

# 从一道题目的解法试谈网络流的构造与算法

福建师大附中 江鹏

## 1. 引论

### A. 对网络流算法的认识

网络流算法是一种高效实用的算法，相对于其它图论算法来说，模型更加复杂，编程复杂度也更高，但是它综合了图论中的其它一些算法（如最短路径），因而适用范围也更广，经常能够很好地解决一些搜索与动态规划无法解决的，看似 NP 的问题。

### B. 具体问题的应用

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。

## 2. 例题分析

### 【问题 1】项目发展规划 (Develop)

Microsoft® 公司准备制定一份未来的发展规划。公司各部门提出的发展项目汇总成了一张规划表，该表包含了许多项目。对于每个项目，规划表中都给出了它所需的投资或预计的盈利。由于某些项目的实施必须依赖于其它项目的开发成果，所以如果要实施这个项目的話，它所依赖的项目也是必不可少的。现在请你担任 Microsoft® 公司的总裁，从这些项目中挑选出一部分，使你的公司获得最大的净利润。

- 输入

输入文件包括项目的数量  $N$ ，每个项目的预算  $C_i$  和它所依赖的项目集合  $P_i$ 。格式如下：

第 1 行是  $N$ ；

接下来的第  $i$  行每行表示第  $i$  个项目的信息。每行的第一个数是  $C_i$ ，正数表示盈利，负数表示投资。剩下的数是项目  $i$  所依赖的项目的编号。

每行相邻的两个数之间用一个或多个空格隔开。

- 输出

第 1 行是公司的最大净利润。接着是获得最大净利润的项目选择方案。若有多个方案，则输出挑选项目最少的一个方案。每行一个数，表示选择的项目的编号，所有项目按从小到大的顺序输出。

- 数据限制

$0 \leq N \leq 1000$

$-1000000 \leq C_i \leq 1000000$

- 输入输出范例

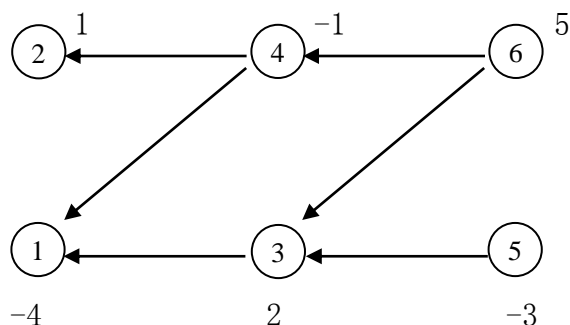
Sample Input	Sample Output
6	3
-4	1
1	2
2 2	3
-1 1 2	4
-3 3	6
5 3 4	

### 【分析解答】

#### 1. 抽象原题（图论模型）

给定包含  $N$  个顶点的有向图  $G = (V, E)$ ，每个顶点代表一个项目，顶点有一权值  $C_i$  表示项目的预算。用有向边来表示项目间的依赖关系，从  $u$  指向  $v$  的有向边表示项目  $u$  依赖于项目  $v$ 。

问题：求顶点集的一个子集  $V'$ ，满足对任意有向边  $\langle u, v \rangle \in E$ ，若  $u \in V'$ ，则  $v \in V'$ ，使得  $V'$  中所有顶点的权值之和最大。



#### 2. 搜索

枚举  $V$  的所有符合条件的子集，时间复杂度  $O(2^n)$ ，指数级。无论如何剪枝优化，也摆脱不了非多项式。

#### 3. 动态规划

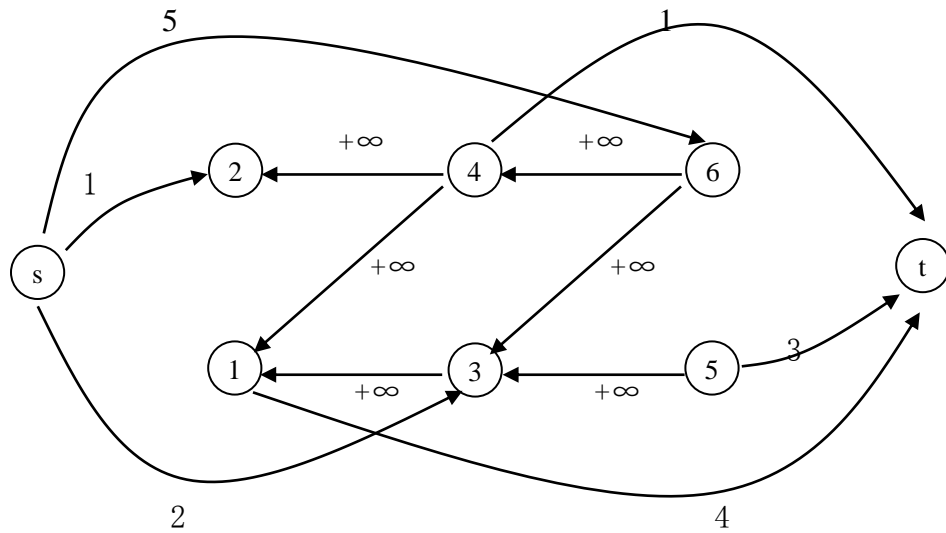
本题的结构是有向无环图，而非树形结构，不适合动态规划。如果一定要做，实质类似于搜索，由于状态数量众多，仍是指数级的时间复杂度。

