

[8] Reif JH. Minimum $s$-$t$ cut of a planar undirected network in $O(n\log^2 n)$ time. SIAM Journal of Computing, 1982,12(1):71~81.

[9] Hassion R, Johnson DB. An $O(n\log^2 n)$ algorithm for maximum flow in undirected planar networks. SIAM Journal of Computing, 1985,14(3):612~624.

[10] Johnson DB. Parallel algorithms for minimum cuts and maximum flows in planar networks. Journal of the Association for Computing Machinery, 1987,34(4):950~967.

[11] Frederickson GN. Fast algorithms for shortest paths in planar graphs, with applications. SIAM Journal of Computing, 1987,16(6): 1004~1022.

[12] Klein RSB, Rauch-Henzinger M, Subramanina S. Faster shortest-path algorithms for planar graphs. Journal of Computer and System Sciences, 1997,55(3):3~23.

[13] Granot F, Hassion R. Multi-Terminal maximum flows in node-capacited networks, Discrete Applied Mathematics, 1986,13(1): 157~163.

[14] Bondy JA, Murty USR. Graph Theory with Applications. North-Holland, 1977.

[15] Ford LR, Fulkerson DR. Maximum flow through a network. Canadian Journal of Mathematics, 1956,8:399~404.

[16] Wang SH. Graph Theory and algorithms. Hefei: University of Science and Technology of China Press, 1994 (in Chinese).

[17] Hochbuam D. The pseudoflow algorithm and the pseudo flow-based simplex for the maximum flow problem. In: Proceedings of the Integer Programming and Combinatorial Optimization the 6th International Conference (IPCO). Houston: 1998. 325~337.

附中文参考文献:

[4] 张宪超.网络最大流问题算法研究[博士学位论文].合肥:中国科学技术大学,2000.

[16] 王树禾.图论及其算法.合肥:中国科技大学出版社,1994.