第五章 组合优化问题的原始－对偶算法-4

5.3 最小费用流的原始－对偶算法

一、最小费用流问题

最短路问题：容量限制平凡，费用非平凡，组合化费用；
最大流问题：费用平凡，容量限制非平凡，组合化容量限制；
最小费用流问题：容量限制平凡，费用非平凡，
作为对偶问题，组合化容量限制——圈算法；
作为原始问题，组合化费用——迭加算法。

定义 5.3 设 $N=(s,t,V,E,b)$ 是有向图 $G=(V,E)$ 上的一个流网络，对每一条弧 $(i,j)\in E$ 有一个权 $c_{ij}\in R^+$，$v_0\in R^+$ 为流的值。最小费用流问题是求一个值为 v_0 的可行流，使得其费用最小。

即：

b_{ij}：弧 $(i,j)\in E$ 上流量的上界；

c_{ij}：弧 $(i,j)\in E$ 上流量的权值；

v_0：指定的流值

问题：求一个可行流 $f(i,j)$，它的流值等于 v_0，并且使得总费用最小。这个问题的线性规划模型为：

$$
\begin{aligned}
&\min c^T f\\
&Af=-v_0 d\\
&f\le b\\
&f\ge 0
\end{aligned}
$$

其中 A 是点－弧关联矩阵，而

$$
d_i=\begin{cases}-1, & i=s\\ +1, & i=t\\ 0, & \text{其余}\end{cases}
$$

二、组合化容量限制——最小费用流问题的圈算法

将最小费用流问题的线性规划模型写成标准形式的原始问题的对偶问题：

$$
\begin{aligned}
&\max -c^T f\\
&Af\le -v_0 d\\
&f\le b\\
&-f\le 0
\end{aligned}
\qquad \text{(D)}
$$

可以用最大流算法求得(D)的一个值为 v_0 的可行流。

原始－对偶算法的过程：

(LP)

$$\begin{aligned} &\max -c^T f \\ &Af \le -v_0 d \\ &f \le b \\ &-f \le 0 \end{aligned} \qquad \text{(D)}$$

f 是(D)的一个可行解

允许列集合 $J=\{Af=-v_0 d$ 的行，

即所有行，和 $f=b$ 的弧

及 $f=0$ 的弧$\}$

(RP) $\Longrightarrow$

$$\begin{aligned} &\max c^T(-f) \\ &Af=0,\ \text{对一切行} \\ &f \le 0,\ \text{对D中}f=b\text{的弧} \\ &-f \le 0,\ \text{对D中}f=0\text{的弧} \\ &\text{-}f \le 1,\ \text{一切弧} \end{aligned} \qquad \text{(DRP)}$$

$f := f+\theta\overline{f}$

求解(DRP)的解释：

用 $Af=0$ 代替了 $Af \le 0$，故又回到了流在每一个节点守恒的情形。满足 $Af=0$ 的可行流有特殊意义：

定义 5.4 满足 $Af=0$ 的可行流 f 称为循环流，它的费用为 $c^T f$。

所以，(DRP)的最优解是一个特殊类型的循环流：

(1) 空弧上的流非负（空弧上可能要加载一些流）

(2) 饱和弧上的流非正（饱和弧上可能要撤去一些流，但不能再加载任何流）

(3) 任意弧上的流大于或等于－1（撤去的流最多一个单位）

所以：求(DRP)的最优解实际上就是寻找满足上述要求的特殊的循环流。

下面将求 $\overline{f}$ 和求 θ_1 联合在一起考虑，即直接求一个费用减少最多的循环流 $\theta_1\overline{f}$，然后加到 f，得到一个新的可行流。

