

最大流在信息学竞赛中应用的一个模型

江 涛

关键字：

网络 可行流 最大流 附加网络

无源汇 必要弧 流的分离 有上下界的最大流 建模

引言：

最大流类的模型在当今信息学比赛中有相当广泛的应用。但在教学中，发现很多同学对流模型的原理和变形并不掌握，只是记下经典的算法和题目，以便比赛中套用。

这当然有很大的局限性，也不是学习之正道。本讲想通过对最大流模型，特别是附加网络的一些分析、讨论，达到抛砖引玉目的。

一、网络与流的概念

一个实例：运输网络

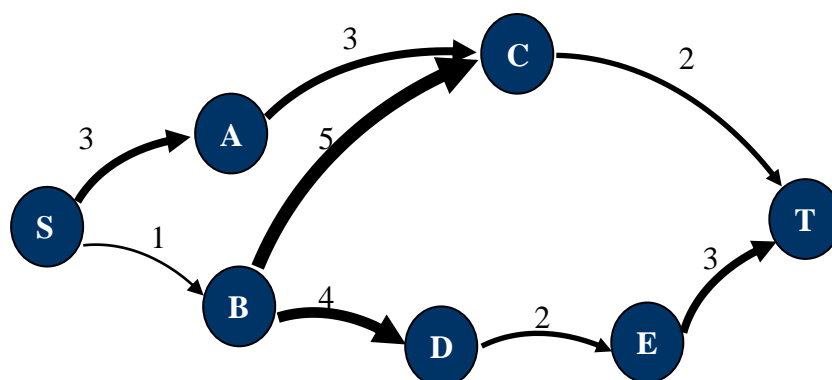


图 1.1

网络定义：

- 一个有向图 $G=(V, E)$;
- 有两个特别的点：源点 s 、汇点 t ;
- 图中每条边 $(u,v) \in E$ ，有一个非负值的容量 $C(u,v)$

记为 $G=(V, E, C)$

网络三要素：点、边、容量

可行流定义：

是网络 G 上的一个“流”，即每条边上有个“流量” $P(u,v)$ ，要满足下面两个条件：

流的容量限制---弧：

$$0 \leq P(u, v) \leq C(u, v) \quad \text{对任意弧 } (u, v) \text{ --- 有向边}$$

流的平衡---点：

除源点和汇点，对任意中间点有：流入 u 的“流量”与流出 u 的“流量”相等。即：

$$\forall u \in V - \{s, t\} \text{ 有}$$

$$\sum_{x \in V} P(x, u) - \sum_{x \in V} P(u, x) = 0$$

网络的流量：源点的净流出“流量”或汇点的净流入“流量”。即：

$$\sum_{x \in V} P(s, x) - \sum_{x \in V} P(x, s) = \sum_{x \in V} P(x, t) - \sum_{x \in V} P(t, x)$$

注意，我们这里说的流量是一种速率，而不是指总量。联系上面所说的实例，下面是一个流量为1的可行流：

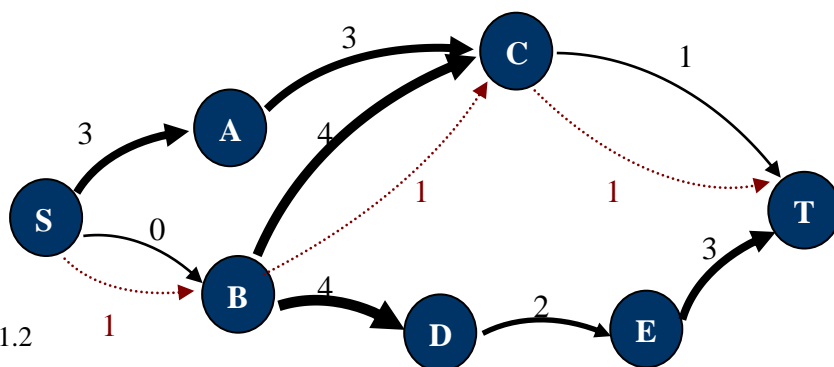


图 1.2

标准图示法：

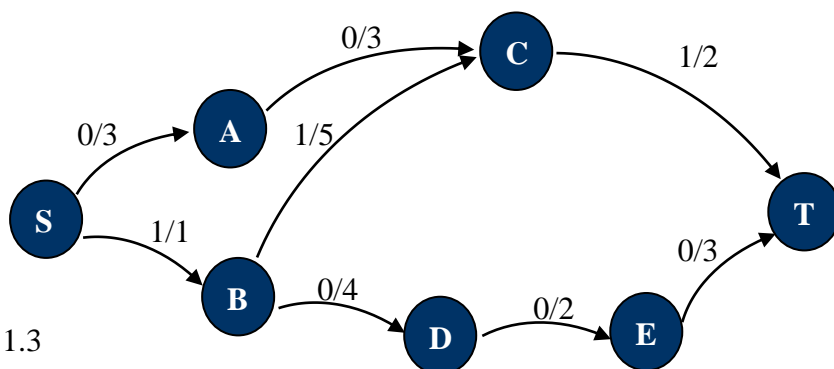


图 1.3

例 1.1 有一个 $n*m$ 的国际棋盘，上面有一些格子被挖掉，即不能放棋子，现求最多能放多少个棋子“车”，并保证它们不互相攻击(即：不在同一行，也不在同一列)？

