# 交通工程学
# Introduction to Traffic Engineering
## 第 13 节　网络优化简介[1]

葛乾

西南交通大学　系统科学与系统工程研究所
西南交通大学　交通工程系

[1]基于麻省理工大学 James Orlin 教授与亚利桑那州立大学 Angelia Nedich 教授讲义。仅供教学使用，严禁传播

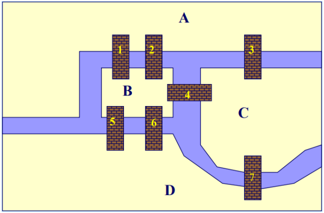
# 本节目录

1. 网络优化简介
2. 最短路问题回顾
3. 最大流问题
   - Ford Fulkerson 算法
4. 最小费用流问题
   - 环消除 (Cycle Canceling) 算法
   - 网络单纯形法 (Network Simplex)
   - 原-对偶法 (Primal-Dual)

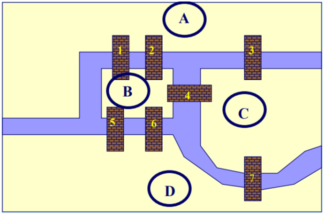
# 柯尼斯堡七桥问题

- 图论与网络优化问题肇始于 1736 年，欧拉所提出的七桥问题
- 柯尼斯堡有七个桥，问题是：是否可以从某点出发，找到一个路径使得刚好穿过每个桥一次，回到该点?
- 通常认为不存在

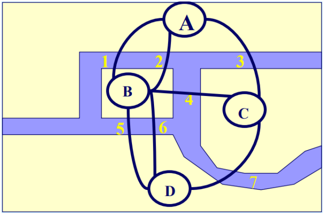
# 图示



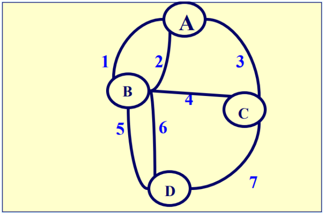

(a) 现实路网




(b) 区域视为节点




(c) 桥梁视为弧段



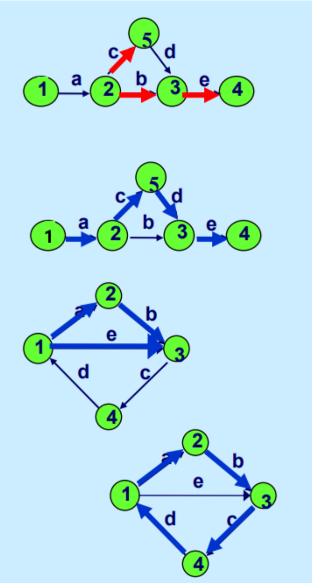

(d) 抽象问题

是否可以从某点出发，找到一个路径使得刚好穿过每个桥一次，回到该点?

# 符号表示

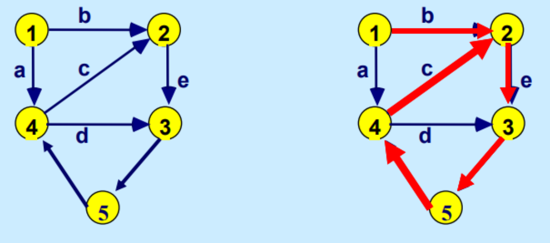

Figure: 无向图与有向图

- 网络（图）：G(N,A)
- 节点集合：N={1,2,3,4}
- 弧段集合：A={(1,2),(1,4),(3,2),(3,4),(2,4)}
    - 无向图中 (i,j)=(j,i)

