《算法设计与分析》

12-线性规划介绍 (An Intoduction to Linear Programming)

杨启哲

上海师范大学信机学院计算机系

2024 年 12 月 17 日

主要内容

- 什么是线性规划
- 线性规划的求解-单纯形法 (Simplex Method)
- 从线性规划的角度理解最大流最小割定理

什么是线性规划

什么是线性规划

线性规划描述了一类特殊的优化问题，这些问题的共同特征是玉树条件和优化准则都可以转化成线性函数。

在**线性规划**问题中，通常会给定一个变量集合，任务是为其赋予实际的值，使其满足：

1. 一组关于这些变量的线性方程或不等式。
2. 是给定的目标线性函数数值最大或最小。

$$
\begin{aligned}
\max \quad & x_1 + 6x_2 \\
\text{s.t.} \quad & x_1 \leqslant 200 \\
& x_2 \leqslant 300 \\
& x_1 + x_2 \leqslant 400 \\
& x_1, x_2 \geqslant 0
\end{aligned}
$$

一些具体的例子 (I)

利润最大化

某家巧克力作坊生产两种产品，一种是称为 Pyramide 的旗舰三角巧克力；一种是称为 Pyramide Nuit(简称 Nuit) 的奢华高级巧克力。假设工厂每天可以生产 x_1 盒 Pyramide 和 x_2 盒 Nuit，前者每盒利润为 1 元，后者每盒利润为 6 元。我们需要确定工厂每天生产的巧克力数量，使得利润最大化。当然谈这有一些额外的限制，如：

1. 市场每天仅需 200 盒 Pyramide 和 300 盒 Nuit。
2. 工厂每天最多生产 400 盒巧克力。

$$
\begin{aligned}
\max \quad & x_1 + 6x_2 \\
\text{s.t.} \quad & x_1 \leqslant 200 \\
& x_2 \leqslant 300 \\
& x_1 + x_2 \leqslant 400 \\
& x_1, x_2 \geqslant 0
\end{aligned}
$$

一些具体的例子 (II)

上海师范大学 Shanghai Normal University

利润最大化 (II)

收到市场的积极影响，工厂开发了一种新产品 Pyramide Luxe(简称 Luxe)。每盒 Luxe 的利润为 13 元。用 x_3 表示生产的 Luxe 的盒数，在满足之前的限制下还需要满足如下限制：

1. 由于 Nuit 和 Luxe 需要用相同的包装设备，而且 Luxe 需要使用 3 次，但该包装设备一天只能实现 600 次操作。

$$
\begin{aligned}
\max \quad & x_1 + 6x_2 + 13x_3x_3 \\
\text{s.t.} \quad & x_1 \leqslant 200 \\
& x_2 \leqslant 300 \\
& x_1 + x_2 + x_3 \leqslant 400 \\
& x_2 + 3x_3 \leqslant 600 \\
& x_1, x_2, x_3 \geqslant 0
\end{aligned}
$$

一些具体的例子 (III)

上海师范大学 Shanghai Normal University

生产计划

某公司生产手工编织的地毯，这是一种季节性极强的产品。分析师已就下年度每个月的需求量进行了估算，依次为 $d_1, \dots, d_{12}$。这些数值差异非常巨大，从 440 至 920 不等。

公司的基本情况是，一共有 30 名工人，每人每月可以制造 20 块地毯，月工资 2000 元。公司最初没有任何产品库存。

为了解决起伏不定的需求，一般有 3 种途径：

1. **加班**: 加班的成本是昂贵的，每小时需要多付工人 80% 的工资，并且，加班时间不能超过正常工时的 30%。
2. **聘用和解雇**：对应的一次性费用分别为每人次 300 元和 400 元。
3. **存储**：每块地毯每月的存储费用为 8 元。目前没有任何库存，而到年底时也应该如此。

应该如何安排生产计划，使得总成本最小？

生产计划问题的刻画 (I)

我们先定义一些变量:

- w_i: 第 i 个月的工人数，初始 $w_0 = 30$。
- x_i: 第 i 个月正常工时生产的地毯数。
- o_i: 第 i 个月加班工时生产的地毯数。
- h_i，f_i: 第 i 个月新聘用和解雇的工人数。
- s_i: 第 i 个月底存储的地毯数，初始 $s_0 = 0$。