分析：

- 1) 行、列限制---最多只能一个车控制；
- 2) 车对行、列的影响---一个车控制一个行和边；

例子：

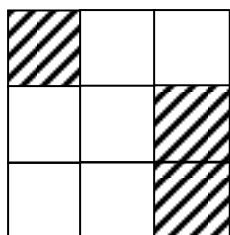


图 1.4

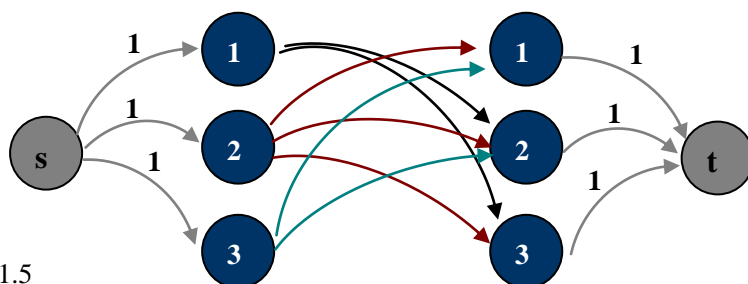


图 1.5

显然，我们要求车最多，也就是求流量最大---最大流问题。下面是一个解：

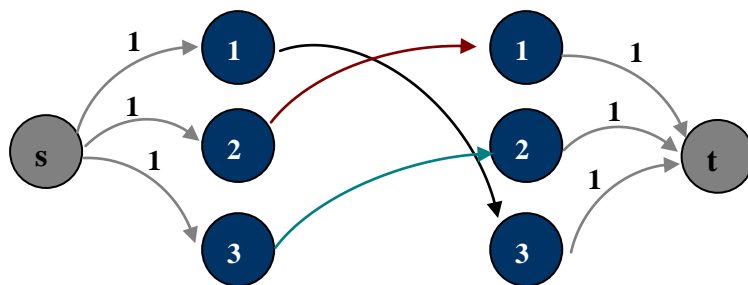


图 1.6

即(1,3)、(2,1)、(3,2)格上各放一个车，可得到一个最优方案。

二、最大流问题

寻找网络 G 上可能的最大流量(和一个有最大流量的可行流方案)，即为网络 G 上的最大流问题。

我们再来看看图 1.1 的运输网络例子，我们可能通过改进图 1.3 得到下面这样的可行流：

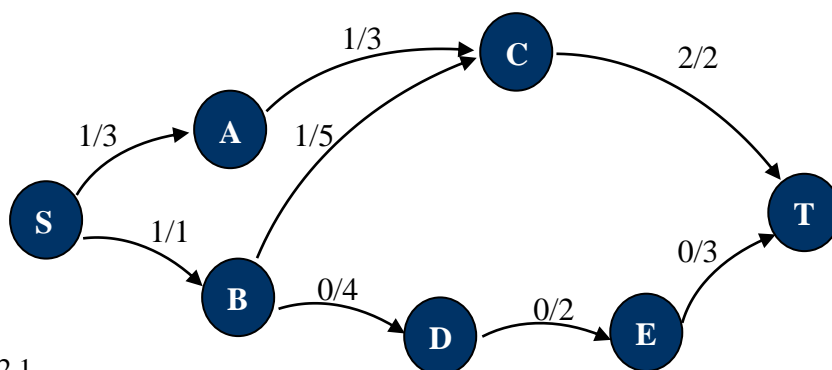


图 2.1

求解过程中的困惑：

[问题 2.1]流量还可能增加吗？

[问题 2.2]如果能增加，怎样改进？

三、最大流算法的核心---增广路径

退流的概念---后向弧

仔细分析图 2.1，我们发现，流量是可以增加的：

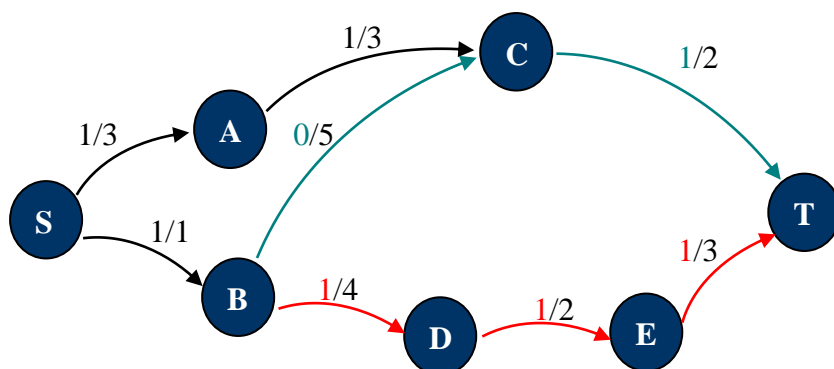


图 3.1

把一个流量弧(B,C)和(C,T)上的流退回到 B 点，改道从 B-D-E-T 走，就可以增加流量了，如下图：

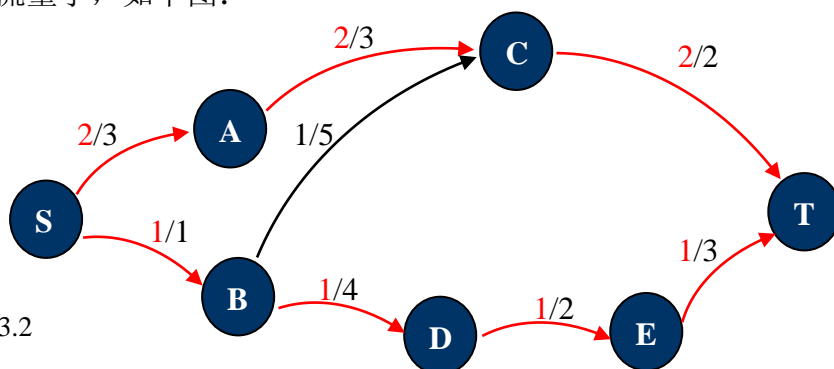