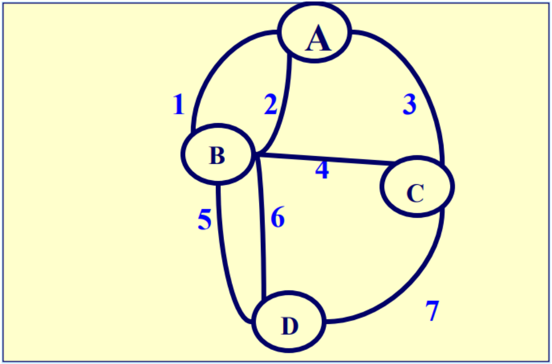


- **路径** (path)：如 $5, 2, 3, 4$ 或 $5, c, 2, b, 3, e, 4$
  - 无重复节点
  - 不考虑路段方向
- **有向路径** (directed path)：如 $1, 2, 5, 3, 4$ 或者 $1, a, 2, c, 5, d, 3, e, 4$
  - 无重复节点
  - 考虑路段方向
- **环** (cycle)：如 $1, 2, 3, 1$ 或 $1, a, 2, b, 3, e, 1$
  - 有 2 个或以上节点构成，第一个节点与最终节点相同
  - 不考虑路段方向
- **有向环** (directed cycle)：如 $1, 2, 3, 4, 1$ 或 $1, a, 2, b, 3, c, 4, d, 1$
  - 无重复节点
  - 考虑路段方向

# 回路 (walk)

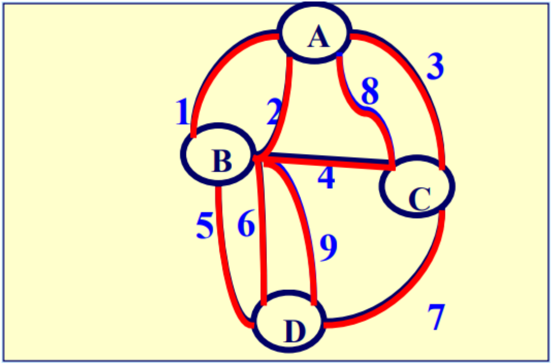



- Walks are paths that can repeat nodes and arcs
- Example of a directed walk: $1 - 2 - 3 - 5 - 4 - 2 - 3 - 5$
- A walk is closed if its first and last nodes are the same
- A closed walk is a cycle except that it can repeat nodes and arcs.




- 回到一开始地问题：是否可以从某点出发，找到一个回路使得刚好穿过每个桥一次，回到该点?
- 这种回路被成为**欧拉回路**

# 例子




- 在柯尼斯堡地路网上增加两个弧段 8 与 9
- 可得一个欧拉回路：A, 1, B, 5, D, 6, B, 4, C, 8, A, 3, C, 7, D, 9, B, 2, A
- 注意此时，与 B 节点相连的路段出现次数，刚好是 B 节点出现次数的 2 倍
- **度** (degree)：在无向图中，一个节点的度为与之相连的路段个数

# 欧拉定理

## 欧拉定理

一个无向图中存在欧拉回路当且仅当其满足以下条件：

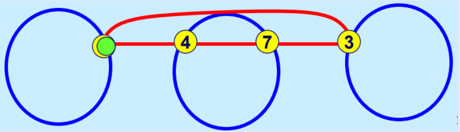
- 每个节点的度均为偶数
- 该图为连通的（任意两点均有路径相连）

## 证明

**必要性**：任意欧拉回路访问每个节点偶数次（进与出）；任意欧拉回路的构成，证明网络全连通

**充分性**：假设对于图的子图以下结果均成立：从某一节点出发，存在一个形成环的回路 $C$

对于剩下部分 $G' = (N, A/C)$: 每个节点的度均为偶数；该图为连通的（任意子图均有路径相连），因此 $G'$ 为欧拉回路的并集，将 $G'$ 与 $C$ 相连，可得一个欧拉回路

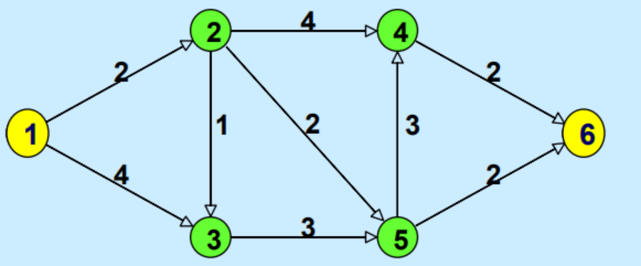


# 三个最基本的网络优化问题

- 最短路问题 (不深入探讨)
- 最大流问题 (Ford Fulkerson 算法)
- 最小费用流问题（Cycle Canceling 算法）

# 本节目录

# 最短路问题描述




- 现有一个网络 $G = (N, A)$，起点为 $s$，终点为 $t$
- 表示：$n = |N|$，$m = |A|$，路段 $(i, j)$ 的权重 $w_{ij}$
- 寻找从 $s$ 到 $t$ 的最短路

