

基于重置顶点下标的网络最大流算法

罗甜甜, 赵礼峰

(南京邮电大学 理学院, 江苏 南京 210023)

摘要:通过分析最短增广链算法中好的一面是对顶点分层的理念,不足之处在于需要反复构建分层剩余网络造成算法步骤的繁琐,并且在构建了比原网络更轻易发现增广链的分层剩余网络后,在选取增广链时还是存在随机性,这就导致了某些增广链的丢失,使得最终流值偏小的结果。针对这一现象,提出了一种重置顶点下标的最大流改进算法。该算法首先根据每个顶点在整个网络图中所处位置的重要程度制定相应规则,然后对顶点下标按照此规则重新编号,使得网络图更加清晰直观,从而避免了最短增广链算法中反复构造分层剩余网络图的缺陷。而且新算法还增加了寻找增广链的方法用以规避随机性的缺陷,这也为后面寻找增广链有规可循节约了时间。最后通过数值实例仿真实验验证了新算法的简易性和准确性。

关键词:最大流;顶点层数;源弧容量;汇弧容量;顶点容差

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2020)10-0026-05

doi:10.3969/j.issn.1673-629X.2020.10.005

A Network Maximum Flow Algorithm Based on Reset Vertex Subscript

LUO Tian-tian, ZHAO Li-feng

(School of Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China)

Abstract: By analyzing the positive side of the shortest augmented chain algorithm, the concept of layering the vertices is inadequate because it requires the repeated construction of a layered residual network to cause tedious algorithm steps, and it is easier to find the augmented chain than the original network. After layering the residual network, there is still randomness when selecting the augmented chain, which leads to the loss of some augmented chains, resulting in a small final flow value. Aiming at this phenomenon, an improved maximum flow algorithm for resetting the vertex index is proposed. The algorithm first formulates corresponding rules according to the importance of the position of each vertex in the entire network graph, and then rennumbers the vertex subscripts in accordance with this rule, making the network graph more clear and intuitive, thereby avoiding the defects of repetitive construction of hierarchical residual network graphs in the shortest augmentation chain algorithm. In addition, the proposed algorithm also adds a method to find the augmented chain to avoid the shortcomings of randomness, which also saves time for the future to find the regularized and augmented chain. Finally, numerical examples are used to demonstrate the simplicity and accuracy of the proposed algorithm.

Key words: maximum flow; vertex layer number; source arc capacity; arcing capacity; vertex tolerance

0 引言

最大流问题是一种组合最优化问题,在许多实际的网络系统中都存在着流量和最大流问题,例如铁路运输系统中的车辆流^[1-3]、城市排水系统中的水流问题^[4-6]、控制系统中的信息流问题等^[7-9]都可以抽象出最大流模型,从而转化为最大流问题。所以最大流问题应用广泛,实践性强。求解网络最大流的常用算法可以分为增广路径算法和预流推进算法^[10]。其中后者的编码程度过高,因而前者为经常使用的算法。而

属于增广路径算法的主要有:Ford-Fulkerson 算法^[11], Edmonds-Karp 算法和最短增广链算法^[12], ISAP 算法。设容量网络 D 的顶点数为 m , 弧数为 n , 弧容量的最大值为 c_{\max} , Ford-Fulkerson 算法的算法复杂度则为 $O(mnc_{\max})$, 当容量网络某些弧容量为无理数时,此算法便会陷入无限循环。Edmonds-Karp 对其修正,用宽度优先取代了深度优先。最短增广链算法是将顶点按照其与源点的最短距离分层,分层时用宽度优先法,而寻求增广路径时采用深度优先策略。两者结合效率

收稿日期:2019-12-05

修回日期:2020-04-08

基金项目:国家自然科学基金(61304169)

作者简介:罗甜甜(1994-),女,硕士研究生,研究方向为网络最大流;赵礼峰,教授,硕导,研究方向为图论及其应用、矩阵论。

明显高于 Edmonds-Karp 算法。ISAP 算法是对最短增广链算法的优化,即通过深度优先不断修改距离标号的方式省去每一次的宽度优先。但由于最短增广链算法在构建分层剩余网络中选取增广链仍然存在随机性使得某些增广链缺失,导致算法执行后的结果偏小,失去其准确性。