图 3.2

图3.1不能“直接寻找增大流的路径”，是因为当初有些弧上的流选择不“恰当”，要“退流”。

一种直观的想法就是：调整或重新搜索“当初的选择”---难！

能不能保留以前的工作，而在这之上改进呢？ 经过专家们研究发现，加入退流思想---**后向弧**，就能再次“直接寻找增大流的路径”。

增广路径(可改进路径)的定义

若 P 是网络中连结源点 s 和汇点 t 的一条路，我们定义路的方向是从 s 到 t ，则路上的弧有两种：

- 前向弧---弧的方向与路的方向一致。前向弧的全体记为 P^+ ；
- 后向弧---弧的方向与路的方向相反。后向弧的全体记为 P^- ；

设 F 是一个可行流， P 是从 s 到 t 的一条路，若 P 满足下列条件：
在 P^+ 的所有前向弧 (u,v) 上， $0 \leq f(u,v) < C(u,v)$ ；

在 P -的所有后向弧 (u,v) 上, $0 < f(u,v) \leq C(u,v)$;
 则称 P 是关于可行流 F 的一条可增广路径。

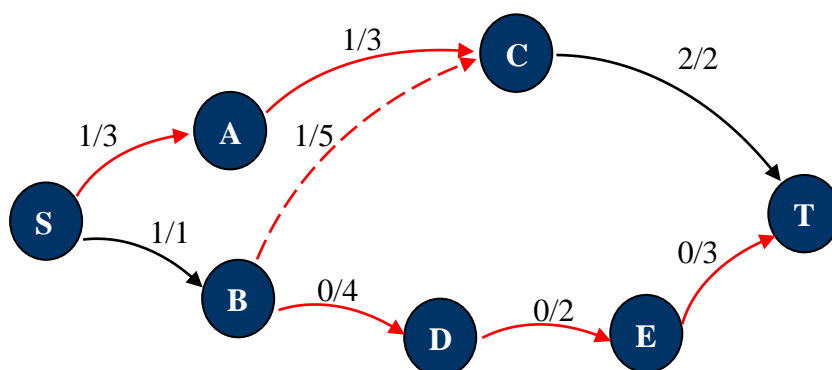


图 3.3

本图中: $S-A-C-B-D-E-T$ 为一增广路径。其中 (C,B) 为后向弧, 其它为前向弧。

在增广路径上的改进算法:

1) 求路径可改进量;

$$C_f(P) = \min_{(u,v) \in P} \{ \text{前向弧 } C(u,v) - f(u,v), \text{ 后向弧 } f(v,u) \}$$

2) 修改增广路径上的流量;