可以看到，算是两个初始值 w_0, s_0，一共有74 **个变量**。

生产计划问题的刻画 (II)

我们考虑描述约束条件和成本最小化:

---

1. 所有的变量应该是非负的:

$$w_i, x_i, o_i, h_i, f_i, s_i \geqslant 0, \quad \forall i \in \{1, \ldots, 12\}$$

2. 每个月生产的地毯数目等于正常工时和加班时间生产的总量:

$$x_i = 20w_i + o_i, \quad \forall i \in \{1, \ldots, 12\}$$

3. 每个月初工人数量会发生变化:

$$w_i = w_{i-1} + h_i - f_i, \quad \forall i \in \{1, \ldots, 12\}$$

4. 每个月底库存的地毯数等于月初的库存数加上本月生产数减去本月销售数量:

$$s_i = s_{i-1} + x_i - d_i, \quad \forall i \in \{1, \ldots, 12\}$$

我们考虑描述约束条件和成本最小化:

5. 加班限制:

$$o_i \leqslant 0.3 \times 20w_i = 6w_i, \quad \forall i \in \{1, \dots, 12\}$$

6. 成本最小化:

$$\min \quad 2000\sum_{i=1}^{12} w_i + 300\sum_{i=1}^{12} h_i + 400\sum_{i=1}^{12} f_i + 8\sum_{i=1}^{12} s_i + 480\sum_{i=1}^{12} o_i$$

生产问题的线性规划表示全貌

$$
\begin{aligned}
\min \quad & 2000\sum_{i=1}^{12} w_i + 300\sum_{i=1}^{12} h_i + 400\sum_{i=1}^{12} f_i + 8\sum_{i=1}^{12} s_i + 480\sum_{i=1}^{12} o_i \\
\text{s.t.} \quad & x_i = 20w_i + o_i, \quad \forall i \in \{1,\dots,12\} \\
& w_i = w_{i-1} + h_i - f_i, \quad \forall i \in \{1,\dots,12\} \\
& s_i = s_{i-1} + x_i - d_i, \quad \forall i \in \{1,\dots,12\} \\
& o_i \leqslant 6w_i, \quad \forall i \in \{1,\dots,12\} \\
& w_i, x_i, o_i, h_i, f_i, s_i \geqslant 0, \quad \forall i \in \{1,\dots,12\}
\end{aligned}
$$

- 利用一些现成的线性规划工具，我们可以在几乎不到 1 秒的时间就解决这个问题。
- 很多时候最优解可能包含一些小数，这时候常见的策略是直接进行取整。大部分时候该策略是很有效的，但是总有些 LP 问题，对小数部分的取舍需要特别小心。
- 事实上，如果要求是得到最优的整数值解，则该问题是**整数线性规划**问题。即使我们把变量限定在$\{0,1\}$上，对应的判定问题也是**NP-完全**的。

线性规划的标准形式 (I)

通过上述例子可以看到线性规划的形式是很自由的:

1. 可以是最大化问题也可以是最小化问题。
2. 约束条件可以是等式也可以是不等式。
3. 变量通常是非负的，但是也可以是具有任意符号的。

事实上，通过一些变换，我们总可以将其进行互相转化。

线性规划的标准形式 (II)

上海师范大学 Shanghai Normal University

1. 可以是最大化问题也可以是最小化问题:
   目标函数两边同时乘以 -1，即可将最大化问题转化为最小化问题，反之亦然。

$$\max \quad c_1x_1 + c_2x_2 \quad \Leftrightarrow \quad \min \quad -c_1x_1 - c_2x_2$$

2. 约束条件可以是等式也可以是不等式。
   - 引入新的参数 s(称为松弛变量)，将不等式约束转化为等式约束:

$$\sum a_ix_i \leqslant b \quad \Leftrightarrow \quad \begin{cases} \sum a_ix_i + s = b \\ s \geqslant 0 \end{cases}$$

   - 对于等式约束 $ax = b$，写为一对不等式 $ax \leqslant b$ 和 $ax \geqslant b$ 即可。
3. 变量通常是非负的，但是也可以是具有任意符号的。
   对于不确定符号的变量 x，引入两个变量 x^+ 和 x^-，令 $x = x^+ - x^-$，同时 $x^+, x^- \geqslant 0$:

$$x \quad \Leftrightarrow \quad \begin{cases} x = x^+ - x^- \\ x^+, x^- \geqslant 0 \end{cases}$$