# 整数规划建模

定义变量 $x_{ij}$ 表示是否使用路段 $(i,j)$，则该问题的整数规划模型为

$$\min \sum_{(i,j)\in A} w_{ij}x_{ij}$$

subject to

$$x_{ij} - x_{ji} = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases}, \forall i \in N$$

$$x_{ij} = \{0, 1\} \ \forall (i,j) \in A$$

# 求解算法

- 标号设置法（Dijkstra 算法）
- 标号修改法（Bellman-Ford 算法）$\rightarrow$ 动态规划的经典应用

此处不展开，感兴趣的同学可以复习下“数据结构与算法”课程所学内容

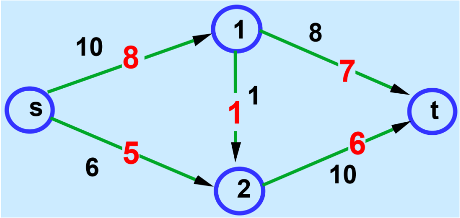
# 本节目录

# 最大流问题

- $G=(N,A)$
- $x_{ij}=$ 弧段 $(i,j)$ 上的流量
- $u_{ij}=$ 弧段 $(i,j)$ 的容量
- $s=$ 起点
- $t=$ 终点

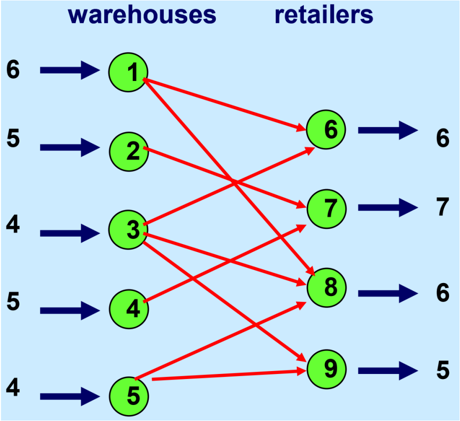
$$
\begin{aligned}
\max \quad & v \\
\text{s.t.} \quad & \sum_j x_{ij} - \sum_k x_{ki} = 0 \ \forall i \in N, i \neq s, t \\
& \sum_j x_{sj} = v \\
& 0 \leqslant x_{ij} \leqslant u_{ij} \ \forall (i,j) \in A
\end{aligned}
$$

# 最大流问题图示

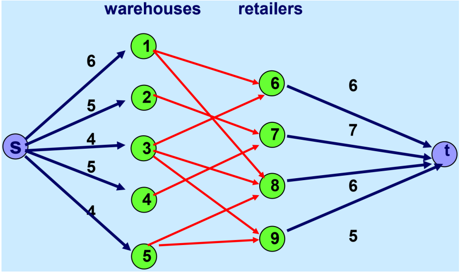



We refer to a flow $x$ as maximum if it is feasible and maximizes $v$. Our objective in the max flow problem is to find a maximum flow
图中所示为一个可行解，但不是最大流