四、附加网络 1---残留网络

由于要考虑前向弧、后向弧, 分析、描述时不简洁, 在图 2.1 上直观看也不容易看出增广路径。

因此我们把已有的流量从容量中分离出来表示, 引入一个与原问题等价的附加网络 1: 残留网络。

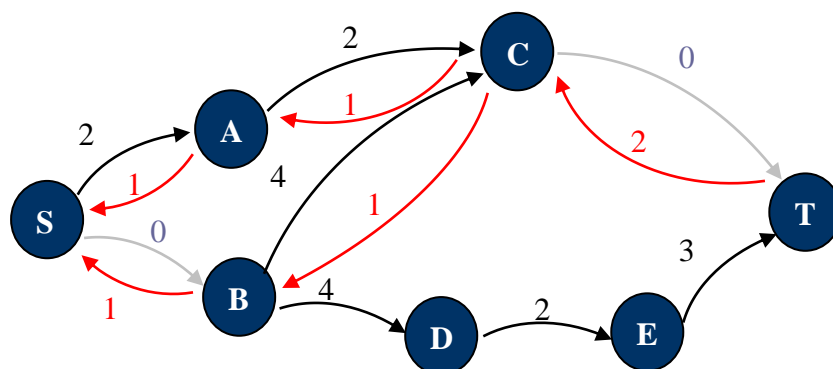


图 4.1

其中, 前向弧(黑色)上的容量为“剩余容量” $= C(u,v) - f(u,v)$; 后向弧(红色)上的容量为“可退流量” $= f(v, u)$ 。

改造后的网如下：

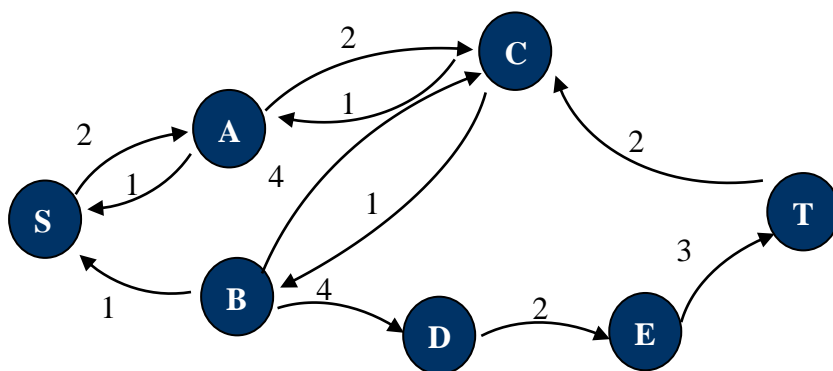


图 4.2

在这张图上，我们找增广路径显的非常直观了！

结合增广路径，我们有如下定理：

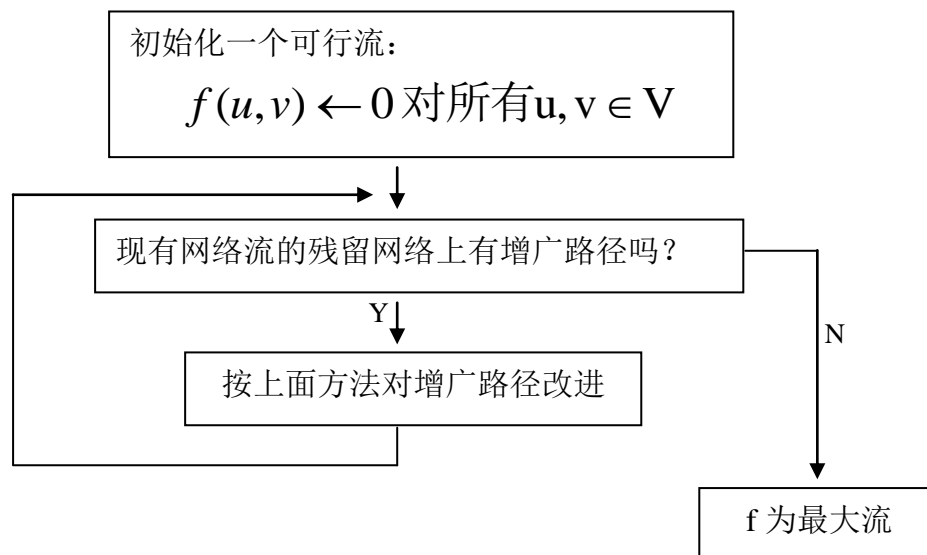
最大流定理

如果残留网络上找不到增广路径，则当前流为最大流；反之，如果当前流不为最大流，则一定有增广路径。

证明涉及**最小割**概念，不是本文重点，由于时间关系，从略。

至此，[问题 2.1]和[问题 2.2]在这个最大流定理中同时获得解决。

求最大流的基本思想：



基于这种思想的算法，关键之处在于怎样找增广路径。常用方法有：

- 深度搜索 dfs
- 宽度搜索 bfs
- 优先搜索 pfs---即类似 Dijkstra 算法的标号法。

另外，求**最小费用最大流问题**(每条边有个费用)，只要把每次“找增广路径”改为“找

费用最小的增广路径”即可。(证明从略)

五、小结一

前面几小节，回顾了最大流算法的一些基本概念和原理，特别提到了把已有流量分离出来单独表示的方法：残留网络。这个与原问题等价的附加网络使问题的描述、思考方便简洁而统一。

注意这几个关键词：

流 分离 单独表示 等价 方便简洁

它们表达了一种思路，一种分析、表示最大流问题的方法。

前几年的国家集训队有几篇关于最大流的论文，作者关注的就是分析问题怎样建立数学模型。

如广东北江中学的李伟星在论文《最大流》的摘要中说：

“许多问题可以先转化为网络流问题，再运用最大流算法加以解决。而发现问题本质，根据最大流算法的特点，设计与之相配的数学模型是运用最大流算法解决问题的重要步骤。”

福建师大附中的江鹏同学在论文《从一道题目的解法试谈网络流的构造与算法》的引论中也有这样一句话：

“网络流在具体问题中的应用，最具挑战性的部分是模型的构造。这没用现成的模式可以套用，需要对各种网络流的性质了如指掌（比如点有容量、容量有上下限、多重边等等），并且归纳总结一些经验，发挥我们的创造性。”

是的，在信息学竞赛中很多最大流相关问题没有现成模式可以套用，但解决最大流问题的原理和思想是相通的，前面所说的构造等价的附加网络的思路和方法就是一种可以借用的“现成模式”。

为了进一步的讨论，先看一个更复杂的问题：

例题 5.1 变形一笔画问题

在一个有向图，判断能不能一笔画访问所有的边，但有些弧上可以画多次。我们用容量 $C(u,v)$ 表示弧 (u,v) 最多可以重复画的次数。

分析：

由于除了开始点和结束点，每个点进入次数与离开次数是相同的。因此满足流平衡条件，可以想到用流模型做。但这里每条弧都要画到，即最少要画一次。因此，成为有上下界的流问题。

这种流的容量有上下界时求解的问题，在信息学比赛中也很常见。下面将对这类问题做进一步讨论，我们会发现，运用“流分离、构造等价附加网”的思想，解决的难度并不比前面最大流问题大。

