基于一个网络图最大流算法的改进

赵礼峰,陈 华,宋常城,白 睿
(南京邮电大学 理学院,江苏 南京 210003)

摘 要:现有的求解网络最大流算法,存在由于增广链选取的顺序不当而无法得到理想的最大流,且在计算过程中每步都需要画一个网络图等问题。针对上述问题展开讨论,并对一些最大流算法进行改进。利用分层网络及容差的概念,在选择增广链的时候优先选择路径最短且容差较大的路径,并将已饱和的弧画上终止符。最后通过具体的算例验证了改进算法可以简单快速地找到增广链,且避免了标号过程,只需要在一个图上即可完成。整个运算过程,直观性强,计算方便。改进的算法较其他的算法具有高效性和实用性的优势。

关键词:最大流;增广链;Ford - Fulkerson算法;增广链算法;容差;消链
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673 - 629X(2010)12 - 0162 - 04

Improvement of an Algorithm Solving Maximum Flow Based on a Network Diagram

ZHAO Li-feng, CHEN Hua, SONG Chang-cheng, BAI Rui
(College of Mathematics and Physics, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

Abstract: Because of improper selection order of augmented path, the existing algorithm for solving the maximum networks flow, can not obtain the ideal maximum flow, and each step in the calculation of the process needs to draw a network diagram. Discuss these questions, and do some improvement of the existing algorithms. The improved algorithm makes use of layered network and the concept of tolerance, gives priority to the shortest path and greater tolerance when choosing augmenting path, and draws terminators on the arcs have reached saturation. Finally, practical examples demonstrate that the algorithm can find the augmented path easily and quickly, and avoid the labeling process, the entire operation process only needs drawing a diagram to be completed. It is strong intuitive and convenient to calculate. The improved algorithm has the advance of efficiency and practicality comparing with the existing algorithms.

Key words: maximum flow; augmenting path; Ford - Fulkerson algorithm; augmenting path algorithm; tolerance; eliminating chain

0 引 言

网络最大流问题是指在一定的条件下,要求通过网络的物流、能量流、信息流等流量为最大的问题。它在工程、计算机原理与通信系统、应用数学以及社会和军事等领域有着广泛的应用。

最大流问题已有五十多年的研究历史,在这五十多年中,人们已经建立了关于最大流问题较为完善的理论,并设计了一系列求解最大流的算法[1~9],还有一些关于网络最大流算法的实现。最大流问题已有很多算法,其中主流算法可以分为两大类:一类是沿路径推进的增广链算法,其中典型的算法有 Ford - Fulkerson 算法[10]以及 Dinic(1970), Edmonds - Karp(1972)独立地提出的最短增广链算法[11];另一类是沿边推进并把多余部分返回的预流推进算法,其中典型的算法有 Karzanov 的分层网络阻塞流算法,以及 Goldberg 和 Tarjan(1986)在 Karzanov 的基础上改进的推进 - 重标号算法等[10]。

但是这些主流算法也存在各自的优缺点, Ford - Fulkerson 算法思路清晰,分为标号过程和增广过程,通过每次在所有的增广链中随机地找一条增广链进行增广,但是正是因为每次选取增广链的任意性,会使计算量变大,为此必须要修正增广链的选取方法,排除任意性,因此 Dinic(1970), Edmonds - Karp(1972)分别提出了最短增广链算法,其主要思想是每次都沿着最短(即弧数最少的)增广链进行增广,这样排除了增广链选取的任意性,但是在计算过程中要先计算剩余网络才能进行增广,步骤较为繁琐,增广链算法每次都只能增广一条路径,但预流推进算法在路径选择时基于网络图的宽度优先搜索策略,关注于每条弧的操作和处

收稿日期:2010 - 04 - 07;**修回日期**:2010 - 07 - 11
基金项目:南京邮电大学科研基金项目(NY207149)
作者简介:赵礼峰(1959 -),男,安徽淮北人,教授,硕士研究生导师,研究方向为图论及其应用。

