第 29 卷　第 5 期
2019 年 5 月

计 算 机 技 术 与 发 展
COMPUTER TECHNOLOGY AND DEVELOPMENT

Vol. 29　No. 5
May　2019

最大流问题的最短增广链改进算法

邵丽萍,赵礼峰
(南京邮电大学 理学院,江苏 南京 210023)

摘　要:BA 无标度网络是现实中常见的网络,在该网络中,任意两节点之间有极大可能存在多条路径,若用 Ford-Fulkerson 算法寻找增广链,效率不高且步骤繁杂。同时,在当今大数据时代背景下,随着网络规模的增加,提高算法效率成为解决大规模网络最大流问题的关键。为了改善以上不足,文中在最短增广链算法的基础上作了一些改进,提出了最短增广链改进算法。该算法基于最短增广链算法,删除原网络中没有起作用的弧;在分层剩余网络中删除的饱和弧,相应的在原网络中删除该弧,降低构建剩余网络和分层剩余网络的复杂性,从而优化最短增广链算法。实验结果表明,在 BA 无标度网络中该算法与最短增广链算法的计算结果相同,并且运行效率比最短增广链算法有所提高。

关键词:最大流;最短增广链;剩余网络;分层剩余网络;BA 无标度网络
中图分类号:TP301.6　　**文献标识码**:A　　**文章编号**:1673-629X(2019)05-0058-04
doi:10.3969/j.issn.1673-629X.2019.05.012

Shortest Augmented Chain Improvement Algorithm for Maximum Flow Problem

SHAO Li-ping,ZHAO Li-feng
(School of Science,Nanjing University of Posts and Telecommunications,Nanjing 210023,China)

Abstract:BA (Barabasi-Albert) scale-free network is a common network in reality. In this network,there exists multiple paths between any two nodes in high possibility. If the Ford-Fulkerson algorithm is used to find the augmented chain,the efficiency is not high and the steps are complicated. At the same time,in the context of the big data,with the increase of network scale,improving algorithm efficiency becomes the key to solve the problem of maximum flow in large-scale networks. In order to improve the above shortcomings,based on the shortest augmented chain algorithm,we make some improvements and propose an improved shortest augmented chain algorithm. The arcs that have no effect in the original network are deleted;the saturated arcs that are deleted in the original network are correspondingly deleted from the layered residual network,which reduces the complexity of constructing the residual remaining network and the layered residual network and optimizes the shortest augmented chain algorithm. The experiment shows that the results of the new algorithm is the same as that of the shortest augmented chain algorithm in BA scale-free network,and the operation efficiency is higher than that of the shortest augmented chain algorithm.

Key words:maximum flow;shortest augmented chain;residual network;layered residual network;BA scale-free network

0　引　言

最大流问题是图论领域中的重要问题之一,该问题有很直观的现实背景,如在交通运输网络中的货物运输、车辆限行,生产制造中的制造工具,通信网络中包的路由选择,电网系统的电流等问题[1-4]都可用最大流模型来描述。因此对最大流算法研究有重要意义。

到目前为止,对最大流问题的研究已有 50 多年的历史,并且形成了一套完整的理论体系。针对最大流问题主要有两类算法,即增载轨算法和预流推进算法。经典的增载轨算法有 Ford-Fulkerson 标号算法[5]、最短增广链算法[6]、最短增载轨算法[7]等。经典的预流推进算法有 Karzanov 的分层阻塞流算法[8]、Cherkassky 的最高标号预流推进算法[9]等。其中增载轨算法是沿路径进行推进的,预流推进算法是沿边进行推进并把多余部分返回。增载轨算法因计算过程简单而应用广泛。除了以上这些最大流问题的经典算法,许多学者也提出了改进的最大流算法[10-12]。如今,这些算法是研究大规模网络的基础。

收稿日期:2018-07-09　　**修回日期**:2018-11-13　　**网络出版时间**:2018-12-21
基金项目:国家自然科学基金青年基金项目(61304169)
作者简介:邵丽萍(1993-),女,硕士研究生,研究方向为网络最大流、最小费用流;赵礼峰,教授,硕导,研究方向为图论及其在通信中的应用。
网络出版地址:http://kns.cnki.net/kcms/detail/61.1450.TP.20181221.1625.072.html