# 可行性问题：寻找一个可行的流量

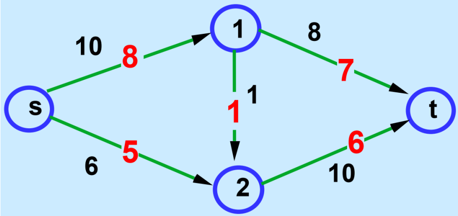



是否存在可行解，使得从仓库到零售商的配送刚好满足需求？

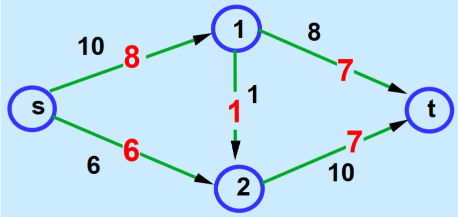



从 $s$ 到 $t$ 的流量为 24，这与上页所述的运输问题的可行解刚好对应

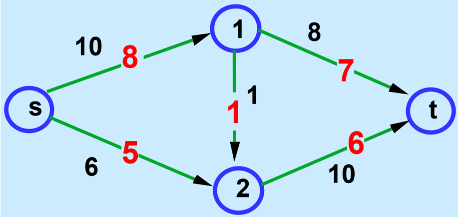
# 在路径中增加流量



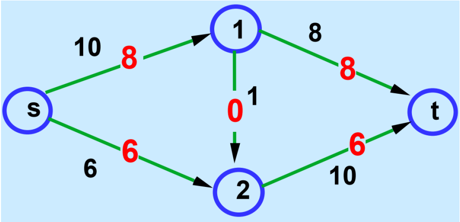

沿着路径 $s-2-t$ 还可以再输送 1 个单位的流量



# 另一种路径




沿着路径 $s-2-1-t$ 也可以再输送 1 个单位的流量

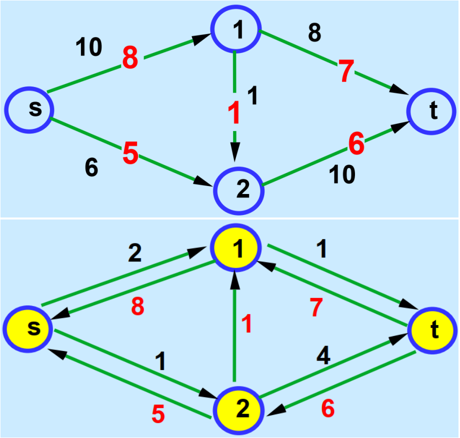


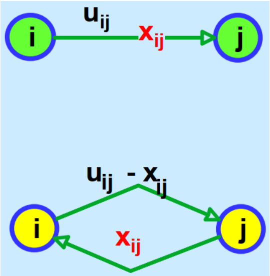

这时，观察到路段 (1,2) 上的流量下降了 1 个单位，从数学上，上图到下图的变化，等同于从反方向 (2,1) 输送了 1 个单位的流量，同时保持约束条件得以满足

# 本节目录

1. 网络优化简介
2. 最短路问题回顾
3. 最大流问题
   - Ford Fulkerson 算法
4. 最小费用流问题
   - 环消除 (Cycle Canceling) 算法
   - 网络单纯形法 (Network Simplex)
   - 原-对偶法 (Primal-Dual)

# 剩余网络 (residual network)



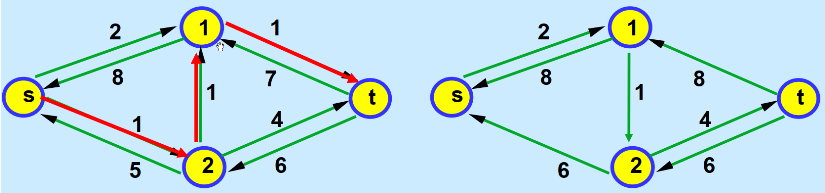

我们一般用 $r_{ij}$ 表示弧段 $(i, j)$ 上的剩余容量，因而 $r_{ij} = u_{ij} - x_{ij}, r_{ji} = x_{ij}$。有时在图上也将二者标注在一起 $r_{ij}/r_{ji}$

下图为剩余网络

# 增量路径

- **增量路径** $P$ 为剩余网络中，从起点 $s$ 到终点 $t$ 的路径
- **增量路径的容量**为构成路径的路段中的最小值 $\delta(P) = \min\{r_{ij} : (i,j) \in P\}$
- 路径 P 上的增量，即经过构成 P 的每个路段，额外运输 $\delta(P)$ 单位的流量，之后对流量 $x_{ij}$ 与剩余容量都需要进行相应修改

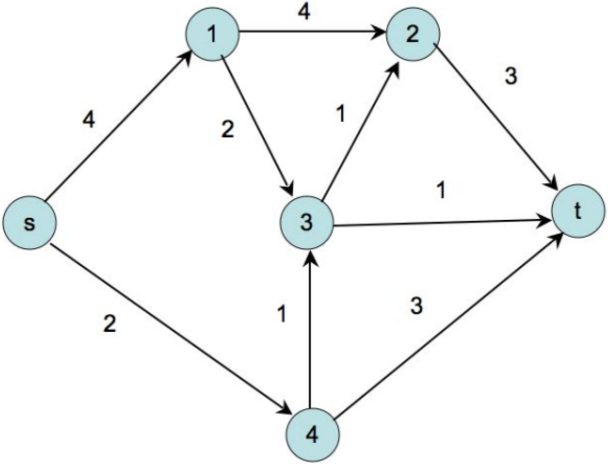
$$r_{ij} := r_{ij} - \delta(P)$$
$$r_{ji} := r_{ji} + \delta(P) \ \forall (i,j) \in P$$