理,而不必一次只处理一条增广路,但是当网络规模较大时,它的直观性就大大下降,实现难度就明显提高。

此外,近年来随着计算机科学技术和网络的快速发展,关于求解网络最大流问题得到了更深的研究,许多学者在主流算法的基础上纷纷提出了许多改进的算法,譬如"消链"算法[3]是基于 Ford – Fulkerson 算法在求解网络最大流的时候需要经过多次的标号与调整的基础上受水流的启发,通过引入极大一致链的概念提出的一种求解最大流的算法,它主要通过寻找容量网络中的极大一致链,并根据所得到的极大一致链对网络逐步地进行调整,避免了标号算法的标号过程。另外还有因 Ford – Fulkerson 算法对于增广链的选取过于随意,造成算法的不稳定,效率较低,基于这些问题受阻塞网络中容差概念的启发,在搜索增广链时加入了对顶点容差的判定,优先选取顶点容差为正的顶点加入增广链中,从而提出了"容差修正网络最大流2F"算法[6]。而"构造式"算法[9]是在前向推进最大流算法的基础上,应用深度优先搜索原理,摒弃前向推进最大流算法的并行控制而着眼于每一支流的增广,逐步构造中间过程"构造图"的结构并最终得到网络最大流,除此之外还有许多简单可行的算法,另外对于容量限制网络国内外也有一些研究[12],在这里不再一一陈述。

文中主要是针对增广链算法进行讨论,它的主要思想是从一个可行流出发,寻找增广链,而后调整增广链,反复这个过程直到找不到增广链为止。对于不同的算法或者算法的改进它们之间的区别大多在于增广链的选取,如果选对增广链那么整个计算过程简单准确,相反地,如果选不对增广链那么有可能造成计算繁琐甚至得不到正确的结果。那么可不可以找到一种计算起来既简单又有效的算法呢? 下面就通过一个具体的网络图来分析解决这个问题。

1　问题的分析

1.1　基本概念

容差:对于有向赋权网络 $N = (s,t,V,A,C)$,所谓顶点 A 的容差 ϕ_A 是指:所有以 A 为始点的有向弧的容量总和与所有以 A 为终点的有向弧的容量和之差,即:

$$\phi_A = \sum\{[C(a) \mid v_i(a) = A]\} - \sum\{[C(a) \mid v_j(a) = A]\} = C_A^+ - C_A^-$$

式中,$v_i(a)$ 为弧 a 的始点;$v_j(a)$ 为弧 a 的终点。

具有负容差的顶点被称为结构堵塞点。简单地说,容差就是某点所有出弧容量之和与该点所有入弧容量之和的差值。

注:文中没有定义的术语及记号均同文献[10]。

1.2　用不同的增广链算法求解

1.2.1　Ford – Fulkerson 算法

Ford – Fulkerson 算法分为标号过程和增广过程,通过每次在所有的增广链中随机地找一条增广链进行增广。因为对顶点的标号顺序是任意的,所以用该算法求出的可增广链(若存在的话)也是任意的。如果经过一系列标号之后先选取增广链:$S - v_1 - v_6 - T$,那么再经过一系列的标号增流得到此网络图的最大流为16;但是如果先选择增广链:$S - v_1 - v_4 - T$ 时,再经过一系列的标号增流过程得到的最大流为 18。由此可知,正是因为这种任意性,不但会使计算量变大,而且有可能会得不到想要的最大流。

1.2.2　Edmonds – Karp 最短增广链算法

Edmonds – Karp 最短增广链算法对 Ford – Fulkerson 算法的修改实质上是对顶点的标号过程中采用了"宽度优先"策略,即"先标号先扫描"也就是对已标号的顶点扫描时,应先检查标号最早但未考察过的点,并对与该点邻接的所有未标号的顶点给予标号。因而可求得可增链是总长度最短的可增广链。它的主要思想是每次都沿着最短(即弧数最少的)增广链进行增广。

但是用该算法算图 1 的时候还是会存在问题。在标号过程中标号顺序是 $S, v_1, v_2, v_3, v_6, v_4, v_5, T$,那么根据先标号先扫描的原则首先得到增广路为 $S - v_1 - v_6 - T$,计算到最后可以得到最大流为16,因为与 v_1 相邻的有 v_4 和 v_6,这两个顶点的地位是一样的,因此当标号顺序是 $S, v_1, v_2, v_3, v_4, v_6, v_5, T$ 时,首先得到的增广链是 $S - v_1 - v_4 - T$,同样的计算到最后可以得到最大流为 18。