按上述(1)-(3)条件，我们可以定义一个新的赋权容量网络。

定义 5.5 给定一个赋权的流网络 $N=(s,t,V,E,b)$ 上的一个可行流 f，定义一

个调整权的流网络 $N'(f)$ 如下：

(i) N' 与 N 有相同的节点集合；

(ii) 对 N 中的每一条弧 (i,j)，设它的流为 v，容量为 d，费用为 c，则 N' 中有两条弧

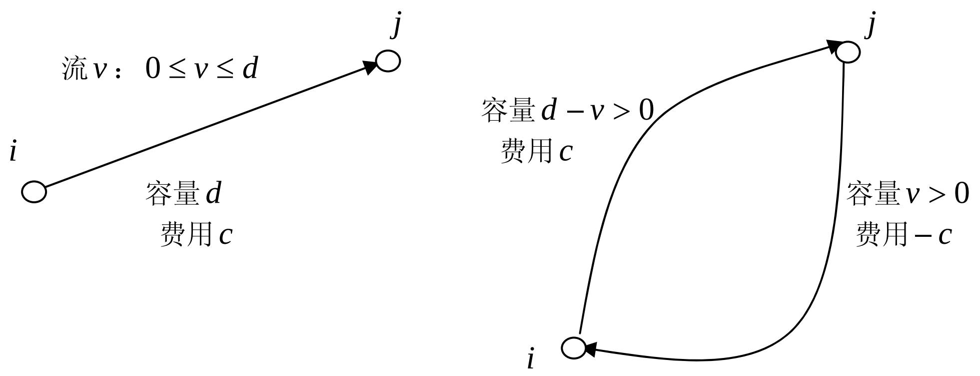
(i,j) 和 (j,i) 与之对应：

(i,j) 的容量为 $d-v\geq 0$，费用为 c；

(j,i) 的容量为 $v\geq 0$，费用为 $-c$；

(iii) 去掉所有容量为 0 的弧。

N 中的一条弧 (i,j) $\longrightarrow$ 对应 N' 中有两条弧 (i,j) 和 (j,i)



说明：在调整网络 $N'(f)$ 上，弧 (i,j) 上流的调整范围在 $-v$ 到 $d-v$ 之间，即：

$i\rightarrow j$ 可以至多增加 $d-v$ 个单位流，增加每个单位流，费用增加为 c

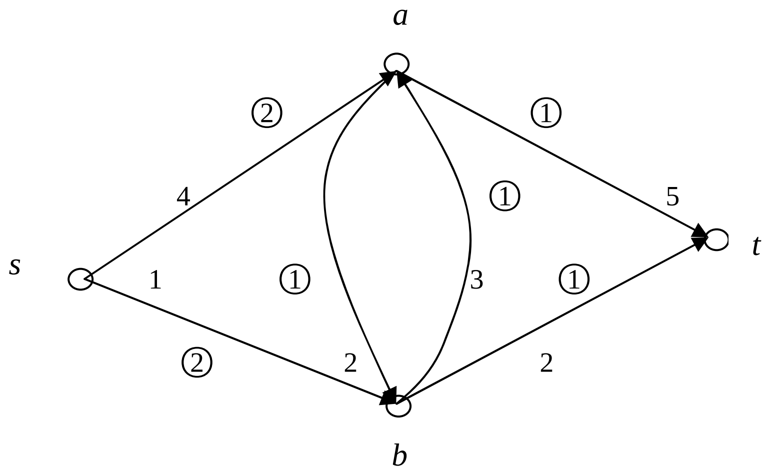
$i\rightarrow j$ 可以至多减少 v 个单位流，减少每个单位流，费用减少为 $-c$

显然，

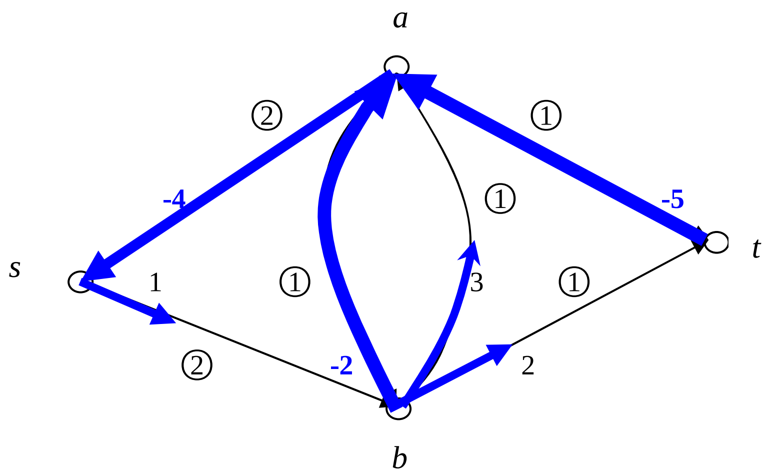
(i) 在调整网络 $N'(f)$ 中自 s 到 t 的一条路，若其弧上的费用和为 x，那么它确定了 N 中的一条增广路，并且在 N 中沿这条增广路增加一个单位的 $s-t$ 流，那么费用的变化为 x；

(ii) $N'(f)$ 中费用为 x 的循环流 $\overline{f}$，确定 N 中的一个新的值为 v_0 可行流 $f+\overline{f}$，其费用变化为 x。

例：4 个节点 6 条弧的赋权网络，求流值为 2 的最小费用流。圈里数字为容量，圈外为费用。



第一个可行流：$f(s,a)=2$，$f(a,t)=1$，$f(a,b)=f(b,t)=1$，其余弧上为 0，$c^T f=17$。$N'(f)$ 为：



$N'(f)$ 中有一个循环流 $\overline{f}(s,b)=\overline{f}(b,a)=\overline{f}(a,s)=1$，费用为 $c^T\overline{f}=-5$，则可以得到新的可行流 $f:=f+\overline{f}$，其中 $f(s,a)=2-1=1$，$f(s,b)=1$，$f(a,t)=1$，$f(b,t)=1$，$f(a,b)=1-1=0$，其余为 0。新的可行流的费用为 $c^T f=17-5=12$。