# Ford Fulkerson 算法

1. $x_{ij} := 0 \ \forall (i,j) \in A$
2. 创建剩余网络 $G(x)$
3. while 剩余网络中存在从 $s$ 到 $t$ 的有向路径
   - 令 $P$ 为 $G(x)$ 中从 $s$ 到 $t$ 的有向路径
   - 计算剩余流量 $\delta(P) = \min\{r_{ij} : (i,j) \in P\}$
   - 沿着路径 $P$ 运送 $\delta(P)$ 单位流量
   - 更新剩余流量

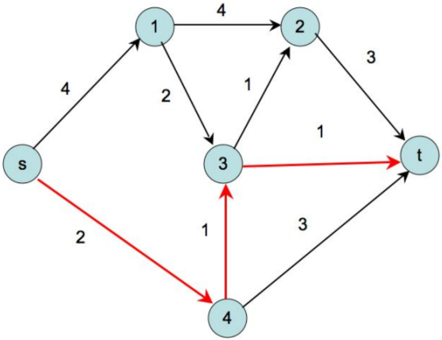
# 一个完整的例子

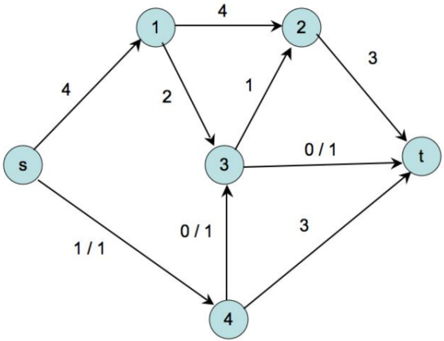



我们用 $v_i$ 表示第 $i$ 次迭代时输送的流量

# 第 1 次迭代

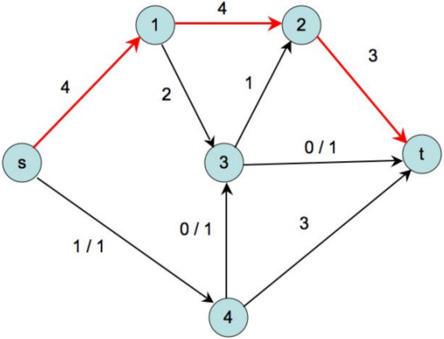
- 我们发现 $s-4-3-t$ 有正向的流量
- 可以运输的流量为 $v_1 = \min\{2, 1, 1\}$
- 沿着该路径运输 1 个单位流量，剩余网络更新如下图

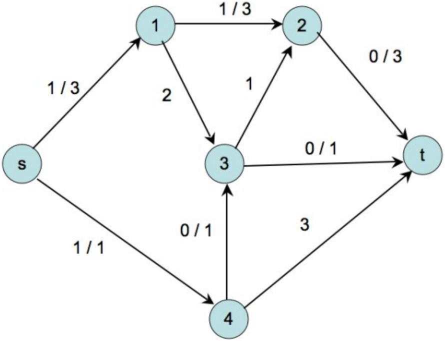


# 第 2 次迭代

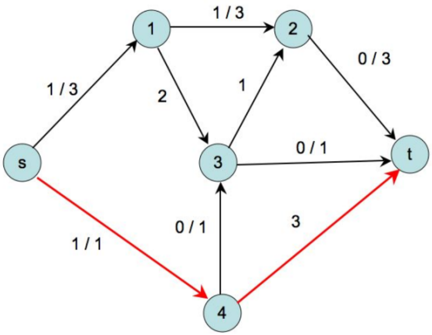
- 我们发现 $s-1-2-t$ 有正向的流量
- 可以运输的流量为 $v_2 = \min\{4,3,3\}$
- 沿着该路径运输 3 个单位流量，剩余网络更新如下图