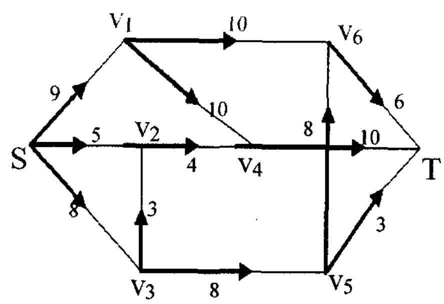

图 1　原网络图

1.2.3　改进的一些算法

"消链"算法是基于 Ford – Fulkerson 算法在求解网络最大流的时候需要经过多次的标号与调整的基础上受水流的启发,通过引入极大一致链的概念提出的一种求解最大流的算法,它主要通过寻找容量网络中的极大一致链,并根据所得到的极大一致链对网络逐步地进行调整,避免了标号过程,大大降低了算法的复

杂度，加快了整个算法的执行速度，但是对于图 1 该算法仍然不适用。同样增广链：$S-v_1-v_6-T$ 与 $S-v_1-v_4-T$ 的长度相等，如果选择不慎，也将得不到结果。

容差修正网络最大流 2F 算法是基于 Ford－Fulkerson 算法对增广链的选取的任意性在搜索增广链时加入了对顶点容差的判定，即在多个未标号的未饱和弧中选取增广链进行下一步推进时，不再一视同仁，而是优先选取顶点容差为正的顶点。该算法增大了每条增广链的可增量，减少了选取的增广链的数量，提高了算法的搜索效率。但是对于容差为同正或者同负的两个顶点，例如图 1 中的顶点 v_4 与 v_6 它们的容差均为负，在选择 $S-v_1-v_6-T$ 与 $S-v_1-v_4-T$ 中的哪条可增广链时也存在问题。

此外，还有许多改进的算法，但是如果没有认识到长度相同的两条增广链如图 1 中的 $S-v_1-v_6-T$ 与 $S-v_1-v_4-T$ 有何区别，不知道先选择哪条增广链就有可能得不出正确的结果。

2 新算法的思想及步骤

2.1 算法的思想

对于所有求解网络最大流的算法，它们的主要区别在于增广链的选取上，文中在现有算法的基础上对增广链的选取过程进行了一些改进，另外省去了标号过程，所有的步骤都可在一个网络图上完成。

首先，由 1.2 人们可以看出如果对增广链选取可能会得不到正确的结果。而 1.2 中的问题主要是由顶点 v_1 之后该先经过 v_4 还是 v_6，按照结构堵塞点的定义 v_4 和 v_6 均为结构堵塞点，(对于当可行流同时遇到结构堵塞点和非结构堵塞点时是如何推进的请见参考文献[6]) 这里主要讨论当可行流同时遇到两个负容差的顶点即结构堵塞点时该如何选择，v_4 和 v_6 唯一的区别在于它们的容差不同，其中 v_4 的容差为 -4，v_6 的容差为 -12。虽然它们均为结构堵塞点，但是试想如果网络流先推进容差为 -12 的阻塞点 v_6 时，可能会由于出现出弧容量小于弧流量而需要减小前面增广流量的情况，甚至可能由于所有出弧都已饱和而终止增广过程。相反的，如果先流经相对结构堵塞点 v_6 容差大一点的 v_4 来说，只要 v_4 的入弧没有饱和，就完全没有网络流不流入 v_4 的出弧的可能。

其次，为了寻找最短增广链，把容量网络 $A(f)$ 中的所有顶点进行分层，当 $A(f)$ 中存在 (v_s, v_i) 路($\forall v_i \in V$) 时，可以求出从发点 v_s 到其余各顶点 v_i 的最短路的长 $h(v_i)$。$h(v_i)$ 就是 v_i(关于 v_s) 的层数，即 v_i 为 $A(f)$ 的第 $h(v_i)$ 层顶点。$A(f)$ 的第 0 层只有一个顶点 v_s，把顶点分层后，$A(f)$ 的弧又可以分为 3 类：

第 1 类为从第 i 层顶点到第 $i+1$ 层顶点的弧；

第 2 类为从第 i 层顶点到同一层顶点的弧；

第 3 类为从第 i 层顶点到第 j 层顶点的弧($j < i$)。

在这里为了寻找最短增广链对于容量网络中的弧优先考虑第 1 类弧，其次是第 2 类弧，当然这个规定也不是绝对的，可以根据实际情况进行适当的变通。

最后，在求解最大流的过程中，可以将可增容量直接标记到对应的弧上，当一条弧达到饱和时，就在所在的弧上画一个终止符“||”表示该弧将终止增流。这样就可以在一个网络图上完成整个运算过程。

算法的主要思想就是先将网络图进行分层，然后在各个顶点上标上对应的容差，按照容差的大小，优先选择路径最短且容差较大的可行流进行增广，并且将已经达到饱和的弧划去，直至不再存在可增广链为止。