我们可以利用矩阵形式来描述这些限制:

$$\begin{aligned} x_1 &\leqslant 200 \\ x_2 &\leqslant 300 \\ x_1 + x_2 &\leqslant 400 \end{aligned} \quad \Longrightarrow \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leqslant \begin{bmatrix} 200 \\ 300 \\ 400 \end{bmatrix} \quad (A\boldsymbol{x} \leqslant \mathbf{b})$$

从而我们可以将一个线性规划问题用矩阵表示为:

$$\begin{aligned} \max \quad & \mathbf{c}^{\mathsf{T}}\boldsymbol{x} \\ \text{s.t.} \quad & A\boldsymbol{x} \leqslant \mathbf{b} \\ & \boldsymbol{x} \geqslant 0 \end{aligned}$$

我们称这样的形式为标准形式。

线性规划的求解-单纯形法 (Simplex Method)

两维的例子 (I)

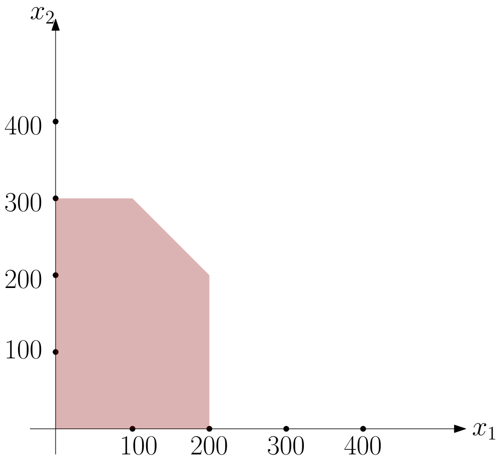
上海师范大学 Shanghai Normal University

回顾我们第一个例子:

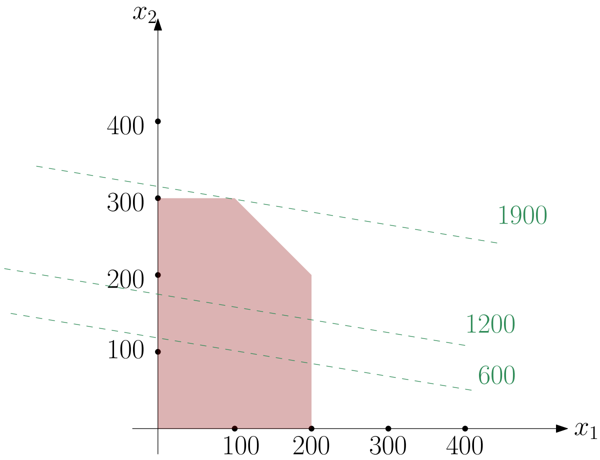
$$\begin{aligned} \max \quad & x_1 + 6x_2 \\ \text{s.t.} \quad & x_1 \leqslant 200 \\ & x_2 \leqslant 300 \\ & x_1 + x_2 \leqslant 400 \\ & x_1, x_2 \geqslant 0 \end{aligned}$$

每个约束对应了二维平面上的一个半平面，因此所有其约束构成了二维平面上的一个区域。

两维的例子 (II)



两维的例子 (III)



最优解的寻找

1. 我们需要寻找到多边形的某点，使得其目标函数值最大。
2. 对应的目标函数可以表示成形如 $x_1 + 6x_2 = c$ 的直线上，在该直线的每个点上对应的利润都是 c。
3. 我们可以通过不断的向上移动这条直线，直到” 再往上”就没有办法找到一个点使得利润更大的地步:
   - 如果该线的斜率和某个约束的斜率相同，那么可能会找到一条边和该直线重合，则该问题下最优解有无数个答案。
   - 如果该线的斜率和某个约束的斜率不同，那么最优解一定在某个顶点上。

最优解也可能不存在:

1. 约束条件过紧，导致无法找到一个点同时满足所有的约束。
2. 约束条件过松，导致可行区域无界，从而使得目标函数可能达到任意大 (小) 的值。

最优解的寻找 (II)

所谓单纯形法，就是从某个顶点出发，不断地寻找目标函数值更优地邻居顶点，直到再也找不到更高的邻居节点，此时该点就是最优解。

算法：单纯形法

输入： 一组线性不等式和线性目标函数 f

输出： 最优可行解

1: 令 v 是可行区域的一个顶点。
2: **while** 存在一个邻居 u 使得 $f(u) > f(v)$ **do**
3: 　　令 $v = u$
4: **return** v