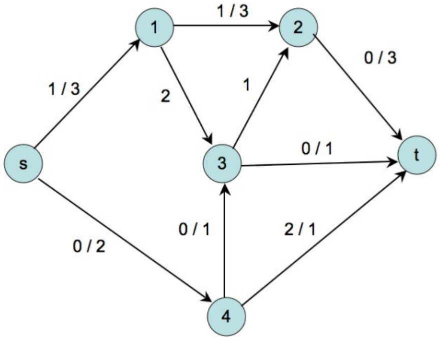


# 第 3 次迭代

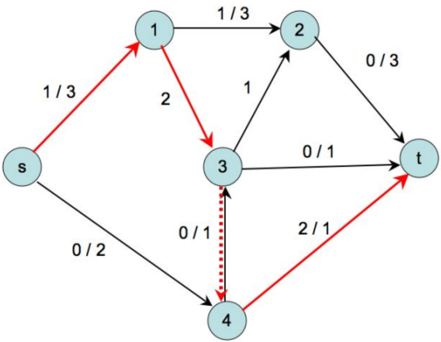
- 我们发现 $s-4-t$ 有正向的流量
- 可以运输的流量为 $v_3 = \min\{1, 3\}$
- 沿着该路径运输 1 个单位流量，剩余网络更新如下图

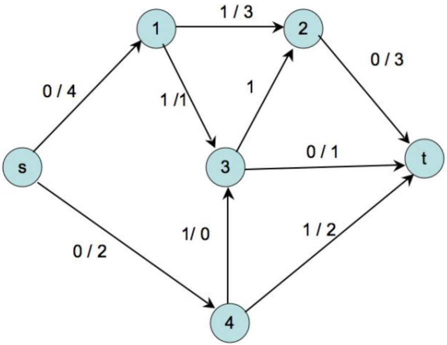


# 第 4 次迭代

- 我们发现 $s-1-3-4-t$ 有正向的流量
- 可以运输的流量为 $v_4 = \min\{1, 2, 1, 2\}$
- 沿着该路径运输 1 个单位流量，剩余网络更新如下图



# 结果说明

- 在第 4 次迭代结束后，我们停止迭代，可观察到节点 s 与其他节点不再连通，剩余网络中也不存在增量路径了
- 最大流为: $v_1 + v_2 + v_3 + v_4 = 1 + 3 + 1 + 1 = 6$

# 算法的准确性证明步骤

- 不变性：每次迭代生成的均是从 $s$ 到 $t$ 的可行流量
- 有限性：剩余容量总是整数值；每次迭代从起点 $s$ 流向的路段，其剩余容量至少下降 1 个单位
- 正确性：当不存在增量路径时，流量必为最大值；**最大流-最小割定理** (max-flow min-cut theorem)

# 本节目录

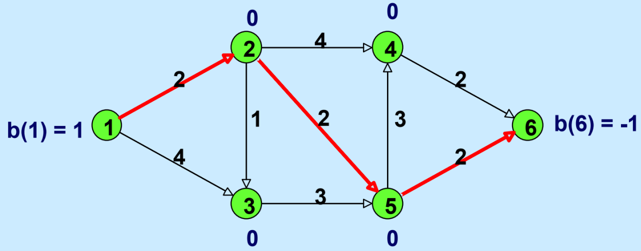
1. 网络优化简介
2. 最短路问题回顾
3. 最大流问题
   - Ford Fulkerson 算法
4. 最小费用流问题
   - 环消除 (Cycle Canceling) 算法
   - 网络单纯形法 (Network Simplex)
   - 原-对偶法 (Primal-Dual)

# 最小费用流问题

- $u_{ij}$= 弧段 (i,j) 的容量
- $w_{ij}$= 在弧段 (i,j) 运送单位流量的成本
- $x_{ij}$= 弧段 (i,j) 的流量

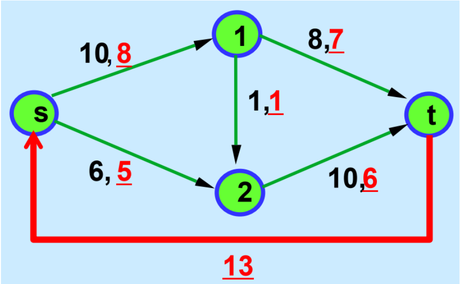
$$
\begin{aligned}
\max \quad & \sum_{(i,j)\in A} w_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{j} x_{ij} - \sum_{j} x_{ji} = b_i \ \forall i \in N \\
& 0 \leqslant x_{ij} \leqslant u_{ij} \ \forall (i,j) \in A
\end{aligned}
$$