2.2 算法步骤

求容量网络 $G=(V,E,c,f)$ 中的最大流 f：

步骤 1：初始化有向网络，置初始可行流为 0。

步骤 2：将容量网络 $G=(V,E,c,f)$ 中所有的弧进行分层，并在每个顶点上方或者下方标上容差，其中发点 v_s 标“$+\infty$”，收点 v_t 标“$-\infty$”。

步骤 3：从源点 v_s 出发，判断是否存在一条到汇点 v_t 的可增广链，若存在转步骤 4，若不存在转步骤 5。

步骤 4：寻找一条从 v_s 到 v_t 的最短可增广链(即经过中间点最少的路径)，在选取相同层次的顶点时优先选取容差相对大的顶点，并将此增广链中对应弧的流量后面相应地标上“$+\delta$”或“$-\delta$”，其中：

$$\delta = \min\{\min_{(v_i,v_j)\in u^+}(c_{ij}-f_{ij}),\ \min_{(v_i,v_j)\in u^-} f_{ij}\}$$

确保每次的增流至少使一条弧饱和，并在饱和弧上画上终止符“||”，转步骤 3。

步骤 5：当不存在 (v_s, v_t) 路时算法终止，求得网络最大流。

3 算例与比较

例题：求图 1 的网络最大流。

解：

(1) 如图 2 所示，将容量网络中的每个顶点均标上容差的大小；

(2) 根据在选择可增广链时优先选择路径最短且容差大的顶点，找到可增广链 u_1：$S-v_1-v_4-T$，其中 $\delta_1 = 9$。由此得出弧 (S, v_1) 饱和，因此在 (S, v_1) 上画上终止符“||，”如图 3 所示；

(3) 同理可以得 u_2：$S-v_3-v_5-T$，其中 $\delta_2 = 3$，且弧 (v_5, T) 为饱和弧，画上终止符“||”，如图 4 所示；

(4) 同理可以得 $u_3: S - v_2 - v_4 - T$,其中 $\delta_3 = 1$,且弧(v_4, T) 为饱和弧,画上终止符"||",如图 5 所示;

(5) 最后对可增广链 $u_4: S - v_3 - v_5 - v_6 - T$ 进行增广,其中 $\delta_4 = 5$。这样弧(S, v_3) 与(v_3, v_5) 成为饱和弧,从而画上终止符"||",如图 6 所示。

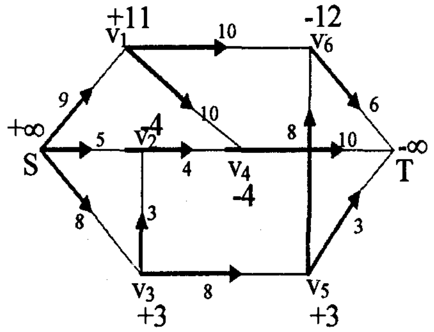

图 2　在每个顶点标上容差

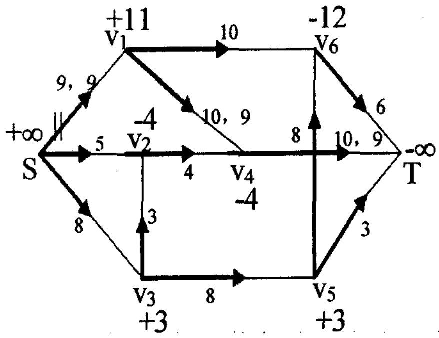

图 3　选取增广链 u_1 并进行增广

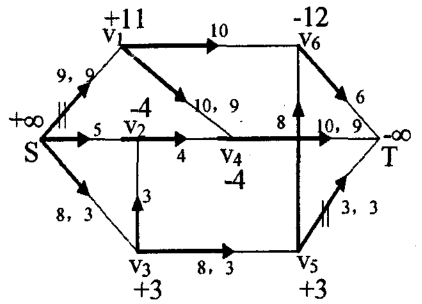

图 4　选取增广链 u_2 并进行增广

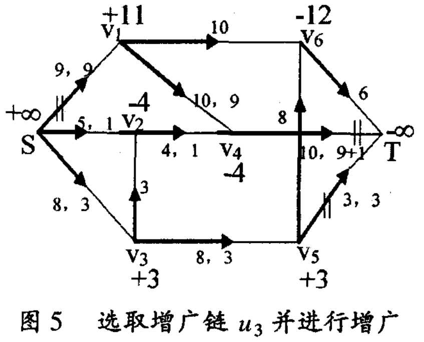

图 5　选取增广链 u_3 并进行增广

可以看到,图 6 中不再含有从始点 S 到终点 T 的可增广链,因此算法结束,求得网络最大流 $f_{max} = 9 + 3 + 1 + 5 = 18$。注:用此算法求解最大流可以直接在一个图上即可完成整个增流过程,如本题中只用图 6 就可标上整个运算过程。