问题是，什么是**顶点**？什么是**邻居**？

n 维空间的顶点和邻居

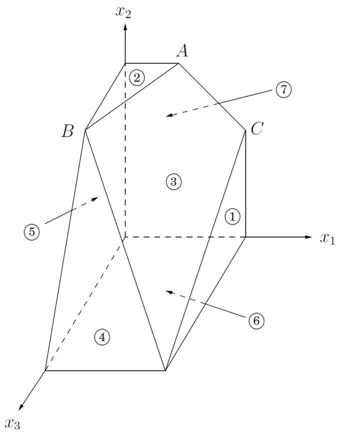
上海师范大学 Shanghai Normal University

一个顶点，实际上是满足某些约束条件对应的等式的唯一解。

---

1. **顶点**：由 n 个不等式所定义的一个点。
2. **邻居**：若两个顶点对应的所有不等式中有 $n-1$ 个相同，则其互为邻居。

一个三维空间的例子



$$
\begin{aligned}
\max \quad & x_1 + 6x_2 + 13x_3 & \\
& x_1 \le 200 & ① \\
& x_2 \le 300 & ② \\
& x_1 + x_2 + x_3 \le 400 & ③ \\
& x_2 + 3x_3 \le 600 & ④ \\
& x_1 \ge 0 & ⑤ \\
& x_2 \ge 0 & ⑥ \\
& x_3 \ge 0 & ⑦
\end{aligned}
$$

事实上，顶点 B 可以由不同的不等式定义，比如 $2, 3, 4$ 和 $2, 4, 5$，这类顶点被称为**退化顶点**。我们先假设不存在退化顶点。

算法的具体描述 (I)-原点的情况

上海师范大学 Shanghai Normal University

考察一个一般形式的线性规划问题:

$$\begin{aligned} \max \quad & \mathbf{c}^{\mathsf{T}}\mathbf{x} \\ \text{s.t.} \quad & A\mathbf{x} \leqslant \mathbf{b} \\ & \mathbf{x} \geqslant 0 \end{aligned}$$

令 $\mathbf{x} = (x_1, \ldots, x_n)$ 是所有的变量，并且令 $\mathbf{b} \geqslant 0$，则原点是一个可行解，并且也是一个顶点 (为什么？)。

1. 原点最优当且仅当对所有的 c_i，有 $c_i \leqslant 0$。
2. 如果存在一个 $c_i > 0$，那我们通过增大 x_i 的值，直至某个新的约束条件被满足，这样我们就找到了一个更优的邻居。

算法的具体描述 (II)-非原点的情况

当前顶点不是原点，该怎么处理？**变成原点!**

通过修改 LP 的形式，我们可以将任意的顶点 $\mathbf{u}$ 转化为原点而不改变问题的解：

1. 令包围 $\mathbf{u}$ 的 n 个超平面对应的不等式为 $\mathbf{a}_i\mathbf{x} \leqslant b_i$。
2. 我们记一个点 $\mathbf{x}$ 到对应的超平面的距离为

$$y_i = b_i - \mathbf{a}_i\mathbf{x}$$

3. 通过将其改写成由 $\mathbf{y} = (y_1, \ldots, y_n)$ 表示的坐标，我们得到了一个新形式的 LP，该形式并没有改变问题的解，而顶点 $\mathbf{u}$ 变成了原点：
   3.1 包含不等式 $\mathbf{y} \geqslant 0$。
   3.2 目标函数为 $\max c_u + \hat{\mathbf{c}}^T\mathbf{y}$。

一个例子的具体流程 (I)

考察如下的例子:

$$
\begin{aligned}
\max \quad & 2x_1 + 5x_2 \\
\text{s.t.} \quad & 2x_1 - x_2 \leqslant 4 && (1)\\
& x_1 + 2x_2 \leqslant 9 && (2)\\
& -x_1 + x_2 \leqslant 3 && (3)\\
& x_1 \geqslant 0 && (4)\\
& x_2 \geqslant 0 && (5)
\end{aligned}
$$

