

基于结点的网络最大流算法

胡雄鹰,熊 茜,黎伟东

(武汉工程大学管理学院,湖北 武汉 430074)

摘 要:提出了一个基于结点的网络最大流问题的简单算法,本算法容易理解,计算简便,效率高,还可以很快地找出网络中的瓶颈,并以此来优化整个网络以提高最大流的流量。

关键词:结点;网络;最大流;算法;优化

中图分类号:TP391 文献标识码:A

0 引 言

最大流问题是一类经典的组合优化问题,它也可以看做是特殊的线性规划问题^[1]。这类问题在电力、交通、通信、计算机网络等工程领域和物理、化学等科学领域有着广泛的应用,许多其它的组合优化问题也可以通过最大流问题求解^[2]。

1 网络最大流算法研究进展

1.1 组合算法

假设网络中已经有了一定的流,考虑网络的一条边,一方面,如果该边的流还没有达到该边的容量,则可以继续增加该边上的流;另一方面,也可以减少该流,这相当于在网络中沿该流相反的方向增加流。称由剩余容量不为零的那些边和与现有流方向的相反方向的边构成的网络为剩余网络。组合算法是不断地在剩余网络中增加流量直到找到最大流^[3]。

组合算法一直是最大流算法研究的主流,随着各种组合技术的发展和运用,算法的时间复杂度不断进步。目前具有最好的实际性能的算法也是组合算法,它们是增广链类算法中的 Dinic 阻塞流算法和预流推进类算法中的 Goldberg 和 Tarjan 推进—重标号算法^[4-6]。最近的实验研究表明预流推进算法在实验性能方面更有优势。

1.2 线性规划算法

最大流问题是特殊的线性规划问题。一般求解线性规划问题的方法如单纯形法、椭球法、内点法等^[7-9]都可以用来解决最大流问题。利用最大流问题的特殊性,可以得到比一般算法更有效的算

法。习惯上称利用最大流问题的特殊性得到的单纯形法、内点法为网络单纯形法、网络内点法^[10]。

1.3 其他算法

随机算法和解随机技术也常用于最大流问题。其他算法还有遗传算法,神经网络算法等^[11-12]。

2 基于结点的网络最大流算法

下面提出一个基于结点的网络最大流算法,它属于组合算法中的预流推进类算法。

2.1 基本概念

给定网络 $D=(V,A,C)$,其中 V 代表 D 中结点的集合, A 代表弧的集合。记 $|V|=n, |A|=m$ 。每条弧 (v_i, v_j) 上有权 $c(v_i, v_j)$ (或简写为 c_{ij}) 称作该弧的容量。其中 V_s 和 V_t 分别是 D 中的源点和汇点。流量用 $v(f)$ 表示^[11]。

对于流,有两个明显的要求:一是每条弧上的流量必须是非负的且不能超过该弧的最大通过能力(即该弧的容量);二是中间点的流量为零。

定义1 称流入结点 v_n 的各弧容量和为该结点的流入容量 α_n ,从该结点流出的弧容量和为该结点的流出容量 β_n ,称各结点流入容量与流出容量的差值为堵塞量 $x_n = \alpha_n - \beta_n$ 。

显然,当 $x_n > 0$ 时,它是易堵结点,当 $x_n < 0$ 时,它是不堵结点。

定义2 若弧的起始结点的堵塞量 $x_{n-1} > 0$,结束结点的堵塞量 $x_n < 0$,则该弧称为可优化弧。若弧的起始结点的堵塞量 $x_{n-1} < 0$,结束结点的堵塞量 $x_n > 0$,则该弧称为约束弧。

2.2 算法原理

整个网络源点流出的量与流入汇点的量相等,且各中间结点流入的流量之和必须等于从该点流出的流量之和。

假设从源点以最大的流量流出(各弧按其容量流出),在经过有堵塞的结点($x_n > 0$)时有部分流量会被堵塞在该点,因要满足其平衡条件需把堵塞的部分流量外放。源点流出的总量减去各结点因堵塞而外放的流量为 $\beta_s - |\sum x_n|$,都能流入汇点,此即最大流 $V(f) = \beta_s - |\sum x_n|$ 。