该文针对最短增广链算法在分层剩余网络后寻找增广链不考虑先后顺序使得最大流值偏小这一问题进行改进,沿用了最短增广链算法通过宽度优先将顶点分层的思想,将顶点根据层数,源弧容量,汇弧容量和顶点容差这四个因素来确定顶点被选择的先后顺序。然后根据相应规则来选取增广链,此规则可以保证最短增广链优先选取,并可以井然有序地快速找出所有增广链。

1 基本概念

定义 1^[13]:容量网络 $D = (V, A, c)$, 其中 V 表示 D 中顶点的集合, A 为 D 中弧的集合, c 表示弧上的容量。

定义 2^[13]:流量。设 f 是容量网络 D 中弧集 A 上的一个实函数, $\forall a = (v_i, v_j) \in A$, 记 $f(a) = f_{ij}$ 。如果函数 $f = \{f_{ij} \mid (v_i, v_j) \in A\}$ 满足守恒条件:

$$\sum_{v_j \in N^+(v_i)} f_{ij} - \sum_{v_j \in N^-(v_i)} f_{ji} = 0, \forall v_i \in V \setminus \{v_s, v_t\}$$

则称 f 是 D 的一个流,此时, f_{ij} 称为通过弧 (v_i, v_j) 上的流量。

定义 3^[13]:顶点层数:在容量网络 D 中,用宽度优先搜索^[14-15]找出从源点 v_s 到任意一点 $v_i (v_i \in V)$ 的最短路的长度 h_i 称为顶点 v_i 的层数,即由 v_s 到 v_i 的路径中所含的最少弧数。(第零层只有一个节点就是源点即 $h_s = 0$)。

定义 4^[16]:剩余网络 $D(f) = (V, A(f), c_f)$, 其中 V 表示容量网络 D 中顶点的集合, f 为 D 上的可行流, $A(f)$ 是 $D(f)$ 上弧的集合,其中,

$$A(f) = A^+(f) \cup A^-(f)$$

$$A^+(f) = \{(v_i, v_j) \mid (v_i, v_j) \in A, f_{ij} < c_{ij}\}$$

$$A^-(f) = \{(v_i, v_j) \mid (v_j, v_i) \in A, f_{ji} > 0\}$$

$c_{ij}(f)$ 为 (v_i, v_j) 关于 f 的剩余容量,其中,

$$c_{ij}(f) = \begin{cases} c_{ij} - f_{ij}, & (v_i, v_j) \in A^+(f) \\ f_{ji}, & (v_i, v_j) \in A^-(f) \end{cases}$$

定义 5^[16]:分层剩余网络记为 $AD(f) = (V(f), A(f), c_f)$, 其中,

$$V(f) = \{v_t\} \cup \{v_i \in V \mid h(v_i) < h(v_t)\}$$

$$A(f) = \{(v_i, v_j) \in A(f) \mid h(v_j) = h(v_i) + 1 < h(v_t)\}$$

$$\cup \{(v_i, v_t) \in A(f) \mid h(v_i) = h(v_t) - 1\}$$

定义 6:源弧容量:由源点 v_s 到顶点 $v_i (v_i \in V)$ 所构成的弧的容量 $c(v_s, v_i)$, 记为 α_i 。

定义 7:汇弧容量:由顶点 $v_i (v_i \in V)$ 到汇点 v_t 所构成的弧的容量 $c(v_i, v_t)$, 记为 β_i 。

定义 8:顶点容差:顶点 $v_i (v_i \in V)$ 的所有出弧容量减去该顶点的所有入弧容量的值为顶点 $v_i (v_i \in V)$ 的容差,记为 γ_i 。

定义 9:中间顶点(除源点、汇点外)标号过程:对于顶点 v_i , 当 $h_i = 1$ 时,求出 α_i 和 γ_i , 将其标号为 $(1, \alpha_i, \gamma_i)$; 当 $1 < h_i < h_t - 1$ 时,求出 γ_i , 将其标号为 (h_i, γ_i) ; 当 $h_i \geq h_t - 1$ 时,求出 β_i 和 γ_i , 将其标号为 (h_i, β_i, γ_i) 。

定义 10:重置顶点下标规则:已知网络 D , 其顶点数为 n , 则将源点 v_s 重置为 v_1 , 汇点 v_t 重置为 v_n 。除源点汇点外,中间顶点的重置标号依次为 v_2, v_3, \dots, v_{n-1} 。首先按照顶点层数 h_i 由低至高的顺序依次由小到大编号重置; 当 h_i 相等时,分三种情况考虑:

(1) $h_i = 1$, 按照源弧容量 α_i 由低至高的顺序依次由小到大编号重置, 当 α_i 相等时,再按照容差 γ_i 由低至高的顺序依次由小到大编号重置;

(2) 当最高层的顶点唯一,且 $1 < h_i < h_t - 1 (h_t > 3)$ 时,当最高层顶点不唯一,且 $1 < h_i < h_t (h_t > 3)$ 时,都按照容差 γ_i 由低至高的顺序依次由小到大编号重置;

(3) 当最高层的顶点唯一,且 $h_i = h_t - 1$ 时,或者当最高层的顶点不唯一,且 $h_i = h_t$ 时,都按照汇弧容量 β_i 由低至高的顺序依次由小到大编号重置, 当 β_i 相等时,再按照容差 γ_i 由低至高的顺序依次由小到大编号重置。

定理:设 f 是容量网络 D 中的可行流,则 f 是 D 的最大流当且仅当 D 中不存在 f 增广链。

2 新算法思想及步骤

2.1 算法思想

求网络最大流^[17]的基本思想是根据定理 1, 即从容量网络 D 中的一个可行流出发, 寻找增广链进行增广, 直至 D 中不存在增广链为止。根据此思路, 目的是寻找增广链, 关键是确定选取增广链的先后顺序。首先是根据一定规则对顶点下标进行重置, 形成与原网络 D 相对应的新网络 D_1 。然后按照优先选取顶点标号值较大的增广链增广的原则在 D_1 中寻找增广链, 直至不存在从源点到汇点的增广链即可结束该算法。

2.2 算法步骤

Step1: 利用宽度优先搜索求出原容量网络 D 中各个顶点的层数, 再根据定义 9 对中间顶点进行标号。

Step2: 根据定义 10 对顶点下标进行重置, 形成对

应的新网络 D_1 。

Step3: 初始化新网络 D_1 , 令 D_1 中每条弧的流量为 0。

Step4: 从顶点 v_1 出发, 判断 D_1 中是否存在一条从 v_1 到 v_n (n 为网络 D_1 的顶点个数) 的增广链, 若存在转 Step5, 否则转 Step7。

Step5: 从顶点 v_1 出发, 沿着与其关联弧所在顶点下标值较大的原则找一条从 v_1 到 v_n 的增广链, 若存在转 Step6, 否则转 Step7。

Step6: 调整, 求该链上各弧容量的最小值 δ , 并将此增广链中对应弧的容量后面标上 δ , 其中 $\delta = \min\{c_{ij} - f_{ij}\}$ 。在饱和弧上标上终止弧标志“||”, 增广后转 Step4。

Step7: 计算所有以 v_1 为发点的弧的流量和, 得到最大流值 $v(f)$, 算法结束。

3 新算法可行性与复杂度分析

3.1 可行性分析

新算法的主要思想是沿着所规定好的原则寻找增广路径, 每次增广至少使一条弧达到饱和。假设网络 D 中共有 m 条弧, 那么最多通过 m 次增广后, 网络 D 中就不存在从源点到汇点的增广链, 即算法结束。说明算法经过有限次步骤就会终止, 不会陷入无限循环。

3.2 复杂度分析

新算法主要分为两个过程: 一是标记、比较和编号过程, 通过这些准备工作构建新网络流图; 二是增广过程, 沿着顶点下标值大的原则寻找增广链。设容量网络 D 的顶点数为 n , 弧数为 m 。该算法执行时, 第一个过程属于准备过程只执行一次, 而执行一次准备工作的计算量分别是: (1) 计算层数 $O(n)$ 次; (2) 计算容差最多需要 $O((n-2)m)$ 次; (3) 比较大小重新编号最多需要 $O(2n \log_2 n)$ 次。第二个过程最多执行 $O(m)$ 次, 于是该算法的时间复杂度为:

$$O(n) + O((n-2)m) + O(2n \log_2 n) + O(m) = O(2n \log_2 n + nm)$$

4 应用实例

例: 求出图 1 的容量网络 D 中从源点 v_s 到汇点 v_t 的网络最大流。

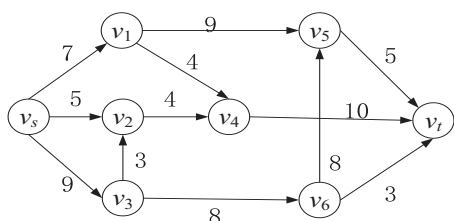


图 1 容量网络 D

解: 方法一(文中算法):