在增载轨算法中,最短增广链算法应用广泛,它是通过给定的初始可行流在分层剩余网络上沿最短路径进行增广流值,同时不断地更新分层剩余网络和最大流流值,直至终点得不到标号为止。虽然最短增广链算法排除了增广链选取的任意性,但在实际应用中,效率并不是很高。所以,文中提出了一种改进算法,删除原容量网络中没有起作用的弧,并且分层剩余网络删除饱和弧时,删除原容量网络中对应的弧,避免产生回溯现象,简化寻找可增广链的过程,以提高算法效率。

1　基本概念

1.1　最大流模型

定义一个含容量的网络[13],记为 $G=(V,A,c)$,其中 $V=\{v_1,v_2,\cdots,v_n\}$, $A=\{(v_i,v_j)\mid i,j\in V\}$,弧 (v_i,v_j) 称为节点 v_i 的出弧,同时也称为节点 v_j 的入弧,并且称 v_i 为 v_j 的入邻点, v_j 为 v_i 的出邻点,定义流 f_{st} 为网络 $G=(V,A,c)$ 中从始点 s 到终点 t 的流,如果 f_{st} 满足下列条件:

$$\sum_j f(i,j)-\sum_j f(j,i)=\begin{cases}f_{st},i=s\\0,i\neq s,t,0\leqslant f(i,j)\leqslant\\-f_{st},i=t\end{cases}$$
$$c(i,j),\forall(i,j)\in A(G)\tag{1}$$

则 f_{st} 是网络 G 的一个可行流,在所有满足条件的可行流中,流量最大的可行流被称为最大流,记作 $f_{\max}$ 。式 1 中满足 $f(i,j)<c(i,j)$ 的弧为非饱和弧,满足 $f(i,j)=c(i,j)$ 的弧为饱和弧。可行流的定义满足流量守恒,体现在两方面,一方面是始点 s 的所有流量全部到达终点 t ,另一方面是始点 s 和终点 t 以外的所有节点只通过流量而不储存流量。

1.2　广探法(广度优先搜索)

广度优先搜索[14]是生成分层网络、寻找最短增广链与构造距离标号函数的基础,节点采用先进先出的遍历方式,其算法步骤如下:

Step1:在包含始点 v_s 终点 v_t 的容量网络 $G=(V,A,c)$ 中,始点 v_s 记为标号未检查,令 $h(v_s)=0$, $l_s=-1$。

Step2:若所有已标号节点已检查完,转 Step4;若存在未检查的已标号节点,找到最先标号的节点 v_i ,转 Step3。

Step3:考察 v_i 的所有出弧 (v_i,v_j) ,若 v_j 未标号,将 v_j 记为未检查已标号,且令 $h(v_j)=h(v_i)+1$, $l_j=i$;若 v_j 已标号则不做处理;若 v_i 的出弧全部考察完,则节点 v_i 记为已检查,转 Step2。

Step4:算法终止, $h(v_i)$ 为容量网络 $G=(V,A,c)$ 中最短 (v_s,v_t) 路的长度。

1.3　分层剩余网络

剩余网络[5]是指在一张含可行流的容量网络图中所有非饱和弧及所有节点的集合,亦可用反向弧在网络中标记出饱和弧及当前流。记剩余网络为 $U(f)=(V,A(f),c(f))$,需满足两个关系:

(1) $\forall(i,j)\in A(G)$,若 $f(i,j)<C_G(i,j)$,则 $C_U(i,j)=C_G(i,j)-f(i,j)$;

(2) $\forall(i,j)\in A(G)$,若 $f(i,j)>0$,则 $C_U(j,i)=f(i,j)$ 。

由剩余网络的定义及广度优先搜索算法,可以定义剩余网络 $U(f)=(V,A(f),c(f))$ 的一个子网络 $\mathrm{AG}(f)=(V',A'(f),c(f))$,如下:

$$\begin{cases}V'=\{v_t\}\cup\{v_i\in V\mid h(v_i)<h(v_t)\}\\A'=\{(v_i,v_j)\in A\mid h(v_j)=h(v_i)+1<h(v_t)\}\cup\\\qquad\{(v_i,v_j)\in A\mid h(v_i)=h(v_t)-1\}\end{cases}\tag{2}$$