六、附加网络 2---无源汇的可行流

如果每个点都考虑流的平衡，而没有了源点 s 和汇点 t ，则我们称为**无源汇的可行流**。

有一种简单的方法可以将普通的网络上的可行流要改成无源汇的可行流：

加一条边 (t,s) ，容量为可根据需要设定为 $+\infty$ 或其它值。

例 1.1 的这种改变如下：

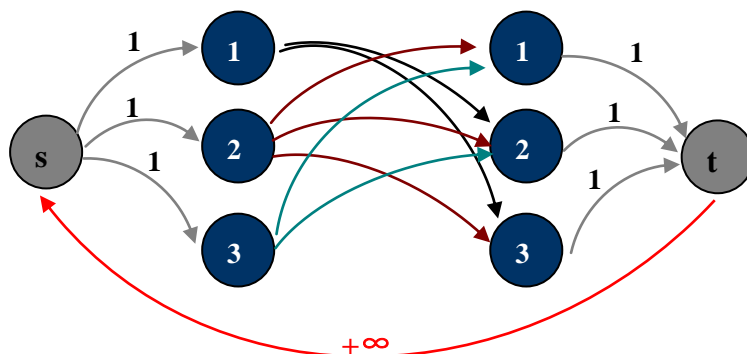


图 6.1

例题 5.1 变形一笔画问题的网络改变模型表示如下：

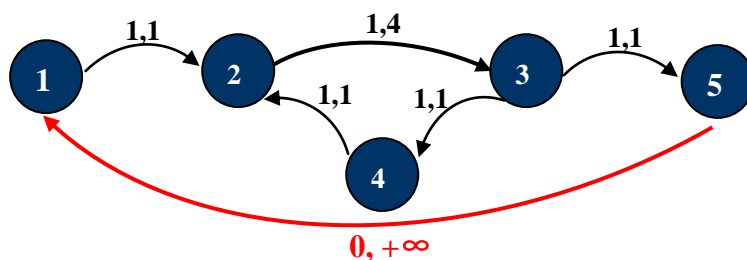


图 6.2

注：我们用弧上的 a,b 数对表示容量的上下界。

无源汇的可行流问题(特指有上下界的流网络)：

在一个有上下界的流网络 G 中，不设源和汇，但要求任意一个点 i 都满足流量平衡条件：

$$\sum_{(u,i) \in E} f(u,i) = \sum_{(i,v) \in E} f(i,v)$$

且每条边 (u,v) 都满足容量限制 $B(u,v) \leq f(u,v) \leq C(u,v)$ 的条件下，寻找一个可行流 f ，或指出这样的可行流不存在。不妨称这个问题为**无源汇的可行流**。

---参见周源的《一种简易的方法求解流量有上下界的网络中网络流问题》

上图中把源、汇连接起来了，没有了源和汇，怎么求其上的可行流呢？

是的，没有了源、汇，以前的最大流算法就成了无的放矢，一定要有源和汇。那我们为什么还要把有源、汇的问题改成无源、汇呢？答案是**破旧立新**：

没了旧的源、汇，我们可以更加自由地按需要设立新的源、汇。从而使网络模型变成一个新的附加网络！

七、附加网络 3---必要性流的分离

对于有上下界的最大流网络流问题，我们可以看成是在满足“必要”的下界情况时，求最大流(或最小费用等问题)。直观的一种思路就是：

- 求一个满足下界的流 f_1 ;
- 在 f_1 基础上用寻找增广路径的方法，扩大流，直到没有增广路径；

实现的关键点在于：

- **问题 7.1:** 怎样求满足下界而又不超过上界的一个流 f_1 ?
- **问题 7.2:** 怎样增大流 f_1 ，而又同时不破坏下界？

总体上看，下界是一条弧必需要满足的**确定值**，我们能不能把这个下界“分离”出去，从而使问题转为下界为 0，也就是普通的最大流问题的可行流问题？

先用一个类似的实例来分析说明：

[例题 7.1]:

N 个石子，分成 M 堆，要求第 i 堆的石子数大于给定的正整数 C_i ，问有多少种方法？

[转化问题]:

$$N' = N - \sum C_i \text{ 个石子，要求分成 } M \text{ 堆，问有多少种排列方法？}$$

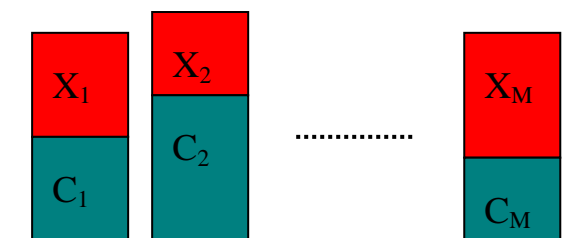


图 7.1

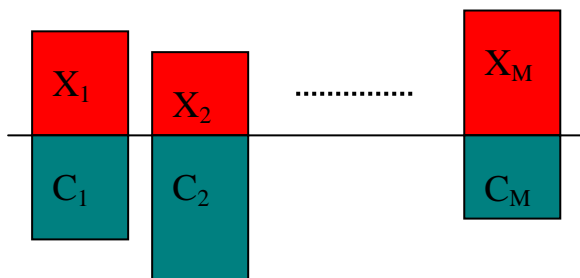


图 7.2

[组合数学问题]:

一列 N' 个 1 中间有 $N'-1$ 个可分隔点，选 $M-1$ 个，把 1 分成 M 段，有多少方法？

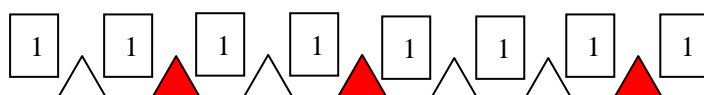


图 7.3

[结果]: $C(N' - 1, M - 1)$

类似地，我们设法运用**流分离**的思想，找一个等价的**附加网络 3**，使**问题 7.1** 和**问题 7.2** 得到方便而统一地解决。

(一)观察一条路径情况

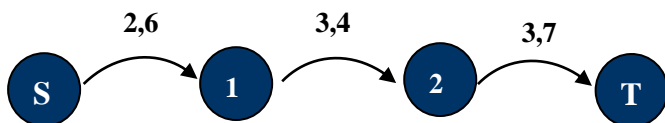


图 7.4

如同例 7.1 直接把容量下界减掉怎样？

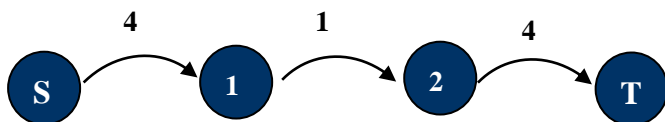


图 7.5

显然，如果图 7.5 中的流想通过加上下界来对应地得到图 7.4 中的解是不可能的。因此，这里流的分离不能用简单减掉的方法，而应该把一个弧分离成两个弧。即加一个容量等于下界的必要弧，使可行流分离在这两个弧上。如下图：