---

1. 原点对应的目标函数值为 0，有系数 >0，因此不是最大值。
2. 选择增大 x_2，当 $x_2 = 3$ 时，约束3被满足，此时目标函数值为 15。
3. 基于新顶点构造的坐标 (y_1, y_2) 满足:

$$y_1 = x_1,\ y_2 = 3 + x_1 - x_2$$

一个例子的具体流程 (II)

考察如下的例子:

$$
\begin{aligned}
\max \quad & 15 + 7y_1 - 5y_2 \\
\text{s.t.} \quad & y_1 + y_2 \leqslant 7 && (6)\\
& 3y_1 - 2y_2 \leqslant 3 && (7)\\
& y_2 \geqslant 0 && (8)\\
& y_1 \geqslant 0 && (9)\\
& -y_1 + y_2 \leqslant 3 && (10)
\end{aligned}
$$

---

1. 原点对应的目标函数值为 15，有系数 > 0，因此不是最大值。
2. 选择增大 y_1，当 $y_1 = 1$ 时，约束7被满足，此时目标函数值为 22。
3. 基于新顶点构造的坐标 (z_1, z_2) 满足:

$$z_1 = 3 - 3y_1 + 2y_2, \; z_2 = y_2$$

一个例子的具体流程 (III)

考察如下的例子:

$$
\begin{aligned}
\max \quad & 22 - \frac{7}{3}z_1 - \frac{1}{3}z_2 \\
\text{s.t.} \quad & -\frac{1}{3}z_1 + \frac{5}{3}z_2 \leqslant 6 \\
& z_1 \geqslant 0 \\
& z_2 \geqslant 0 \\
& \frac{1}{3}z_1 - \frac{2}{3}z_2 \leqslant 1 \\
& \frac{1}{3}z_1 + \frac{1}{3}z_2 \leqslant 4
\end{aligned}
$$

---

1. 原点对应的目标函数值为 22，所有系数均 $\leqslant 0$，因此是最大值，算法结束。

起始顶点的寻找

一个需要考虑的问题是，如何找到一个合适的起始顶点？原点并不总是可行的。

$$\begin{aligned}\min \quad & \mathbf{c}^{\mathsf{T}}\mathbf{x} \\ \text{s.t.} \quad & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geqslant 0\end{aligned}$$

---

1. 我们总可以假设所有的 $b_i \geqslant 0$。
2. 假设共有 m 个方程，添加 m 个新的变量 $z_1, \ldots, z_m$，并将 z_i 加入到第 i 个方程的左端。
3. 构造一个新的 LP，最小化 $z_1 + \cdots + z_m$。

- 新的 LP 显然有一个起始顶点，$z_i = b_i$，其余都为 0。
- 如果 $z_1 + \cdots + z_m$ 的最优值为 0，则原问题有解。
- 如果 $z_1 + \cdots + z_m$ 的最优值大于 0，则原问题无解。

事实上，一个顶点可能会是超过 n 个平面的交集。

1. 前面的例子中，顶点 B 是平面 $2, 3, 4, 5$ 的交集。

这导致了一个问题：单纯形化可能返回次优的顶点，仅仅由于该顶点的所有邻居都与其目标值相同。(退化的原因)

一个解决方案是引入一些微小的扰动 ϵ_i，使其能够区分不同的线性方程组解。

最后我们来关注一下单纯形法的运行时间:

考察一个普通的线性规划，其有 n 个变量 m 个不等式约束:

$$\begin{aligned} \max \quad & \mathbf{c}^{\mathsf{T}}\mathbf{x} \\ \text{s.t.} \quad & A\mathbf{x} \leqslant \mathbf{b} \\ & \mathbf{x} \geqslant 0 \end{aligned}$$

---

1. 一个顶点由 n 个不等式所确定，其至多可能有 mn 个邻居。
2. 朴素的检查每个邻居的时间复杂度为 $O(mn^4)$，因为这需要包含检查一个顶点是否是某 n 个方程组的解，运用 Gauss Elimination 需要 $O(n^3)$ 的时间。
3. 一个改进的方式是注意到重写仅仅替代了一个不等式，因此我们可以很快的判断一个坐标改变的最大幅度，从而整体的时间复杂度可以被优化至 $O(mn)$.
4. 但顶点数可能有 $\binom{n+m}{n}$，因此单纯形法是一个**指数时间**的算法。