称 $\mathrm{AG}(f)$ 为 G 的关于 f 的分层剩余网络。

1.4　BA 无标度网络

BA 无标度网络[15-16]被用来模拟不断增长的网络节点的无标度网络模型,其创建过程如下:

(1)开始于一个包括 m_0 个节点的网络,每次新增一个节点,都要相应地连接到 $m(m<m_0)$ 个已有节点上。

(2)已有节点 i 与新节点连接的概率为: $p_i=k_i/\sum_j k_j$ (其中 k_i 、k_j 分别表示节点 i 与节点 j 的度)。

2　一种改进的最大流算法

2.1　最短增广链算法步骤

输入:原始容量网络 $G=(V,A,c)$ 和始点 v_s 终点 v_t ,节点数 n ,弧数 m ;

输出:容量网络的最大流 f 。

Step1:在原始容量网络 $G=(V,A,c)$ 中,取零流 f 作为初始可行流,根据 f 构造网络 G 的剩余网络 $U(f)$,并利用广探法对剩余网络 $U(f)$ 进行分层,进而求出分层剩余网络 $\mathrm{AG}(f)$ 。若在 $\mathrm{AG}(f)$ 中终点 v_t 得不到标号,结束算法, f 为容量网络的最大流,否则转 Step2。

Step2:求分层剩余网络 $\mathrm{AG}(f)$ 中 $v_s\to v_t$ 的一条最短路 p ,沿路 p 增加流值,并删除 $\mathrm{AG}(f)$ 中所有的饱和弧。

Step3:求分层剩余网络 $\mathrm{AG}(f)$ 中 $v_s\to v_t$ 的一条最短路 p ,若不存在,则转 Step1;否则转 Step2。

2.2　算法思想

在寻找最短增广链的过程中,经常出现需要遍历没有起作用的弧,使得算法效率降低。针对这种情况,

对最短增广链算法进行两处改进。第一处在 Step1 之前,先删除原容量网络中没有起作用的弧,简化 Step1 中构造剩余网络及分层剩余网络过程。第二处在 Step2 中删除 $AG(f)$ 中所有饱和弧后,同时删去原容量网络中对应的弧,简化 Step1 中重新构造网络的过程。

2.3 算法步骤

Step1:遍历原始容量网络 $G=(V,A,c)$ 中除了始点终点的其余节点,判断其是否有出弧,若没有删除 v_i 的所有入弧。

Step2:取零流 f 作为初始可行流,根据 f 构造网络 G 的剩余网络 $U(f)$,并利用广探法对剩余网络 $U(f)$ 进行分层,进而求出分层剩余网络 $AG(f)$ 。若在 $AG(f)$ 中终点 v_t 得不到标号,结束算法, f 为容量网络的最大流,否则转 Step3。

Step3:求分层剩余网络 $AG(f)$ 中 $v_s \to v_t$ 的一条最短路 p ,沿路 p 增加流值,删除 $AG(f)$ 中所有饱和弧。

Step4:求分层剩余网络 $AG(f)$ 中 $v_s \to v_t$ 的一条最短路 p ,若不存在,则在原网络中删除 Step3 中相应的饱和弧,再转 Step1;否则转 Step3。

2.4 可行性分析

在新算法的操作过程中,删除了没有起作用的弧,对最大流的求解并没有影响。因为在算法求解过程中,这些弧不进入最短增广链。删除 $AG(f)$ 中所有的饱和弧后,同时删去原容量网络中对应的弧,删除的饱和弧在以后的增广过程中,它的流值不可能增加或减少[17],只是避免产生回溯现象,从而简化了重新寻找可增广链的过程。设容量网络 $G=(V,A,c)$ 有 m 条弧,每次增广后,至少删除一条饱和弧,则最多经过 m 次增广后,网络 $G=(V,A,c)$ 中就不存在从始点 v_s 到终点 v_t 的可增广链,此时算法终止,即新算法不会进入死循环中。

