第42卷 第6A期 2015年6月
计 算 机 科 学
Computer Science
Vol. 42 No. 6A
June 2015

带上下限的网络最大流的算法

赵晓蓉

(黔南民族师范学院计算机科学系 都匀 558000)

摘 要 对弧的容量带上下界约束的最大流问题进行了讨论，给出带上下界约束的最大流问题的数学模型和求解算法，即将求解带上下限的网络最大流问题转化为求解网络的最小费用最大流问题，并给出了求解算法的实例。

关键词 网络最大流，容量，模型，算法

中图法分类号 TP393.0 文献标识码 A

Taking Lower Network Maximum Flow Algorithm

ZHAO Xiao-rong

(Department of Computer Science, Qiannan Normal College for Nationalities, Duyun 558000, China)

Abstract This paper aimed at discussing the maximum flow problem on the capacity of the arc with upper and lower bounds, as well as giving an algorithm and a mathematical model associated. In other words, in the situation where the network has upper and lower limits, the maximum flow problem was transferred to minimum cost flow problem. Furthermore, an algorithm used to solve the problem was provided.

Keywords Maximum network flow, Capacity, Model, Algorithm

1 引言

网络最大流问题是一个既属于图论又属于运筹学的问题[1,2]，许多学者对它进行了大量的研究，通常讨论网络最大流问题中弧的容量只有上界的约束，可行流是存在的[3,4]，Ford-fulkerson于1956年提出的标号算法是被广泛应用的一种经典算法。但实际应用中网络是很复杂的，容量受多种因素的影响，很多情况下不能直接用标号算法来求网络的最大流。对带上下限的网络最大流问题，已有学者对它进行了研究并提出了自己的算法，文献[5]对带上下界网络最大流与最小割问题提出了数值算法。在已有的研究基础上，本文给出带上下界约束的最大流问题的数学模型，对模型的可行流求解进行了讨论，最后将求解带上下限的网络最大流问题转化为求解网络的最小费用最大流问题。

2 基本概念与数学模型

2.1 基本概念

定义1[6](网络) 给定一个有向图 $D=(V,A)$，在 V 中指定了一点称为发点(记为 v_s)，而另一点称为收点(记为 v_t)，其余的点叫中间点，对于每一条弧 $(v_i,v_j)\in A$ 对应有一实数 $c(v_i,v_j)\geqslant 0$(简记为 c_{ij})，称为弧的容量。这样的有向图称为一个网络，记作 $D=(V,A,C)$。

在弧集合 A 上定义一个函数 $f=\{f(v_i,v_j)\}$，称 $f(v_i,v_j)$ 为弧 (v_i,v_j) 上的流量，简记为 f_{ij}。

定义2(可行流) 满足下述条件的流 f 称为可行流，

(1)容量限制条件：对每一条弧 $(v_i,v_j)\in A$ 有 $0\leqslant f_{ij}\leqslant c_{ij}$；

(2)平衡条件：

$$\sum_j f_{ij}-\sum_j f_{ji}=0,\ \forall v_i\in V\backslash\{v_s,v_t\}$$

$$\sum_j f_{sj}=\sum_j f_{jt}=v(f)$$

式中，$v(f)$ 为流 f 的流量，网络 D 中的最大流是指 D 中流值最大的可行流。

数学模型为：

$$\text{Max } v(f)$$

$$\text{s.t.}\begin{cases}\sum_j f_{sj}-\sum_j f_{js}=v(f)\\ \sum_j f_{tj}-\sum_j f_{jt}=-v(f)\\ \sum_j f_{ij}-\sum_j f_{ji}=0,\ \forall v_i\in V\backslash\{v_s,v_t\}\\ 0\leqslant f_{ij}\leqslant c_{ij},\ \forall (v_i,v_j)\in A\end{cases}\tag{1}$$

定义3(带上下限的网络) 将弧上赋于两个权的简单连通有向图 $D=(V,A,\underline{C},\overline{C})$ 称为容量带上下限的有向网络，其中发点为 v_s、收点为 v_t，其余的点为中间点；$\underline{C}$ 为弧的容量下界集合，$\underline{c_{ij}}$ 是弧 (v_i,v_j) 的容量下界的大小；$\overline{C}$ 为弧的容量上界集合，$\overline{c_{ij}}$ 是弧 (v_i,v_j) 的容量上界的大小($\underline{c_{ij}}\leqslant\overline{c_{ij}}$)。