2.3 算法描述

从以上算法原理的描述中可得到基于结点的网络最大流算法如下:

输入:源点 V_s 、网络中间结点和汇点 V_t 。

输出:满足约束条件的网络最大流。

方法:

(1)首先求出各结点的流入容量 α_n 和流出容量 β_n ,算出堵塞量 $x_n = \alpha_n - \beta_n$,标于各结点之上;

(2)分析各弧起始结点和结束结点的堵塞量,找出所有的可优化弧,对可优化弧的容量做出相应减少,该优化不会影响整个网络图的流量;

(3)求出所有 $x_n > 0$ 的和 $\sum x_n$ (其中可优化弧的堵塞量为优化后的);求出源点的流出量 β_s ; $\beta_s - |\sum x_n|$ 即为该网络的最大流 $V(f)$ 。

2.4 算法分析

(1)算法正确性。上述标号算法中,源点流出的总量减去各结点因堵塞而外放的流量,都能流入汇点,此即最大流量,符合最优化原理,因此算法是正确的。

(2)时间复杂性分析。结点数为 n ,弧数为 m ,设每个结点的最大连接弧数为 m_{\max} ,则计算各结点堵塞量的总计算量不大于 $(n-2)m_{\max}$,寻找约束弧的计算量不大于 $2(n-2)$,综上所述,总的算法时间复杂性为 $O(n)$ 。

(3)空间复杂性分析。算法的空间复杂度是指算法需要消耗的空间资源。空间是为了保存中间结果所需要的存储器的大小。本算法可以对结点逐个进行分析,所需空间在计算各结点堵塞量时不大于 m_{\max} ,寻找每个约束弧时不大于 2,计算所需空间极小。

(4)并行时间复杂性分析。并行时间是并行计算所需要的时间,亦即如果多人或多处理机协同计算,解决一个问题所需要的时间。如果所给网络庞大,对于不共享弧的独立结点可以并行地计算它们的结点堵塞量以及寻找约束弧,这样可以加快求解过程。网络越大,各结点的独立性越强,采

用并行机制加速越明显。

2.5 算法示例

有输油管道如图 1 所示。要求算出该网络的最大流。

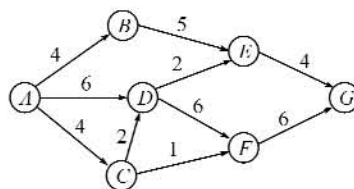


图 1 输油管道网络

Fig. 1 Pipeline network

首先求出各结点的流入容量和流出容量,再算出堵塞量 $x_n = \alpha_n - \beta_n$,标于各结点之上如图 2 所示。

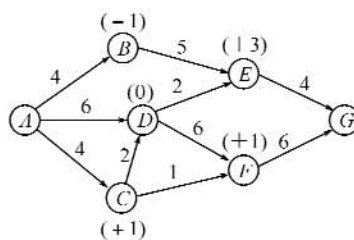


图 2 堵塞量

Fig. 2 Blocking volume

再分析各弧起始结点和结束结点的堵塞量,找出所有的约束弧。图中弧 V_{BE} 为约束弧,优化步骤如图 3 所示。

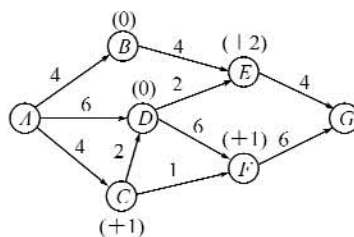


图 3 优化步骤

Fig. 3 Optimization steps

然后求出所有 $x_n > 0$ 的和 $\sum x_n$ (其中约束弧的堵塞量为优化后的):

$$\sum x_n = x_E + x_F + x_C = (+2) + (+1) + (+1) = 4$$

最后求出源点的流出量 β_s , $\beta_s - |\sum x_n|$ 即为该网络的最大流:

$$\beta_s = 4 + 6 + 4 = 14$$

$$V(f) = \beta_s - |\sum x_n| = 14 - 4 = 10$$

同时可以得出弧 v_{BE} 、 v_{CF} 、 v_{FG} 均为可优化弧。相应地提高弧 v_{BE} 、 v_{CF} 、 v_{FG} 的容量即可提高该网络的最大流流量。