2.5 复杂度分析

在容量网络 $G=(V,A,c)$ 中,设节点数为 n ,弧数为 m 。在 Step1 中执行一次搜索的复杂度为 $O(m)$,步数不超过 $O(n)$,在执行 Step2 时,由广探法知,每次构造剩余分层网络的复杂性为 $O(m)$,最多执行 $n-1$ 次。Step3 和 Step4 中最多增广 m 次,每次增广的计算量为 $O(n)$,所以新算法的时间复杂度为:

$$O(mn)+O(mn)+O(mn^2)=O(mn^2) \tag{3}$$

3 数值实例与分析

例:容量网络 G 如图 1 所示,求从始点 v_s 到终点 v_t 的网络最大流。

图中,每条弧旁的数字表示容量,取初始可行流为零流。首先通过 Step1,删除原容量网络中没有起到作用的弧 (v_1,v_3) , (v_4,v_3) , (v_t,v_3) 。通过 Step2–Step3 得到分层剩余网络中最短增广链有 3 条弧,分别为 $P_1=v_sv_1v_4v_t$ 、 $P_2=v_sv_2v_4v_t$,增广流值为 12。在增广流值的过程中弧 (v_s,v_1) 、 (v_2,v_4) 达到饱和,此时 $AG(f)$ 中不存在 $v_s \to v_t$ 的一条路 P ,如图 2 所示,即原容量网络中不存在只有 3 条弧的增广链。

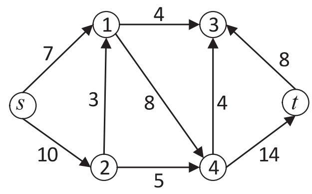

图 1 原容量网络 G

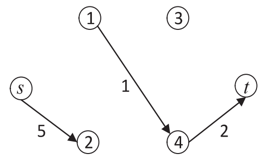

图 2 删除饱和弧 (v_s,v_1)、(v_2,v_4) 后的分层剩余网络图

通过 Step4,在原容量网络中删除弧 (v_s,v_1) 、 (v_2,v_4) ,返回 Step1–Step3 再重新构建分层剩余网络,如图 3 所示。此时有最短增广链 $P=v_sv_2v_1v_4v_t$,含有 4 条弧。

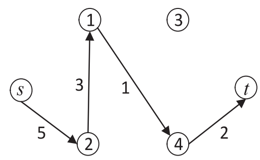

图 3 重新构建的分层剩余网络图

增广流值 1,在增广流值的过程中弧 (v_1,v_4) 达到饱和,此时 $AG(f)$ 中不存在 $v_s \to v_t$ 的一条路 P ,即原容量网络中不存在只有 4 条弧的增广链。返回 Step1–Step2 再重新构建分层剩余网络,此时分层剩余网络中 v_t 得不到标号,于是可得最大流 f ,流值为 13。

4 仿真实验

4.1 实验环境与设计

该算法在 MATLAB2012b 软件中进行仿真,处理器为 Intel(R) Core(TM)i5–4210U CPU @ 1.70 GHz 2.40 GHz,内存 4 GB,Window7 版本 64 位操作系统。

仿真实验采用的随机网络是由 BA 无标度网络的方法随机生成的,网络规模设为 500,1 000,1 500,2 000,2 500,3 000 个节点,且针对给出的网络规模均进行 10 次仿真实验求平均值。在这样的仿真条件下,对最短增广链算法与新算法进行平均运行时间的求解。

4.2 实验结果衡量指标

将可行流流值、运行时间作为实验结果的衡量指标。f:最短增广链算法求得的最大流流值; f :新算法

求得的最大流流值; t :最短增广链算法的平均运行时间; t' :新算法的平均运行时间。

4.3　实验数据统计与分析

通过观察表 1 中两种算法在不同规模网络上的实验数据统计,可以看出最短增广链算法与新算法都能精确地求出网络最大流,而且新算法的平均运行时间低于最短增广链算法。

表 1　两种算法在不同规模网络上的实验数据统计

网络规模	最大流值		平均运行时间/s	
	f	f'	t	t'
500	156	156	1.516 3	1.106 5
1 000	100	100	5.722 0	3.522 3
1 500	65	65	8.850 1	6.419 1
2 000	185	185	23.326 1	15.727 2
2 500	212	212	41.119 0	31.304 5
3 000	251	251	50.165 9	38.261 0