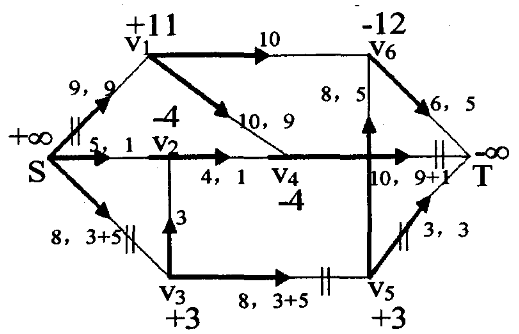

图 6　选取增广链 u_4 并进行增广

下面简单地说一下用 Ford - Fulkerson 算法求解本例题(图略)。

(1) 首先,我们对图 1 的顶点进行标号,找到可增广链 $u_1: S - v_1 - v_6 - T$,其中 $\delta_1 = \min\{\infty, 9, 9, 6\} = 6$。对 u_1 进行增广;

(2) 其次,对增广后的图进行重新标号后可以得到可增广链 $u_2: S - v_1 - v_4 - T$,其中 $\delta_2 = 3$,再对 u_2 进行增广;

(3) 同理,依次得到可增广链 $u_3: S - v_2 - v_4 - T$,$u_4: S - v_3 - v_5 - T$,其中 $\delta_3 = 4, \delta_4 = 3$。

经过增广链 $u_4: S - v_3 - v_5 - T$ 增广之后,可以看出不再存在(S, T) 可增广路,算法结束。从而得到最大流 $f_{max} = 6 + 3 + 4 + 3 = 16$。

由此可以看出用 Ford - Fulkerson 算法不仅步骤多,而且会由于可增广链的选取不慎得不到最大流。用文中所列举的其它的增广路算法也是如此,另外用预流推进算法也存在同样的问题,在这里不再赘述。

4　结束语

文中是由一个常见的网络图所引出的问题进行讨论,从而在现有算法的基础上给出的算法。利用分层网络及容差的概念,在选择增广链的时候优先选择路径最短且容差较大的路径,避免了多次对网络的标号过程,而且将已饱和的弧画上终止符,这样避免了重复计算,把复杂的网络简单化,加快了整个算法的执行速度,既节省计算时间,又不易漏掉增广链,直观性强,计算方便,而且在整个算法过程中只需要画一个图就可以找出最大流,最后通过具体的算例比较验证了算法的效率和实用性。

参考文献:

[1] Zhang Xian chao. Research on the maximum network flow problem[J]. Journal of Computer Research and Development,

(下转第 176 页)