定义4(带上下限网络的可行流) 设 $f=\{f_{ij}\}$ 是 D 的任意流，f_{ij} 是弧 (v_i,v_j) 上的流量，f 为 D 中的可行流是指 f 满足：

(1)流量限制条件：$\underline{c_{ij}}\leqslant f_{ij}\leqslant\overline{c_{ij}},\ \forall (v_i,v_j)\in A$；

(2)平衡条件：$\sum_j f_{ij}-\sum_j f_{ji}=0,\ \forall v_i\in V\backslash\{v_s,v_t\}$

$$\sum_j f_{sj}=\sum_j f_{jt}=v(f)$$

其中，$v(f)$ 表示 D 中可行流 f 的流值，D 中的最大流是指 D 中流值最大的可行流。

2.2 容量带上下限的最大流的数学模型

$$\text{Max } v(f)$$

本文受贵州省科学技术厅联合基金项目(黔科合J字LKQS[2013]29号)资助。

赵晓蓉(1961－)，女，副教授，主要研究方向为计算机数学和算法，E-mail:490296523@qq.com。

$$\text{s.t.}\begin{cases}\sum_j f_{sj}-\sum_j f_{js}=v(f)\\ \sum_j f_{tj}-\sum_j f_{jt}=-v(f)\\ \sum_j f_{ij}-\sum_j f_{ji}=0,\forall v_i\in V\backslash\{v_s,v_t\}\\ \underline{c_{ij}}\leqslant f_{ij}\leqslant\overline{c_{ij}},\forall(v_i,v_j)\in A\end{cases}\tag{2}$$

3 模型求解的算法

3.1 可行流的求解方法

首先要确定模型(2)是否存在可行流,如果所有的$\underline{c_{ij}}$都为0,那么零流就是一个可行流。但是当$\underline{c_{ij}}$不全为零时,零流不一定是可行流了,可行流就不容易寻找了。将模型(2)与传统的最大流模型进行比较,它们的区别在于流量限制条件:

模型(2)流量限制条件为:

$\underline{c_{ij}}\leqslant f_{ij}\leqslant\overline{c_{ij}},\forall(v_i,v_j)\in A$

传统的最大流模型(1)流量限制条件为:

$0\leqslant f_{ij}\leqslant c_{ij},\forall(v_i,v_j)\in A$

在模型(2)中进行一个流量变换:令$h_{ij}=f_{ij}-\underline{c_{ij}}$,则$0\leqslant h_{ij}\leqslant\overline{c_{ij}}-\underline{c_{ij}},\forall(v_i,v_j)\in A$,变换后的流$h=\{h_{ij}\}$的下界变为0,于是模型(2)变换为:

$$\text{Max } v(h+\underline{C})$$

$$\text{s.t.}\begin{cases}\sum_j(h_{sj}+\underline{c_{sj}})-\sum_j(h_{js}+\underline{c_{js}})=v(h+\underline{c})\\ \sum_j(h_{tj}+\underline{c_{tj}})-\sum_j(h_{jt}+\underline{c_{jt}})=-v(h+\underline{c})\\ \sum_j(h_{ij}+\underline{c_{ij}})-\sum_j(h_{ji}+\underline{c_{ji}})=0,\forall v_i\in V\backslash\{v_s,v_t\}\\ 0\leqslant h_{ij}\leqslant\overline{c_{ij}}-\underline{c_{ij}},\forall(v_i,v_j)\in A\end{cases}\tag{3}$$

记$\Delta_i=\sum_j\underline{c_{ij}}-\sum_j\underline{c_{ji}},v_i\in V$(即流出$v_i$点的下界总和减去流入$v_i$点的下界总和),于是模型(3)可以写成:

$$\text{Max } v(h+\underline{C})$$

$$\text{s.t.}\begin{cases}\sum_j h_{sj}-\sum_j h_{js}+\Delta_s=v(h+\underline{c})\\ \sum_j h_{tj}-\sum_j h_{jt}+\Delta_t=-v(h+\underline{c})\\ \sum_j h_{ij}-\sum_j h_{ji}+\Delta_i=0,\forall v_i\in V\backslash\{s,t\}\\ 0\leqslant h_{ij}\leqslant\overline{c_{ij}}-c_{ij},\forall(v_i,v_j)\in A\end{cases}\tag{4}$$