关于多项式时间的线性规划讨论

单纯形法实际中很少出现需要迭代指数次的情形，因此是一个被广泛应用的高效算法。

椭球法 (Ellipsoid Method)

1979 年，Leonid Khachiyan 提出了椭球法，通过将多面体限制在一个越来越小的椭球体中得到最优解，且能证明其是一个多项式时间算法 (证明非常繁琐)，是一个非常重要的理论成就。

然而其在实际应用中却从来没有和单纯形法产生过对抗，从而形成了一个关于线性规划的悖论：一个理论上很成功，但另一个在实际运用中却很高效。

内点法 (Interior Point Method)

1984 年，UC Berkeley 的研究生 Karmarkar 提出了一个截然不同的思路-内点法，通过在多面体内部切割出一条巧妙地路径，从而达到最优解。该算法同样是一个多项式时间算法，且在实际应用中表现优异。

通过椭球法和单纯形法两个方法地惨烈竞争，最终发展出了极其高效的 LP 求解程序。

从线性规划的角度理解最大流最小割定理

我们实际上可以把最大流问题转换成如下的线性规划问题:

$$
\begin{aligned}
\max \quad & \sum_{e=(s,\_)\in E} f_e & & \\
\text{s.t.} \quad & \sum_{e=(\_,v)\in E} f_e - \sum_{e=(v,\_)\in E} f_e = 0 & \forall v \in V \setminus \{s,t\} & \quad (11)\\
& f_e \leqslant c(u,v) & \forall\, e = (u,v) \in E & \quad (12)\\
& f_e \geqslant 0 & \forall\, e = (u,v) \in E & \quad (13)
\end{aligned}
$$

- 我们给每一条边 e 都赋值一个未知数 f_e
- 约束式12和13表示了流的容量限制。
- 约束式11表示了流的守恒性。

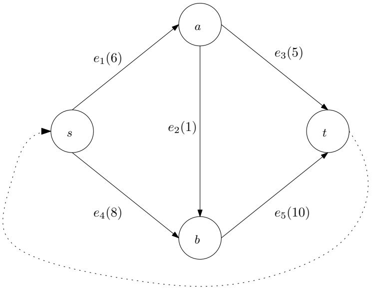
最大流问题的另一种线性规划表示

我们可以稍微修修改一下表示方法，，假设存在一条 t 回到 s 的边 $e_0=(t,s)$，令其容量没有限制，则我们可以把流守恒性的约束同样作用在 s 和 t 上，从而整个线性规划问题转换为：

$$
\begin{aligned}
\max \quad & f_{e_0} \\
\text{s.t.} \quad & \sum_{e=(\_,v)\in E} f_e - \sum_{e=(v,\_)\in E} f_e = 0 \quad && \forall v \in V && (14) \\
& f_e \leqslant c(u,v) && \forall\, e=(u,v)\in E && (15) \\
& f_e \geqslant 0 && \forall\, e=(u,v)\in E && (16)
\end{aligned}
$$

一个例子

对于下面的流网络：



对应的线性规划问题为：

$$
\begin{aligned}
\max\quad & f_{ts} \\
\text{s.t.}\quad & f_1 = f_2 + f_3,\ f_2 + f_4 = f_5,\ f_{ts} = f_1 + f_4,\ f_3 + f_5 = f_{ts} && (17)\\
& f_1 \leqslant 6,\ f_2 \leqslant 1,\ f_3 \leqslant 5,\ f_4 \leqslant 8,\ f_5 \leqslant 10 && (18)\\
& f_1,\ f_2,\ f_3,\ f_4,\ f_5 \geqslant 0 && (19)
\end{aligned}
$$

我们可以看到，约束式的线性组合其实给了目标函数的一个上界:

- $f_3 \leqslant 5$ 和 $f_5 \leqslant 10$ 组合可以得到 $f_{ts} = f_3 + f_5 \leqslant 5 + 10 = 15$
- $f_1 \leqslant 6$ 和 $f_4 \leqslant 8$ 组合可以得到 $f_{ts} = f_1 + f_4 \leqslant 6 + 8 = 14$

上述上界并不能一定取到，因为其不一定满足所有的约束。

将这些约束全部组合起来!