(1) 首先根据宽度优先搜索求出各顶点层数, 得:

$$h_s = 0, h_1 = 1, h_2 = 1, h_3 = 1$$

$$h_4 = 2, h_5 = 2, h_6 = 2, h_t = 3$$

(2) 对于 $h_i = 1$, 要求出 α_i 和 γ_i , 于是可得:

$$\begin{cases} \alpha_1 = 7, \gamma_1 = 6 \\ \alpha_2 = 5, \gamma_2 = -1 \\ \alpha_3 = 9, \gamma_3 = 2 \end{cases}$$

因为 $h_t = 3 < 4$, 对于 $1 < h_i < h_t$, 要求出 β_i 和 γ_i ,

于是可得:

$$\begin{cases} \beta_1 = 10, \gamma_1 = 2 \\ \beta_2 = 5, \gamma_2 = -12 \\ \beta_3 = 3, \gamma_3 = 3 \end{cases}$$

(3) 由(2)所得数据对顶点标号, 如图 2 所示。

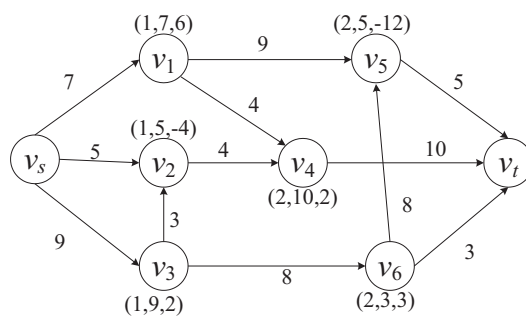


图 2 标号后的容量网络 D

(4) 再根据定义 8 重置顶点下标: 令

$$v_s = v_1, v_t = v_8, \begin{cases} v_1 = v_3 \\ v_2 = v_2 \\ v_3 = v_4 \end{cases}, \begin{cases} v_4 = v_7 \\ v_5 = v_6 \\ v_6 = v_5 \end{cases}$$

重新构造容量网络 D_1 , 如图 3 所示。

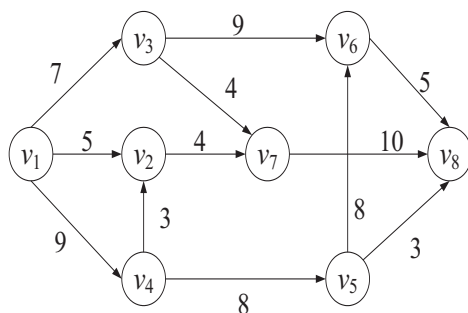


图 3 容量网络 D_1

(5) 在 D_1 中按照优先选取顶点下标值较大的原则寻找增广链:

$$p_1 = v_1 - v_4 - v_5 - v_8, \delta_1 = 3$$

$$p_2 = v_1 - v_4 - v_5 - v_6 - v_8, \delta_2 = 5$$

$$p_3 = v_1 - v_4 - v_2 - v_7 - v_8, \delta_3 = 1$$

$$p_4 = v_1 - v_3 - v_7 - v_8, \delta_4 = 4$$

$$p_5 = v_1 - v_2 - v_7 - v_8, \delta_5 = 3$$

增广后的网络流图如图 4 所示。

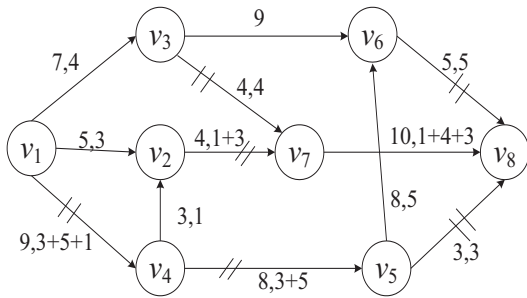


图 4 可行流 f

在图 4 中找不到 v_1 到 v_8 的增广链了,故图 4 即为最大流 f ,计算最大流值为 $V(f) = 4 + 3 + 3 + 5 + 1 = 16$ 。