# 与最短路问题的关系



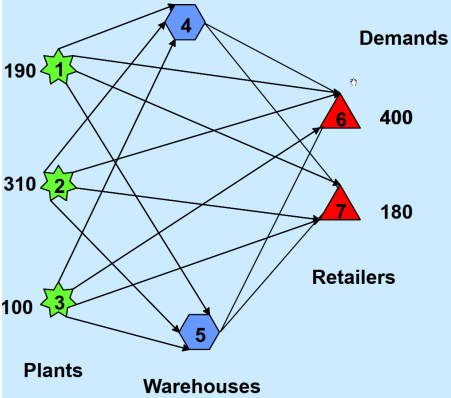

- 从 1-6 的最短路问题的最优解，是通过路径 1-2-5-6 运送单位流量
- 当网络中不存在负环时，该转换有效

# 与最大流问题的关系




- 对于所有节点 $i$， b(i) = 0；添加虚拟路段 (t,s)， 其费用为-1， 容量为很大值；其余路段费用为 0；
- 对应的最小费用流问题将求得从 $t$ 到 $s$ 的最大流

# 转运问题

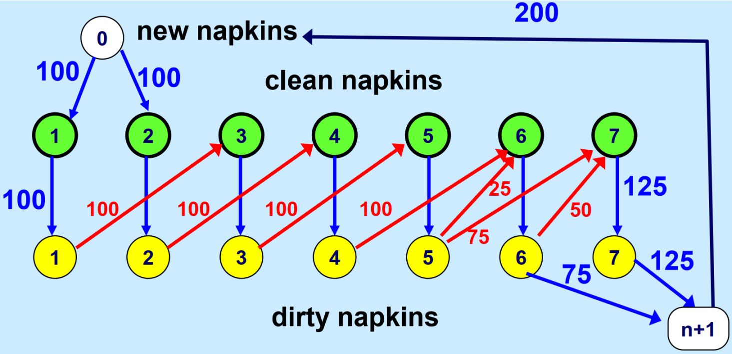



- 各制造厂对某种类产品有既定的生产能力
- 可从制造厂直运到零售商，也可从制造厂运货到仓库，然后由仓库再配送到零售商
- 每个零售商的需求是给定的
- 每次配送的成本也是给定的
- 满足零售商需求的最小成本配送策略是什么?

# Caterer 问题

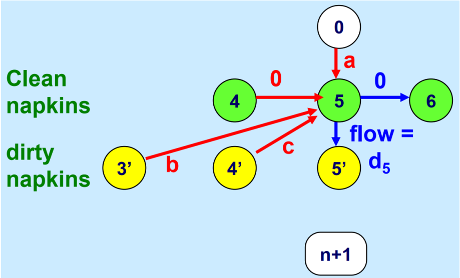
- 问题描述
  - 从周一到周日，第 $i$ 天需要 $d_i$ 个毛巾
  - 每个新毛巾的单价为 a
  - 正常清洗（花费 2 天）单价为 b
  - 加急清洗（花费 1 天）单价为 c
- 假设
  - 刚开始没有毛巾的库存
  - 以一周为单位，结束时没有干净的毛巾剩余

# 具体例子



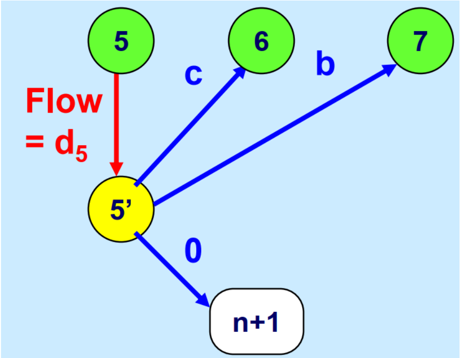

- 新毛巾单价为 10，正常清洗价格为 1，加急清洗价格为 2
- 平日需求量为 100，周末为 125
- 节点 0 表示毛巾起始状态，$n+1$ 表示结束状态，上排 1 到 7 表示星期 X 干净毛巾数，下排表示使用过的毛巾数