模型(4)与传统的最大流模型是非常接近的,要找到模型(4)的可行流$h=\{h_{ij}\}$关键是要保证中间点满足流量平衡条件。对一个给定带上下界限制的有向网络D,每个中间点$v_i\in V\backslash\{v_s,v_t\}$,$\Delta_i$就是一个常数。根据$\Delta_i$的取值情况不同进行如下不同的处理:

当$\Delta_i<0$,$\sum_j h_{ij}=\sum_j h_{ji}-\Delta_i,\forall v_i\in V\backslash\{v_s,v_t\}$,就添加一条弧$(s,v_i)$,容量定义为$-\Delta_i$;当$\Delta_i>0$,$\sum_j h_{ij}+\Delta_i=\sum_j h_{ji}$,$\forall v_i\in V\backslash\{v_s,v_t\}$,可以添加一条弧$(v_i,t)$,容量为$\Delta_i$;当$\Delta_i=0$,其中$v_i\in V\backslash\{v_s,v_t\}$就不做任何变化。这样就能保证中间点的流量平衡。例如图1给出了$\Delta_i>0,\Delta_j<0$增加弧的示意情况。

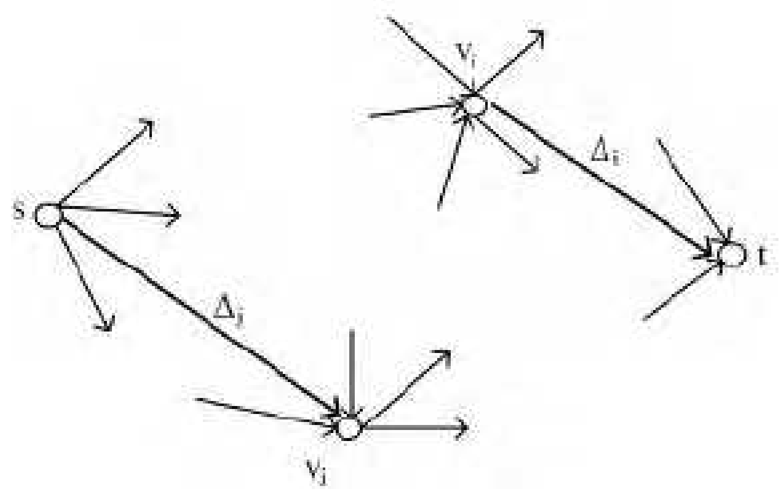

图1 增加弧的示意图

3.2 模型求解的算法

对任意给定的容量带上下限的网络$D=(V,A,\underline{C},\overline{C})$,求最大流的具体算法如下:

(1)对D先进行容量处理:把每条弧的容量下界变为0,上界为$\overline{c_{ij}}-\underline{c_{ij}}$;并且计算中间点的下界容量差$\Delta_i$,其顶点是原容量网络的顶点,处理后的容量网络记为$D_1$。

(2)对D_1进行添加弧的处理:对中间点中v_i,当$\Delta_i<0$则添加一条弧(s,v_i),并定义容量为$-\Delta_i$;当$\Delta_i>0$,则可以添加一条弧(v_i,t),并且定义容量为Δ_i;当$\Delta_i=0$,则什么也不做。

(3)把D_1重新定义为一个容量——费用网络,记为D_2;其中添加上去的弧的费用为-1,其余弧的费用为0,顶点是原容量网络的顶点。

(4)在D_2用文献[2]中最小费用最大流的求解方法去求解从s到t的最小费用最大流。把D_2中所求得最小费用最大流的各弧(除去添加的弧)的流值加上原网络D的容量下界就得到带上下限容量网络模型(2)的最大流。