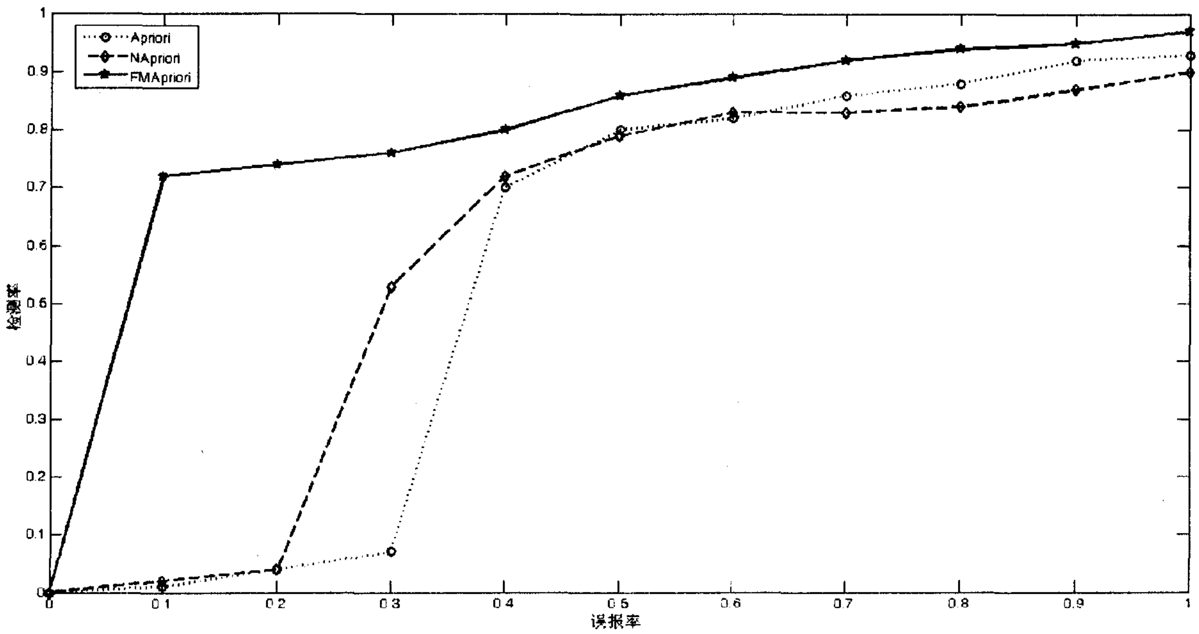

图 2 检测性能对照图

参考文献：

[1] 毛国君，段立娟，王 实，等. 数据挖掘原理与算法[M]. 北京:清华大学出版社，2005.

[2] 韩家炜，坎 伯. 数据挖掘概念与技术[M]. 第 2 版. 范明，孟小峰译. 北京:机械工业出版社，2007.

[3] Agrawal R，Imieilnski T，Swami A. Mining association rules between sets of items in large databases[C]// Proceedings of the ACM SICMOD International Conference on Management of Data. Washington DC:[s. n.]，1993:207－216.

[4] Ding Ya－Li，Li Lei，Luo Hong－Qi. A novel signature searching for intrusion detection system using data mining[C] // Machine Learning and Cybernetics，2009 International Conference. [s. l.]:[s. n.]，2008:122－126.

[5] 杨义先，钮心忻. 入侵检测理论与技术[M]. 北京:高等教育出版社，2006.

[6] 李 雷，丁亚丽，罗红旗. 基于规则约束制导的入侵检测研究[J]. 计算机技术与发展，2008，18(3):143－146.

[7] 罗军生，李永忠. 基于模糊 C－均值聚类算法的入侵检测[J]. 计算机技术与发展，2008，18(1):178－180.

[8] Yun Unil. An efficient mining of weighted frequent patterns with length decreasing support constraints[J]. Knowledge－Based Systems archive，2008，21(8):741－752.

[9] Seno M，Karypis G. Finding Frequent Patterns Using Length－Decreasing Support Constraints [J]. Data Mining and Knowledge Discovery，2005，10(3):197－228.

[10] 孙晓霞，刘晓霞. 模糊 C 均值聚类算法的实现[J]. 计算机应用与软件，2008，25(13):48－50.

[11] 李 雷，罗红旗，丁亚丽. 一种改进的模糊 C 均值聚类算法[J]. 计算机技术与发展，2009，19(12):71－73.

[12] KDD (1999)，the third international knowledge discovery and data mining tools competition data set (KDD99 Cup)[EB/OL]. 1999. http://kdd. ics. uci. edu/databases/kddcup99. html.

(上接第 165 页)

2003，40(9):1281－1292.

[2] Martens M，Skutella M. Flows on few paths: Algorithms and lower bounds[J]. Networks，2006，48(2):68－76.

[3] 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.

[4] 罗会兰. 网络最大流算法的改进[J]. 邵阳高专学报，1996，9(3):201－203.

[5] 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.

[6] 陈 静，单 锐. 容差修正网络最大流 2F 算法[J]. 长春工业大学学报，2008，29(6):713－716.

[7] 张宪超，江 贺. 一个新的最大流问题增载轨算法[J]. 小型微型计算机系统，2006，27(9):1726－1730.

[8] 孙泽宇. 基于标号法求解网络最大流算法的研究[J]. 甘肃联合大学学报(自然科学版)，2009，23(4):64－66.

[9] 郑宣耀，张 帆. 求解最大流问题的"构造式"算法[J]. 深圳职业技术学院学报，2005，4(1):18－20.

[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] 张先迪，李正良. 图论及其应用[M]. 北京:高等教育出版社，2005:244－253.

[12] Martens M，Skutella M. Flows with unit path capacities and related packing and covering problems[J]. J. Comb Optim，2009，18:272－293.