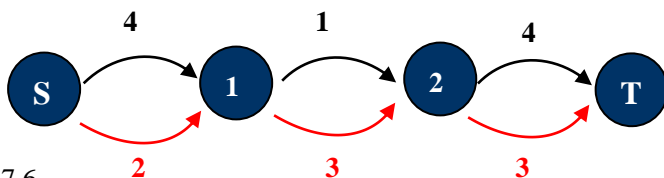


图 7.6

一个无源汇的可行流的方案一定是必要弧是满的。因此现在我们先要找一个把必要弧充满的可行流(当然也要满足上界限制)。

现在我们的目光焦点是必要弧，把他们集中起来：

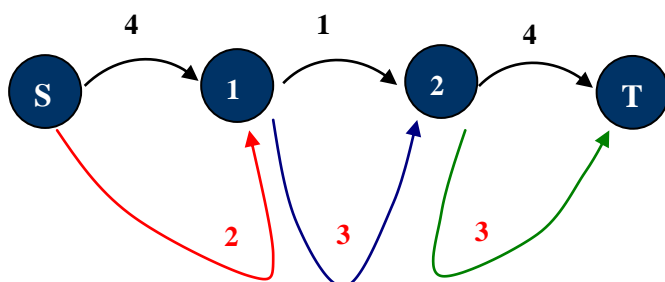


图 7.7

由于必要弧都是要饱和的，显然与下面图是等价的。

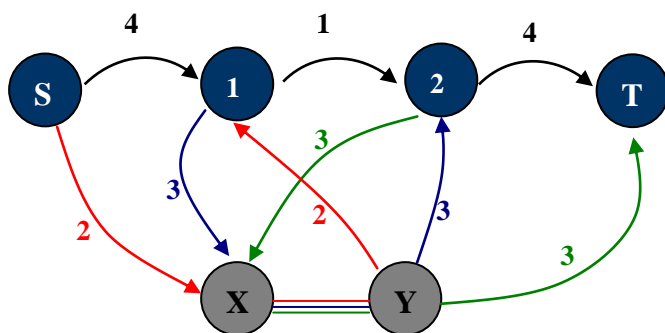


图 7.8

进一步，加弧(T,S)，容量不限，成无源汇可行流问题。再去除 X,Y 之间的连线，使 Y、X 分别成为新的源和汇。

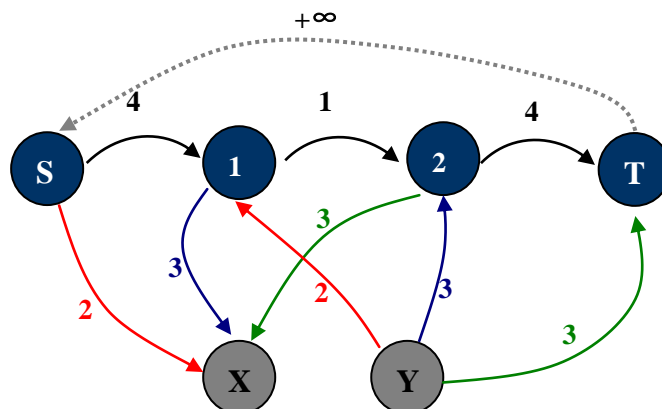


图 7.9

如果 Y 到 X 的最大流能为 $2+3+3$ ，则显然有：

- 必要弧为满
- 满足容量上界限制
- 保持流平衡

(二)网络整体情况

再来看看例 5.1，先分离必要弧：

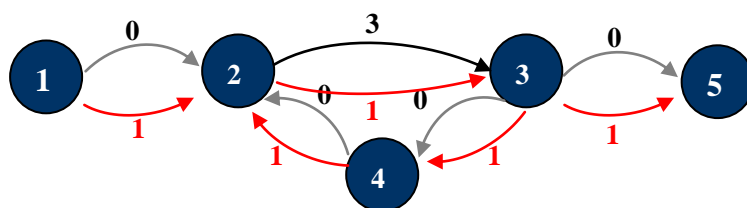


图 7.10

改造：由于可行流的源 S 流出与汇 T 流出应该是相等的，加条边(T,S)，容量 $+\infty$ 。割开所有必要弧，加两个附加源 Y 和汇 X：

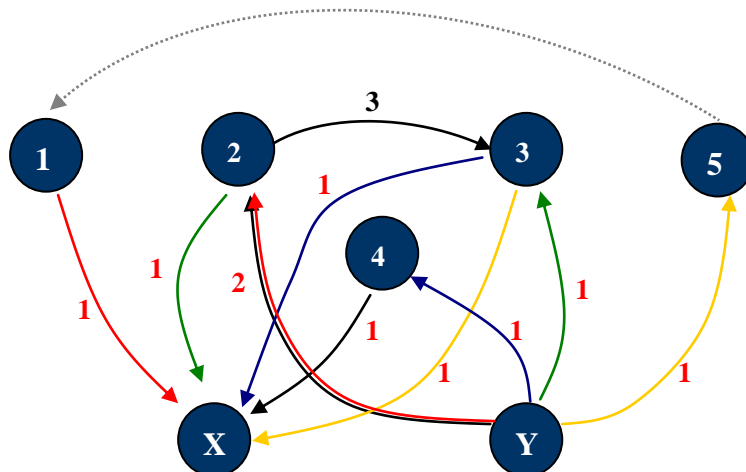


图 7.11

至此：问题成功转化为求 Y 到 X 的普通最大流问题。我们称这个改造后的网络模型为附加图 3。

回顾本节我们要解决的问题：

- **问题 7.1:** 怎样求满足下界而又不超过上界的一个流 f_1 ?
- **问题 7.2:** 怎样增大流 f_1 ，而又同时不破坏下界?