4 应用实例

求如图2所示的一个容量带上下限的网络图D中s到t的最大流。

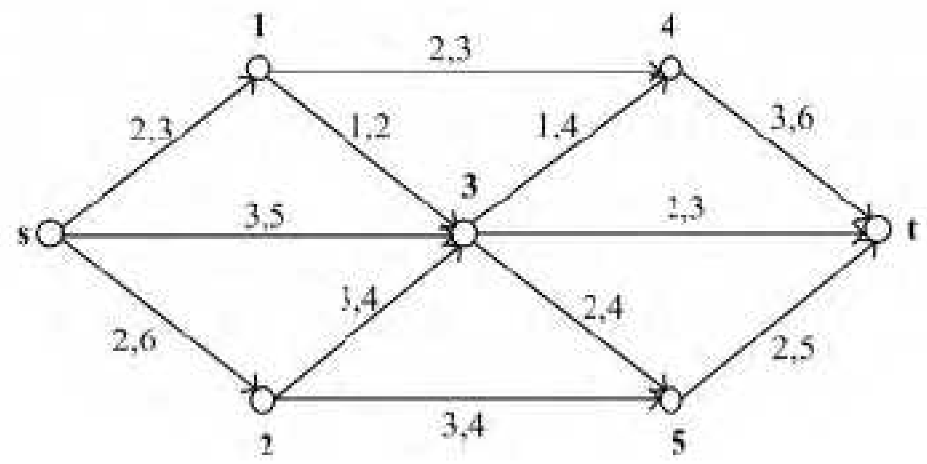

图2 带上下限容量的网络图D

顶点集$V=\{s,t,1,2,3,4,5\}$,每条弧上标的数值是容量的下界和上界$(\underline{c_{ij}},\overline{c_{ij}})$。

(1)对D进行处理,令每条弧容量下界为零(省略不写),上界为$\overline{c_{ij}}-\underline{c_{ij}}$,得到图$D_1$。

(2)计算带上下限容量的网络图D(见图2)中各中间点容量下界的差Δ_i:

$\Delta_1=1$,增加弧$(1,t)$容量为1;

$\Delta_2=4$,增加弧$(2,t)$容量为4;

$\Delta_3=-2$,增加弧$(s,2)$容量为2;

$\Delta_4=0$,不增加弧;

$\Delta_5=-3$,增加弧$(s,5)$容量为3;

在图D_1(见图3)中增加以上几弧,定义费用为-1,其余弧的费用定义为0,得一容量费用网络D_2。

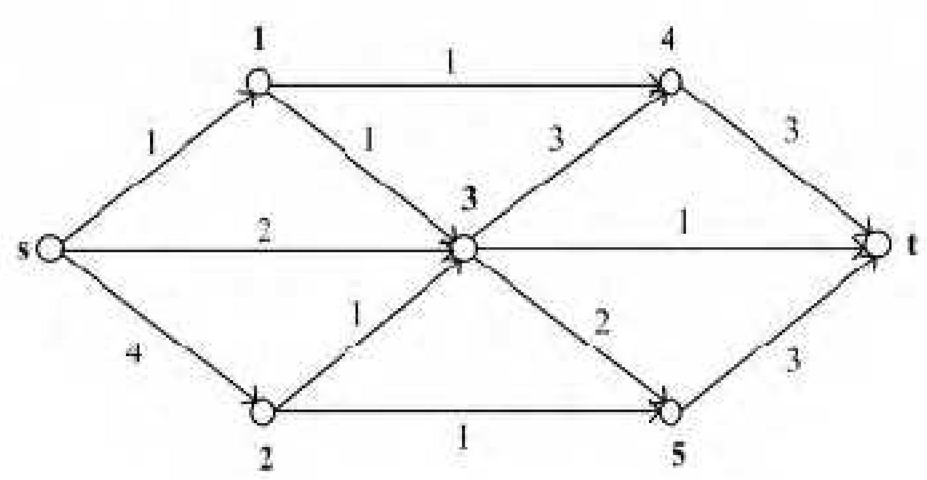

图3 下容量为零的网络图D_1

(3)求解容量费用网络图D_2(见图4)中的最小费用最大流。

(下转第377页)

hoc and Sensor Networks. ACM,2005:21-31

[15] Lin X,Sun X,Ho P H,et al. GSIS:a secure and privacy-preserving protocol for vehicular communications[J]. IEEE Transactions on vehicular technology,2007,56(6 Part1):3442-3456

[16] Boneh D,Boyen X,Shacham H. Short group signatures[C]// Advances in Cryptology-CRYPTO 2004. Springer, 2004: 227-242

[17] Shamir A. Identity-based cryptosystems and signature schemes [C]// Lecture Notes in Computer Science,Advances in cryptology-CRYPTO'84. Springer,1984,196:47-53