方法二(最短增广链算法):

(1)在 D 中取 f_1 为初始可行流,令 f_1 为零流。然后构建分层剩余网络 $AD(f_1)$,见图 5。

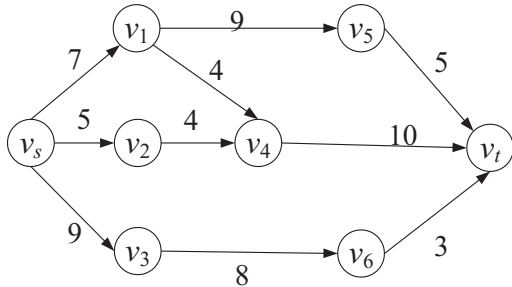


图 5 分层剩余网络 $AD(f_1)$

(2)在图 5 中找增广链,找到了如下几条增广链:

$$p_1 = v_s - v_1 - v_5 - v_t, \delta_1 = 5$$

$$p_2 = v_s - v_1 - v_4 - v_t, \delta_2 = 2$$

$$p_3 = v_s - v_2 - v_4 - v_t, \delta_3 = 4$$

$$p_4 = v_s - v_3 - v_6 - v_t, \delta_4 = 3$$

增广之后,得到可行流 f_2 ,见图 6。

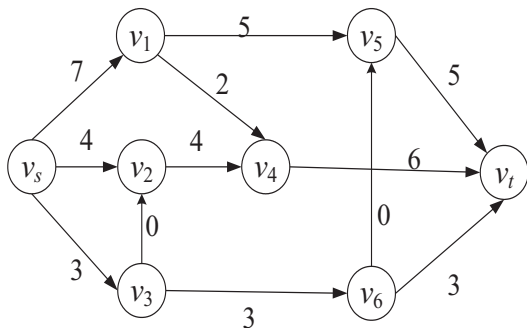


图 6 可行流 f_2

(3)继续构建剩余网络 $D(f_2)$,见图 7。发现 $D(f_2)$ 中不存在 (v_s, v_t) 路,算法结束。 f_2 即为容量网络 D 的最大流,在图 6 中计算与源点关联的所有弧的流量和为 14,即最大流的流值为 14。可见与方法一相比流值偏小。

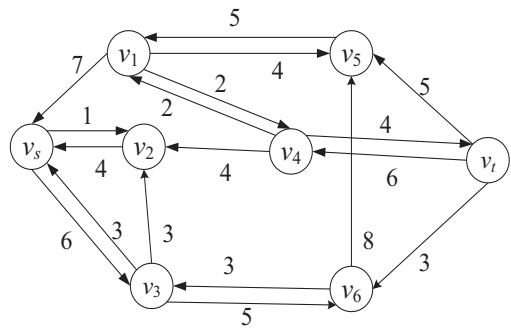


图 7 剩余网络 $D(f_2)$

5 算法的仿真与比较

5.1 实验环境与设计

该算法使用的实验仿真平台是 MATLAB2016a,处理器为 Intel(R) Core(TM) i3-4030U CPU @ 1.90 GHz,内存 4 GB,Windows7 版本 64 位操作系统。

仿真实验采用的是 BA 无标度随机网络,并将网络规模的顶点数依次设为 200,400,600,800,1 000,1 200,1 400,1 600,1 800,2 000。然后在给定的网络规模下,对新算法和最短增广链算法进行 10 次仿真实验。其中参数 f_1 和 t_1 分别表示最短增广链算法的最大流值和它的运行时间的平均值,参数 f_2 和 t_2 分别表示新算法的最大流值和其运行时间的平均值。

5.2 实验结果统计与分析

根据表 1 的内容可以看到两种算法在不同的网络规模下得出的最大流值情况以及算法的平均运行时间。明显可以得出新算法比最短增广链算法的结果更精准。由图 8 可以看到新算法的运行时间较短。

表 1 新算法与最短增广链算法在不同网络规模的运行数据

顶点数量	最大流值		平均运行时间/s	
	f_1	f_2	t_1	t_2
200	63	3	0.034 4	0.022 2
400	67	89	0.043 3	0.038 4
600	116	158	0.047 6	0.040 8
800	9	92	0.043 0	0.034 8
1 000	41	77	0.042 5	0.036 3
1 200	56	144	0.054 6	0.051 7
1 400	119	159	0.058 1	0.053 7
1 600	81	122	0.039 4	0.037 9
1 800	62	82	0.049 8	0.035 9
2 000	37	54	0.041 1	0.023 8