图 4 的实验结果表明,在相同的网络规模下,新算法要比最短增广链算法所用的运行时间短,这与之前的分析相吻合。

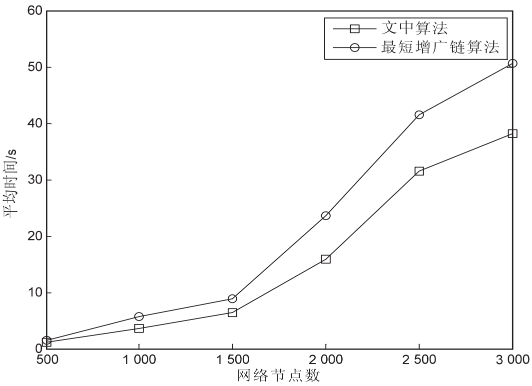

图 4　算法平均运行时间的比较

5　结束语

随着网络技术的发展,网络最大流的应用在现实生活中扮演着越来越重要的角色。文中在最短增广链算法的基础上,提出了一种最短增广链改进算法。该算法删除了原容量网络中没有起作用的弧,并且在分层剩余网络中删除的饱和弧,相应的在原网络中删除该弧,简化了构造剩余网络及分层剩余网络的过程,降低了计算量,使整个算法的执行效率得以提高。通过数值实例验证了该算法的简便性、实用性,并在 BA 无标度网络下进行了仿真实验。实验结果表明,在相同的网络规模下,该算法要比最短增广链算法效率高。

参考文献:

[1]　FAN J, LIAO I F, TAN S X D, et al. Localized on-chip power delivery network optimization via sequence of linear programming[C]//Proceedings of the 7th international symposium on quality electronic design. San Jose, CA, USA: IEEE, 2006:272-277.

[2]　CARLONI C, NOBILE L. Maximum circumferential stress criterion applied to orthotropic materials[J]. Fatigue and Fracture of Engineering Materials and Structures, 2005, 28(9):825-833.

[3]　RISETTI G G, RIZZINI D L, STACHNISS C, et al. Online constraint network optimization for efficient maximum likelihood map learning[C]//IEEE international conference on robotics and automation. Pasadena, CA, USA: IEEE, 2008: 1880-1885.

[4]　DALMAS D, LAKSIMI A. On the method of determination of strain energy release rate during fatigue delamination in composite materials[J]. Applied Composite Materials, 1999, 6(5):327-340.

[5]　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.

[6]　刘焕淋,陈　勇. 通信网图论及应用[M]. 北京:人民邮电出版社,2010:100-104.

[7]　AHUJIA R K, MAGNANTI T L, ORLIN J B. Network flows: theory, algorithms and applications[M]. New Jersey: PREN TICEHALL, 1993.

[8]　KARZANOV A V. Determining the maximal flow in a network by the method of preflows[J]. Soviet Mathematics Doklady, 1974, 15:434-437.

[9]　GOLDBERG A V, RAO S. Beyond the flow decomposition barrier[C]//Proceedings 38th annual symposium on foundations of computer science. Miami Beach, FL, USA: IEEE, 1998:2-11.

[10]　张柏礼,王媛瑗,洪　亮,等. 动态网络中最大流快速增量求解[J]. 东南大学学报:自然科学版,2017,47(3):450-455.

[11]　赵礼峰,严子恒. 基于增广链修复的最大流求解算法[J]. 计算机应用,2015,35(5):1246-1249.

[12]　江锦成,吴立新,杨宜舟,等. 网络最大流的自适应求解算法——SAPR 算法[J]. 计算机应用研究,2014,31(10):2969-2973.

[13]　EDMONDS J, KARP R M. Theoretical improvements in algorithmic efficiency for networks flow problems[J]. Journal of ACM, 1972, 19:248-264.

[14]　BEAMER S, ASANOVIC K, PATTERSON D. Direction-optimizing breadth-first search[C]//International conference for high performance computing, networking, storage and analysis. Salt Lake City, UT, USA: IEEE, 2012:1-10.

[15]　FORGERINI F L, DOROGOVTSEV S N, MENDES J F F. Emergence of scale-free networks from optimization process[J]. Journal of Physics: Conference Series, 2013, 410: 012094.

[16]　GENIO C I, GROSS T, BASSLER K E. All scale-free networks are sparse[J]. Physical Review Letters, 2011, 107(17):178701.

[17]　赵礼峰,纪亚宝. 最大流问题的改进最短增广链算法[J]. 计算机技术与发展,2016,26(8):52-54.