$$
\begin{aligned}
\max \quad & f_{ts} \\
\text{s.t.} \quad & f_2 + f_3 - f_1 = 0,\ f_5 - f_4 - f_2 = 0,\ f_1 + f_4 - f_{ts} = 0,\ f_{ts} - f_3 - f_5 = 0 && (20) \\
& f_1 \leqslant 6,\ f_2 \leqslant 1,\ f_3 \leqslant 5,\ f_4 \leqslant 8,\ f_5 \leqslant 10 && (21) \\
& f_1,\ f_2,\ f_3,\ f_4,\ f_5 \geqslant 0 && (22)
\end{aligned}
$$

---

1. 第二行的每个等式都是跟某个顶点 u 相关的，我们记相应的等式左边为 M_u 并赋给每个等式一个变量 x_u。
2. 第三行的每个不等式都是跟某条边 e 相关的，我们赋给每个不等式一个变量 y_e。由边的记号这里可以用 y_i 来表示。

其线性组合可以表示为:

$$\sum_{u \in U} x_u \cdot M_u + \sum_{i=1}^{5} y_i \cdot c_i \leqslant \sum_{i=1}^{5} y_i \cdot c_i$$

当我们有:

$$\sum_{u \in U} x_u \cdot M_u + \sum_{i=1}^{5} y_i \cdot c_i \geqslant f_{ts} \text{ 恒成立}$$

上述不等式的右边 ($\sum_{i=1}^{5} y_i \cdot c_i$) 提供了一个 f_{ts} 的上界.

- 我们可以考虑将这个值最小化的问题，它也是个线性规划。

我们将相应的问题表示出来为：

$$\min \quad \sum_{i=1}^{5} y_i \cdot c_i$$

$$
\begin{aligned}
\text{s.t.}\quad & x_a - x_s + y_1 \geqslant 0,\ x_b - x_a + y_2 \geqslant 0,\ x_t - x_a + y_3 \geqslant 0 && (23)\\
& x_b - x_s + y_4 \geqslant 0,\ x_t - x_b + y_5 \geqslant 0 && (24)\\
& x_s - x_t \geqslant 1 && (25)\\
& x_u, y_i \geqslant 0 \qquad u \in V,\ i \in \{1.2.3.4.5\} && (26)
\end{aligned}
$$

这恰巧是求上述流网络最小割的线性规划问题！

- 直观上理解，可以认为 x_u 是一个用来划分顶点集合的变量，当 $x_u \neq x_v$ 不相等时 (u, v) 就是一条跨边，也就是割的一部分，从而相应的 y_i 必须得是 1。

我们称这样的问题为原问题的**对偶问题**。

一般情况下，给定一个流网络最大流的线性规划问题：

$$
\begin{aligned}
\max \quad & f_{e_0} & & \\
\text{s.t.} \quad & \sum_{e=(\_,v)\in E} f_e - \sum_{e=(v,\_)\in E} f_e = 0 & \forall v \in V & \qquad (27) \\
& f_e \leqslant c(u,v) & \forall\, e = (u,v) \in E & \qquad (28) \\
& f_e \geqslant 0 & \forall\, e = (u,v) \in E & \qquad (29)
\end{aligned}
$$

其相应的对偶问题为：

$$
\begin{aligned}
\min \quad & \sum_e c_e \cdot y_e & & \\
\text{s.t.} \quad & x_v - x_u + y_e \geqslant 0 & \forall e = (u,v) \in E & \qquad (30) \\
& x_s - x_t \geqslant 1 & & \qquad (31) \\
& y_e, x_u \geqslant 0 & \forall\, e \in E,\ u \in V & \qquad (32)
\end{aligned}
$$

最大流与最小割问题联系-对偶问题 (II)

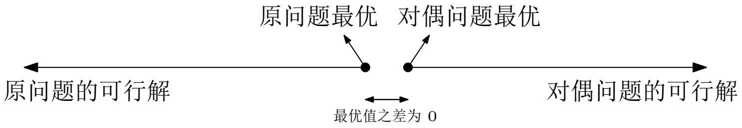
最大流与最小割定理告诉我们，上述两个问题的解是相同的，即：

$$\max\ f_{e_0} = \min \sum_e c_e \cdot y_e$$

这不是一个个例，事实上：

定理 1 **[对偶定理].**

如果线性规划问题的目标函数有界，则对偶问题的目标函数也有界，并且两者相等。



本章总结

本章具体内容

- 线性规划介绍。
- 线性规划求解-单纯形法。
- 从线性规划的角度理解最大流最小割定理。