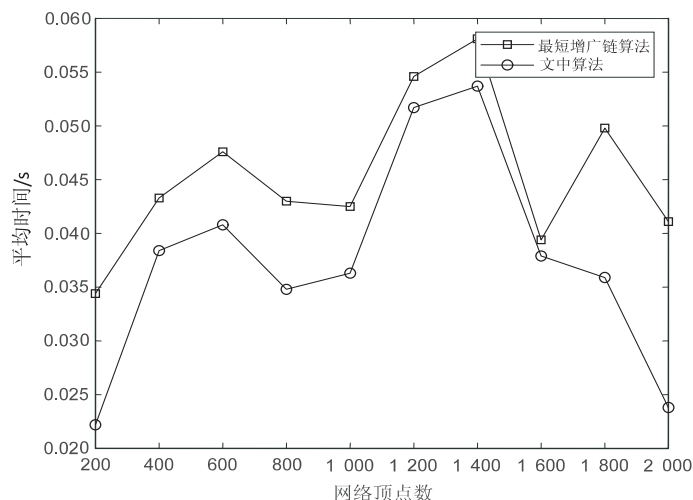


图8 新算法与最短增广链算法的平均运行时间

6 结束语

文中针对最短增广链算法在构建分层剩余网络后选取增广链增广的先后顺序不当而影响最大流值偏小的问题进行了改进处理。首先对顶点标号根据其在整个网络图所处位置按一定规律重新编号,为后面寻找增广链有规可循且节约了时间,也使得网络图更加清晰直观。通过很多实例和实验仿真证实了算法的准确性和高效性。

参考文献:

- [1] 陈兴菊. 关于开放小区车辆通行模型的研究[J]. 科技创新与应用, 2019(13): 41-42.
- [2] 侯志伟, 申涵瑞, 谢强, 等. 深圳关内外交通拥堵探究与治理[J]. 数学建模及其应用, 2013, 2(3-4): 14-22.
- [3] 寇玮华, 朱雪丽, 张聪聪. 交通网络两个相邻结点之间有流量约束的最大流分配算法[J]. 交通运输工程与信息学报, 2010, 8(1): 7-13.
- [4] 杨秋侠, 赵瑞垠. 基于网络最大流的城市雨水管网系统脆弱性评价[J]. 系统工程理论与实践, 2018, 38(11): 2987-2992.
- [5] 高芸, 张海艳, 徐建新, 等. 提高贾鲁河水系抵御暴雨能力的研究[J]. 人民黄河, 2014, 36(3): 7-9.
- [6] 耿少阳. 最大流算法在城市排水管网中的应用[J]. 科技通报, 2012, 28(4): 20-21.
- [7] 章小卫. 基于MAS的应急系统任务调度问题研究[D]. 扬州: 扬州大学, 2009.
- [8] 吴艳. 应急系统选址问题的优化[D]. 西安: 西安电子科技大学, 2008.
- [9] 唐少先, 陈建二. 分布式系统中有向流的最优调控算法[J]. 计算技术与自动化, 2001, 20(3): 45-47.
- [10] GOLDBERG A V, RAO S. Beyond the flow decomposition barrier[J]. Journal of ACM, 1998, 45(5): 783-797.
- [11] FORD L R, FULKERSON D R. Maximum flow through a network[J]. Canadian Journal of Math., 1956, 8(5): 399-404.
- [12] EDMONDS J, KARP R M. Theoretical improvements in algorithmic efficiency for networks flow problems[J]. Journal of ACM, 1972, 19(2): 248-264.
- [13] 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.
- [14] 李明哲, 金俊, 石端银. 图论及其算法[M]. 北京: 机械工业出版社, 2010: 20-47.
- [15] 匡桂娟. 广度优先搜索算法在互连网络通信中的应用[D]. 青岛: 青岛大学, 2005.
- [16] 田丰, 张运清. 图与网络流理论[M]. 北京: 科学出版社, 2015.
- [17] 邵丽萍, 赵礼峰. 基于宽度优先的网络最大流求解算法[J]. 计算机技术与发展, 2019, 29(6): 62-65.