当求出**附加网络 3**的最大流为 $1+1+1+1+1=5$ 时，我们再反过来做：把“进 X”和“出 Y”的对应边连上，则找到一个有上下界容量限制和无源汇可行流。再把弧(T,S)拿走，则**问题 7.1** 解决。

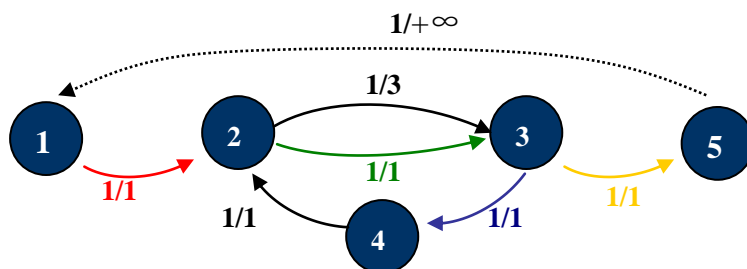


图 7.12

注意：原问题可行流存在无解情况，即附加网络 3 的最大流不能把源(或汇)相连的弧饱和。不同于普通最大流：因下界为 0 时，一定有解。

在得到一个可行流 f_1 之后，现在我们再来看看**问题 7.2**。

再去掉边(T,S)----上图即为弧(5,1)，还原成有源汇最大流的原问题。

如果要求这时的最大流，则可在这个基础上，找**增广路径**来增大流，最终求得一个符合要求的最大流方案。

但是要注意的是，对**必要弧**，我们不能退流，因此相应的残留网络对必要弧要单独处理，或直接暂时拿走，再做**附加网络 2**，如果对上例处理，则：

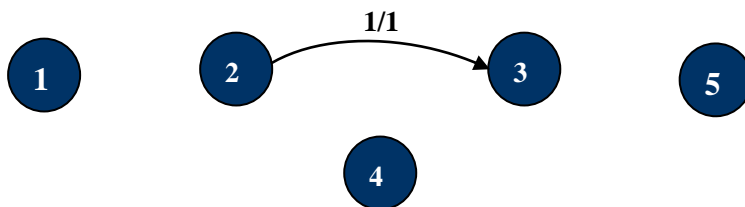


图 7.13

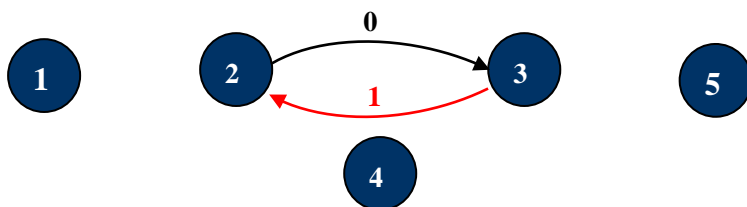
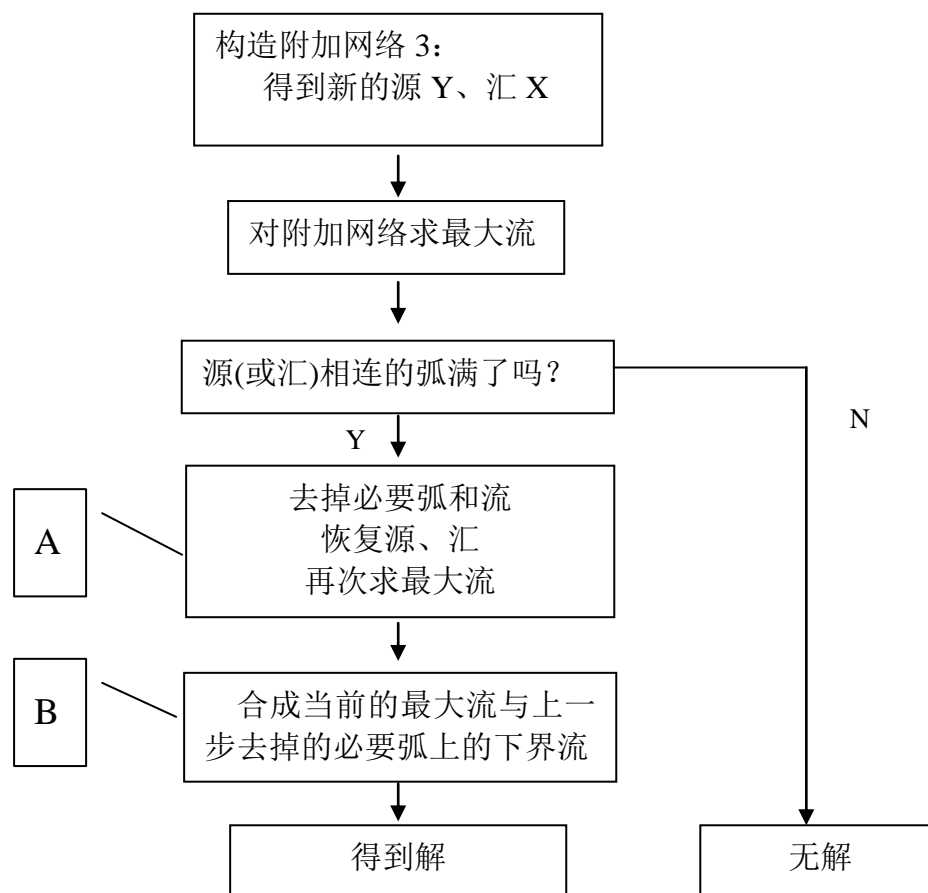