[18] Boneh D,Shacham H. Group signatures with verifier-local revocation[C]// 11th ACM Conference on Computer and Communications Security. ACM,2004:168-177

[19] Boneh D, Franklin M. Identity-based encryption from the Weil pairing[C]// Advances in Cryptology — CRYPTO 2001. Springer,2001:213-229

[20] Lu R,Lin X,Zhu H,et al. ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications[C]// INFOCOM'2008. IEEE,2008:1229-1237

[21] Wasef A,Jiang Y,Shen X. DCS: An Efficient Distributed-Certificate-Service Scheme for Vehicular Networks[J]. IEEE Transactions on Vehicular Technology,2010,59(2):533-549

[22] Yang Tao,Xiong Hu,Hu Jian-bin,et al. A traceable privacy-preserving authentication protocol for VANETs based on proxy re-signature[C]// Eighth International Conference on Fuzzy Systems and Knowledge Discovery(FSKD 2011). IEEE,2011,4:2217-2221

[23] Calandriello G, Papadimitratos P, Hubaux J P, et al. Efficient and robust pseudonymous authentication in VANET[C]// Proceedings of the 4th ACM International Workshop on Vehicular Ad hoc Networks(VANET'07). ACM,2007:19-28

[24] 赵宝康. 无线传感器网络隐私保护关键技术研究[D]. 长沙:国防科学技术大学,2009

[25] Scott M. Efficient implementation of crytographic pairings [OL]. http://ecryptss07.rhul.ac.uk/Slides/Thursday/mscottsamos07.pdf,2014

[26] Wasef A,Shen X. Efficient Group Signature Scheme Supporting Batch Verification For Securing Vehicular Networks[C]// IEEE International Conference on Communications(ICC 2010). IEEE,2010:1-5

(上接第 350 页)