#### 4. 网络流

流网络的构造方法：

建立  $N$  顶点代表  $N$  个项目，另外增加源  $s$  与汇  $t$ 。若项目  $i$  必须依赖于项目  $j$ ，则从顶点  $i$  向顶点  $j$  引一条容量为无穷大的弧。对于每个项目  $i$ ，若它的预算  $C$  为正（盈利），则从源  $s$  向顶点  $i$  引一条容量为  $C$  的边；若它的预算  $C$  为负（投资），则从顶点  $i$  向汇  $t$  引一条容量为  $-C$  的边。

求这个网络的最小割  $(S, T)$ ，设其容量  $C(S, T) = F$ 。设  $R$  为所有盈利项目的预算之和（净利润上界），那么  $R - F$  就是最大净利润； $S$  中的顶点就表示最优方案所选择的项目。



最小割:  $S = \{s, 1, 2, 3, 4, 6\}$ ;  $T = \{5, t\}$   
 $C(S, T) = 5$     净利润  $R - C(S, T) = 8 - 5 = 3$

证明算法的正确性:

- 建立项目选择方案与流网络的割  $(S, T)$  的一一对应关系:

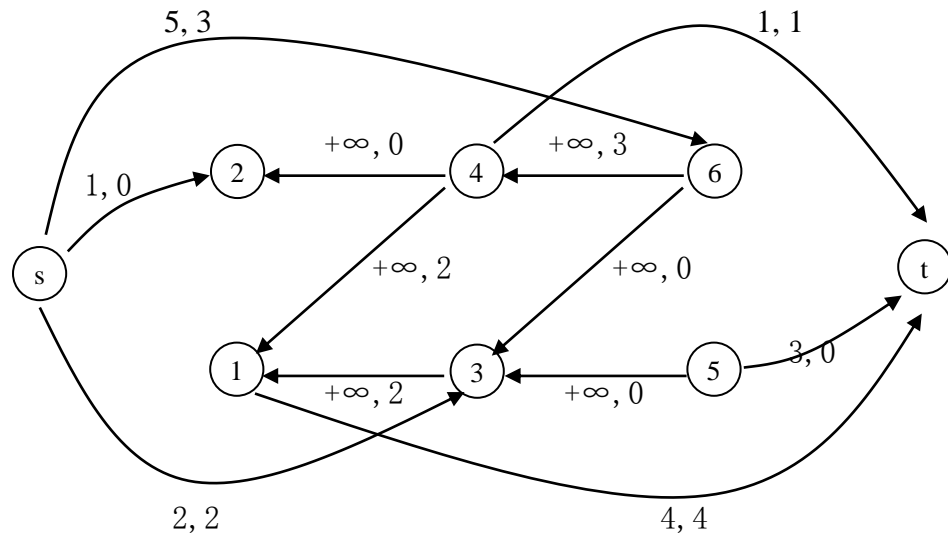
任意一个项目选择方案都可以对应网络中的一个割  $(S, T)$ ,  $S = \{s\} + \{\text{所有选择的项目}\}$ ,  $T = V - S$ 。  
 对于任意一个不满足依赖关系的项目选择方案, 其对应的割有以下特点:

存在一条容量为  $+\infty$  弧  $\langle u, v \rangle$ ,  $u$  属于  $S$  而  $v$  属于  $T$ 。这时割的容量是无穷大, 显然不可能是网络的最小割。

- 对于任意一个割  $(S, T)$ , 如果其对应一个符合条件的方案, 它的净利润是  $R - C(S, T)$ 。导致实际净利润小于上届  $R$  的原因有:
  1. 未选取盈利项目  $i$ , 即顶点  $i$  包含在  $T$  中, 那么存在一条从源  $s$  至顶点  $i$  的容量为  $C_i$  的弧
  2. 选取投资项目  $i$ , 即顶点  $i$  包含在  $S$  中, 那么存在一条从顶点  $i$  至汇的容量为  $-C_i$  的弧 $C(S, T)$  就是上述两种弧的容量之和。

综上所述, 割的容量越小, 方案的净利润就越大。

- 最小割的求法:  
 根据最大流最小割定理, 网络的最小割可以通过最大流的方法求得。



本题解题的关键在于流网络数学模型的建立。本题建模的独到之处在于：以前的网络流问题通常使用流量表示解答方案，而本题使用割表示解答方案，并充分利用了割的性质，流只是求得最小割的手段。这为我们开辟了一条构造网络流解决问题的新思路。

初看这个问题，要把它和网络流联系起来，有相当的难度。必须熟练地掌握流网络的各种性质，经过反复的类比尝试，才能发现它们之间的共性。

### 【联想思考】

作为本题的一个衍生，给每个项目估计一个完成时间，并假设公司同时只能进行一个项目。现在的问题是：如何选择一些能在给定时间内完成的项目，使得公司得到最大收益。这个问题我至今还没有找到有效算法，希望有兴趣的同学来共同研究。

## 3. 编程技巧

◆ 数据结构：邻接表

◆ 直接表示原问题

优点：节省空间

缺点：编程复杂度大，不具有通用性