在现实生活中,各种输油管道的铺设都是需要很高的费用的,各条管道的费用随着其容量的

增大而增大,在约束弧中有部分的容量自始至终都不曾被使用过,这样就造成了浪费;提高可优化弧的容量能立刻提高该网络的最大流流量.通过该方法可以确定铺设输油管道时各弧的最优容量,降低成本,提高流量.

3 算法比较

通过实例演示了求解网络最大流的一个基于结点的新算法,它与一般运筹学课本上的基于增广链的算法不一样.传统上,基于增广链的算法是根据最大流量最小截量定理,转化为它的对偶问题,即通过寻找网络的最小截集实现的,它是一种基于全局的优化算法.而算法是基于局部(结点)的优化算法,它暗含着可以采用并发机制计算求解.

4 结 语

上述基于结点的网络最大流算法,容易理解,计算简便,效率高,能快速找出网络瓶颈,并以此优化网络以提高最大流的流量.

参考文献:

- [1] 韩伯棠. 管理运筹学[M]. 北京:高等教育出版社, 2005:242-247.
- [2] Huja R K, Magnanti T L, Orlin J B. Algorithms and Applications[M]. Network Flows: Englewood Cliffs, 1993:78-82.
- [3] 陈国良, 梁维发, 沈 鸿. 并行图论算法研究进展[J]. 计算机研究与发展, 1995, 32(9):1-16.
- [4] Goldberg A V, Tarjan R E. A new approach to the maximum flow problem[J]. Journal of the Association for Computer Machine, 1988, 35(4):921-940.
- [5] Ahuja R K, Magnanti T L, Orlin J. B. Network Flows: Theory, Algorithms and Applications[M]. New Jersey: Prentice-Hall, 1993:123-130.
- [6] Goldberg A V. Recent developments in maximum flow algorithms (Invited Lecture)[C]. Arnborg S, Ivansson L, ed. Proc of the Sixth Scandinavian Workshop on Algorithm Theory. Heidelberg: Springer-Verlag, 1998:1-10.
- [7] Dantzig G B. Application of the simplex method to a transportation problem[C]. Activity Analysis and Production and Allocation. New York: Wiley, 1951: 359-373.
- [8] Ford L R, Fulkerson D R. Maximum flow through a network[J]. Canadian Journal of Math, 1956, 8(5): 399-404.
- [9] Dinic E A. Algorithm for solution of a problem of maximum flow in networks with power estimation[J]. Soviet Math Dok, 1970, 11(8):1277-1280.
- [10] Goldberg A V, Rao S. Beyond the flow decomposition barrier[J]. Assoc Comput Mach, 1998, 45(5):783-797.
- [11] 张彦铎, 葛林凤. 一种新的基于 MMAS 的机器人路径规划方法[J]. 武汉工程大学学报, 2009, 31(5):76-79.
- [12] 孔伟, 张彦铎. 基于遗传算法的自主机器人避障方法研究[J]. 武汉工程大学学报, 2008, 30(3):110-113.

Network maximum flow algorithm based on nodes

HU Xiong-ying, XIONG Qian, LI Wei-dong

(School of Management, Wuhan Institute of Technology, Wuhan 430074, China)

Abstract: We provide a simple algorithm on network's max flow problems, based on nodes. The algorithm is easy to understand, and its calculation is simple, and efficient. You can quickly identify the network bottlenecks and optimize the entire network in order to increase the maximum flow.

Key words: nodes; network; maximum flow; algorithm; optimize

本文编辑:陈晓革