图 7.14

当然，本例不可能再增大流了，例子仅对解决**问题 7.2** 的图示说明。另外，这种情况下的增广路径改进，不会影响**必要弧**---即：不破坏下界。

八、小结二

求容量有上下界最大流的基本思想：



竞赛中的实用算法：

周源同学在 IOI2004 国家集训队作业《一种简易的方法求解流量有上下界的网络中网络流问题》，是对上面方法的简单化。

简化的模型没有了上面的步骤 A 和 B，避免了二次改造网络，使编程容易很多。当然，代价是要多次求最大流。

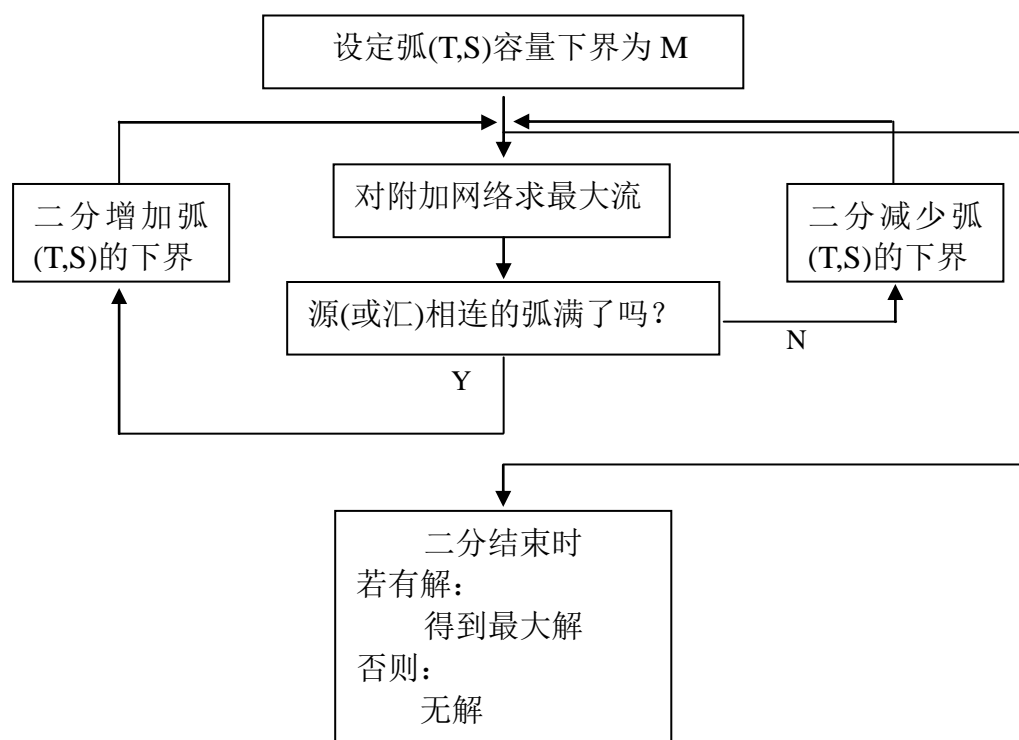
这个方法中用二分法，确定容量下界的最大值，求(T,S)可能的最大流。

另外，文中还提到：

“在一个容量有上下界的流网络 G 中，求源点 s 到汇点 t 的一个可行的最小流。... 类似，只是在加入弧 (t, s) 后二分 (t, s) 的容量上界 $C(t, s)$ 即可。”

可见这个简化的方法在竞赛中用途很广。

简化方法的基本思想：



至此，我们已经清楚地阐述了一个关于网络流方面的模型：对于“必要的流量”，用**必要弧**分离出来，用**附加网络 3**求得其上的一个可行流方案。当然，如果无解，也可以判断出。

上面两节，我们对流的**分离**和流网络的**等价变换**做了进一步发挥。与之前所用方法技巧的比较：

	附加网络 1	附加网络 3
退流	全部可退	只能部分退，下界要保证
分离	流： 后向弧	容量： 必要弧---下界容量
模型转换	直接转换	有源汇→无源汇→有源汇

刘汝佳的名著《算法艺术与信息学竞赛》P324-P325 也有这样几句话：

“本题和流有关，但是看起来不是去求什么最大流，而且连源和汇都没有，.....”

“.... 本题中处理下界的方法可以推广到任意网络。”

显然，**无源汇**、**必要弧**这些概念和方法早为专家们所用。本文只是做些整理和解释工作。

九、例题分析

本节通过几个经典题目的模型构造，进一步体会上述的网络模型。

[例 9.1]经典问题

[例 9.2] 经典问题

[讨论题] 经典问题

参考文献：

吴文虎 王建德，《图论的算法与程序设计》

吴文虎 王建德，《实用算法的分析与程序设计》

Thomas H.Cormen 等，《算法导论》

刘汝佳 黄亮，《算法艺术与信息学竞赛》

朱全民，《奥赛兵法》

周 源，论文《一种简易的方法求解流量有上下界的网络中网络流问题》

李伟星，论文《最大流》

江 鹏，论文《从一道题目的解法试谈网络流的构造与算法》