所以，$N'(f)$ 中费用为-5 的循环流 $\overline{f}$，确定 N 中的一个新的值为 2 可行流 $f+\overline{f}$，其费用变化为-5。

最优性条件：

定理 5.7 赋权的流网络 $N=(s,t,V,E,b)$ 中的流值为 v_0 的 $s-t$ 流 f 是最小费用的 $\Leftrightarrow$ $N'(f)$ 中没有负费用的有向圈。

证明：由原始－对偶算法，流值为v_0的$s-t$流f是最优的$\Leftrightarrow$(DRP)的最优值为 $0\Leftrightarrow$ $N'(f)$中没有负费用的循环流（若$N'(f)$中存在负费用的循环流，则(DRP)的最优值为-1）$\Leftrightarrow$ $N'(f)$中没有负费用的有向圈。

证毕。

赋权流网络$N=(s,t,V,E,b)$中的流值为v_0的最小费用$s-t$流的求解算法 1：

第 1 步：用最大流算法求得(D)的一个值为v_0的$s-t$可行流f；

第 2 步：构造调整权的流网络$N'(f)=(s,t,V,E_{N'(f)},b_{N'(f)})$，

$N'(f)$中弧$(i,j)\in E_{N'(f)}$上的容量为$b_{N'(f)}(i,j)$，费用为$c_{N'(f)}(i,j)$；

第 3 步：用 Floyd-Warshall 算法或推广的 Dijkstra 算法搜索$N'(f)$中的负费用有向圈；

第 4 步：若$N'(f)$中不存在负费用有向圈，则当前的可行流f即为流值为v_0的最小费用$s-t$流，算法终止；

第 5 步：若得到$N'(f)$中的负费用有向圈C，则得到(DRP)的最有解$\overline{f}$，在C上寻找最大的循环流$\theta_1\overline{f}$，其中$\theta_1=\min\{b_{N'(f)}(i,j):(i,j)\in C\}$；

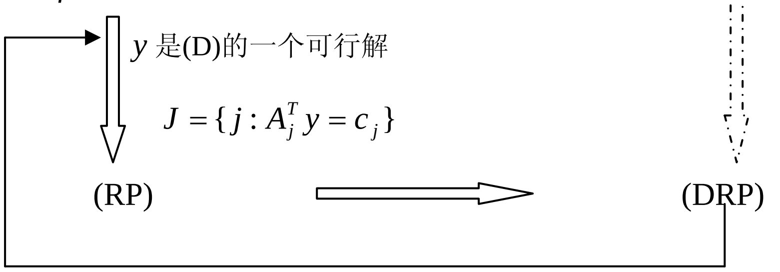
第 6 步：$f:=f+\theta_1\overline{f}$，转第 2 步。

——以上算法实质上就是 Klein 圈算法，又称为问题可行算法。

三、组合化费用——最小费用流问题的迭加算法

把最小费用流问题规划模型作为原始问题，组合化费用：

$$\begin{aligned}&\min c^T f\\&Af=-v_0 d\\&f\le b\\&f\ge 0\end{aligned}\quad\text{(LP)}\qquad\qquad\text{(D)}$$