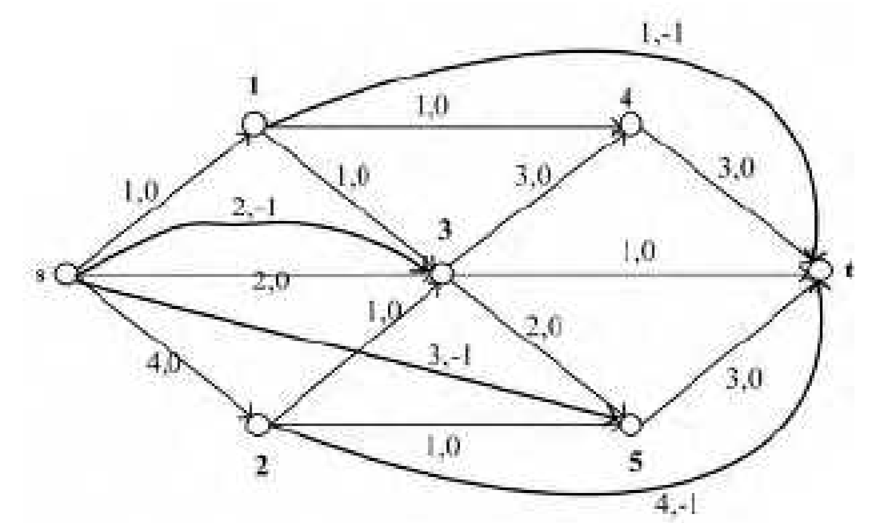

图 4　容量费用网络图 D_2

在容量费用网络图 D_2 中,用文献[2]中所讲述求最小费用最大流方法,取零流为初始可行流,求得最小费用最大流结果如图 5 所示。

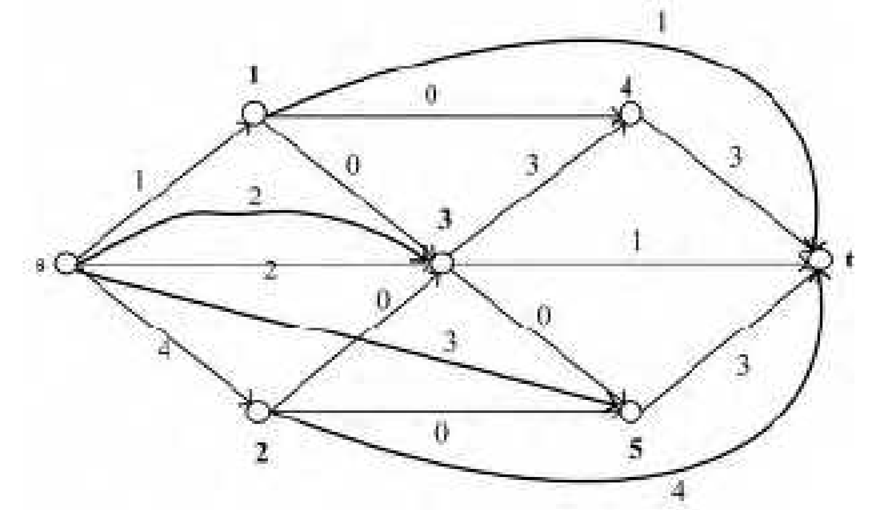

图 5　D_2 最小费用最大流图

(4)在图 5 中,去掉增加的弧((1,t),(2,t),(s,2),(s,5)),其余弧上的流量加上网络 D 中对应弧的容量下限 c_{ij}(如 $f_{s3}=2+c_{s3}=2+3=5$),得容量带上下限的网络图 D 的最大流为 14,如图 6 所示。

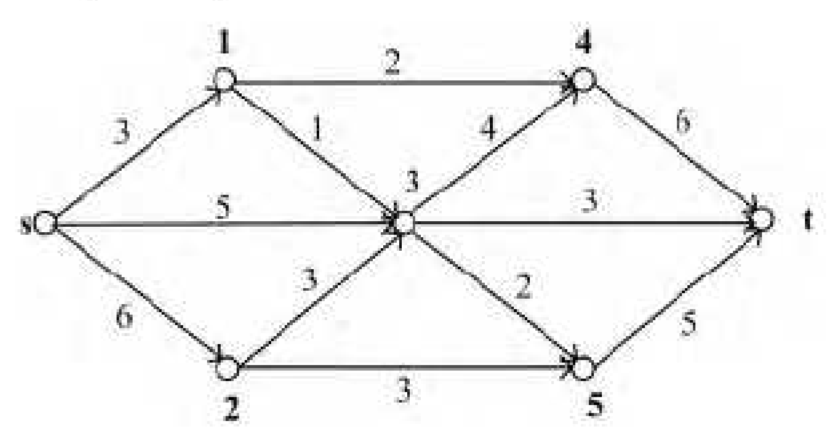

图 6　容量带上下限的网络 D 的最大流图

结束语　通过实例详细说明了将带上下限的网络最大流问题转化为求解网络的最小费用最大流问题算法的方法和步骤,本文所讲的算法更适合顶点集的个数不是很多的有限有向网络图。

参 考 文 献

[1] [美]Johnsonbaugh R. 离散数学[M]. 黄林鹏,译. 北京:电子工业出版社,2009:497-508

[2] 钱颂迪,等. 运筹学[M]. 北京:清华大学出版社,2012:312-320

[3] 胡运权. 运筹学教程权[M]. 北京:清华大学出版社,2007:233-268

[4] 赵礼峰,白睿,等. 求解网络最大流的标号算法[J]. 计算机技术与发展,2001,21(12):113-115

[5] 谢凡荣,贾仁安. 有上下界网络最大流与最小截问题[J]. 运筹管理,2008,17(2):24-31

[6] 张宏斌,等. 运筹学方法及应用[M]. 北京:清华大学出版社,2008:123-133

[7] 王志强,孙小军. 网络最大流的新算法[J]. 计算机工程与设计,2009,30(10):2357-2359

[8] 赵礼峰,白睿. 求解网络最大流问题的标号算法[J]. 计算机技术与发展,2011,21(12):113-115

[9] 库向阳. 点和边有容量约束的网络最小费用最大流算法[J]. 计算机应用研究,2010,27(8):3112-311

[10] 谢政. 网络算法与复杂性理论[M]. 长沙:国防科技大学出版社,2003:116-123

[11] 徐翠霞. 深度优先搜索最大流问题的简单算法[J]. 潍坊学院学报,2006,6(6):30-32

[12] 陈静,单锐. 容差修正网络最大流问题 2F 算法[J]. 长春工业大学学报,2008,29(6):713-716

[13] 谢利民,朱恩强,等. 计算机网络最小割的问题的注记[J]. 数学的实践与认识,2009,39(7):170-171

[14] 高随祥. 图论与网络流理论[M]. 北京:高等教育出版社,2006:292-331