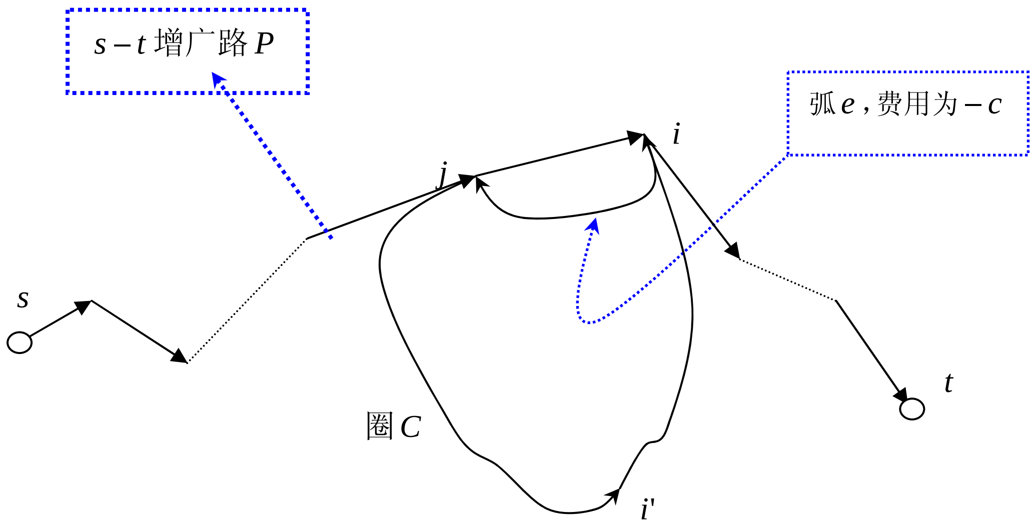
算法将始终保持对偶规划(D)的可行性和原始规划的最优性，逐步改变原始

规划的不可行性，即最后使得 $Af=-v_0 d$， $f\le b, f\ge 0$ 都成立。

算法要首先求对偶规划，然后求允许列集合，问题变得很复杂。根据组合化费用的原始一对偶算法的上述思想，直接从原始规划着手，始终保持 $s-t$ 流 f 的最优性（即是最小费用流）和保持除流值小于或等于 v_0 约束不成立外的其它约束成立（即流量守恒， $f\le b, f\ge 0$ 都保持成立），然后逐步增加流的值，直至流值为 v_0 为止，从而不可行性消失，得到流值为 v_0 的最小费用流。这样就得到流值增加的流的一个序列，使得序列中每一个流都是最小费用的。

定理 5.8 设 f_1 是流网络 $N=(s,t,V,E,b)$ 中的流值为 v 的最小费用流，f_2 是调整权的流网络 $N'(f_1)$ 中沿最小费用的 $s-t$ 增广路 P 的单位流，那么 f_1+f_2 是 N 中值为 $v+1$ 的最小费用流。

证明：用反证法。显然 f_1+f_2 是 N 中值为 $v+1$ 的可行流。假设 f_1+f_2 不是最小费用流，那么根据定理 4.5， $N'(f_1+f_2)$ 中必有负费用的有向圈 C，但是由于 f_1 是流网络 $N=(s,t,V,E,b)$ 中的最小费用流，所以 $N'(f_1)$ 没有负费用的有向圈。因此，负费用的有向圈 C 是在流的值从 v 增加到 $v+1$ 时产生的。所以 C 中必有一条弧 $e=(i,j)$，设其费用为 $-c$，使得它对应于路 P 中的弧 (j,i)：



设圈 C 的费用为 $\cos t(C)<0$，则 $\cos t(C-\{e\})=\cos t(C)+c$。构造另外一条 $s-t$ 增广路 $\overline{P}=P-\{(j,i)\}+C-\{e\}=\{s\to j\to i'\to i\to t\}$，则 $s-t$ 增广路 $\overline{P}$ 的费用为：

$$\cos t(\overline{P})-\cos(P)=\cos t(C-\{e\})-\cos t\{(j,i)\}=\cos t(C)+c-c=\cos t(C)<0$$

所以，P 不是 $N'(f_1)$ 中沿最小费用的 $s-t$ 增广路，矛盾。所以 f_1+f_2 是 N 中值为 $v+1$ 的最小费用流。

证毕。

所以，我们可以沿 $N'(f_1)$ 中最小费用的增广路（$N'(f_1)$ 中 $s-t$ 最短路）逐步增加流的值，直到流值达到 v_0 为止。

赋权流网络 $N=(s,t,V,E,b)$ 中的流值为 v_0 的最小费用 $s-t$ 流的求解算法 2：

初始流 $v=0, f=0$ 。

第 1 步：构造调整权的流网络 $N'(f)=(s,t,V,E_{N'(f)},b_{N'(f)})$ ，

$N'(f)$ 中弧 $(i,j)\in E_{N'(f)}$ 上的容量为 $b_{N'(f)}(i,j)$ ，费用为 $c_{N'(f)}(i,j)$ ；

第 2 步：求 $N'(f_1)$ 中的 $s-t$ 最短路 P ；

第 3 步：令 $\theta_1=\min\limits_{(i,j)\in P} b_{N'(f)}(i,j)$ ，若 $v_0\le v+\theta_1$，则令

$$f:=\begin{cases} f(i,j)+(v_0-v), & \text{if } (i,j)\in P \\ f(i,j), & \text{oterwise}\end{cases}$$

算法终止，得到流值为 v_0 的最小费用流 f 。否则令

$$f:=\begin{cases} f(i,j)+\theta_1, & \text{if } (i,j)\in P \\ f(i,j), & \text{oterwise}\end{cases}$$

转第 1 步。

注：

(1) $N'(f)$ 中的弧可能存在负费用，故求 $N'(f_1)$ 中的 $s-t$ 最短路 P 的算法必须能适应这种情况。但是，由于在任意阶段，我们总是最优流，所以 $N'(f)$ 中决不可能存在负费用的有向圈。因此，可以用最短路算法求最小费用的增广路 P 。

(2) 与圈算法相比，上述迭加算法在算法结束前，不可能得到值为 v_0 的可行流。因此，称之为问题－不可行算法。