# 状态变化

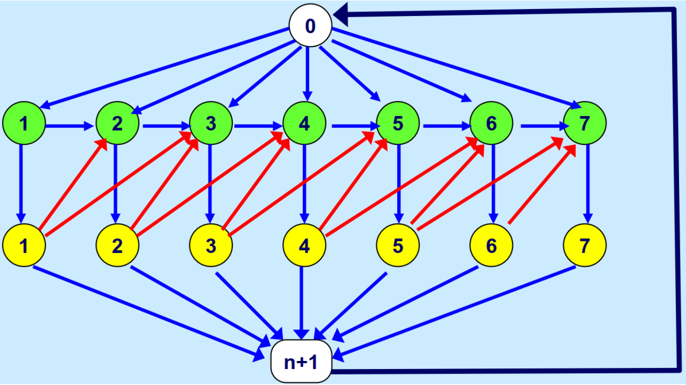



- 周五干净的毛巾来自：新购 (0,5)，昨日剩余 (4,5)，正常清洗 (3',5)，加急清洗 (4',5)
- 周五干净的毛巾去往：使用 (5,5')，剩余 (5,6)

# 状态变化 (cont.)




- 周五使用过的毛巾来自：本日使用 (5,5’)
- 周五使用过的毛巾去往：废弃 (5’,n+1)，加急清洗 (5’,6)，正常清洗 (5’,7)




- Find a minimum cost circulation such that the flow on $(j, j') = d_j$ 寻找 $(j, j')$ 上的最小成本环路，$(j, j')$ 上的流量等于当天需求
- on arcs $(j, j')$ for $j = 1$ to $n$. Lower bound = upper bound = $d_j$ 对于 $(j, j')$ 弧段，其流量的上下界均等于当天需求
- Arc $(n + 1, 0)$ : each purchased napkin ends up dirty. 弧段 $(n + 1, 0)$ 指所有毛巾最终必须均变脏 (无剩余)

# 基本假设

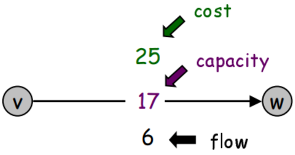
- 所有数据均为整数
- 网络为有向的，且是连通的
- $\sum_i^n b(i) = 0$，否则不存在可行解

# 求解思路

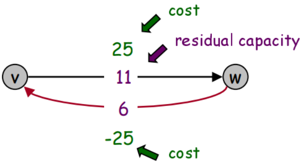
- 先找个可行解，再从可行解出发，寻找能降低成本的新解

# 最小费用流中的剩余网络

相比最大流问题，还需要考虑费用。

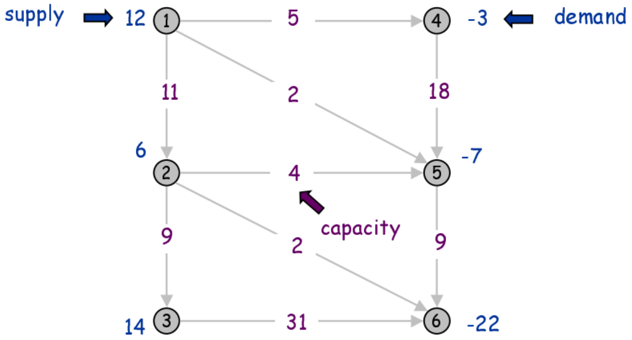



原始网络




剩余网络

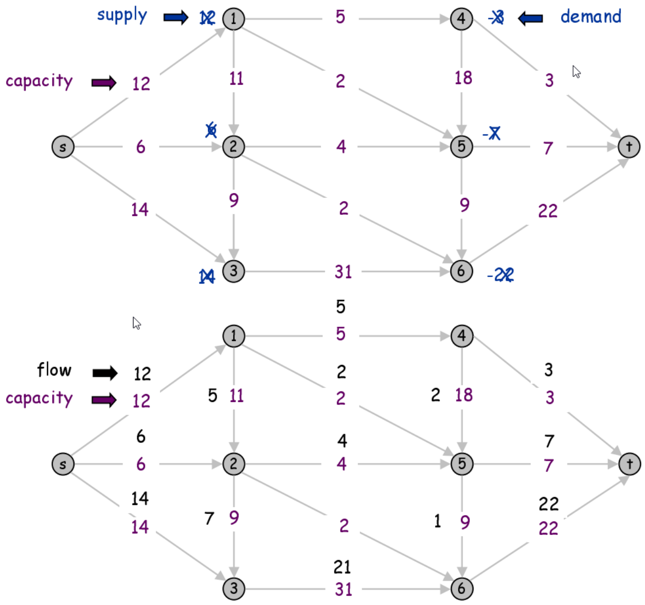
# 寻找可行解



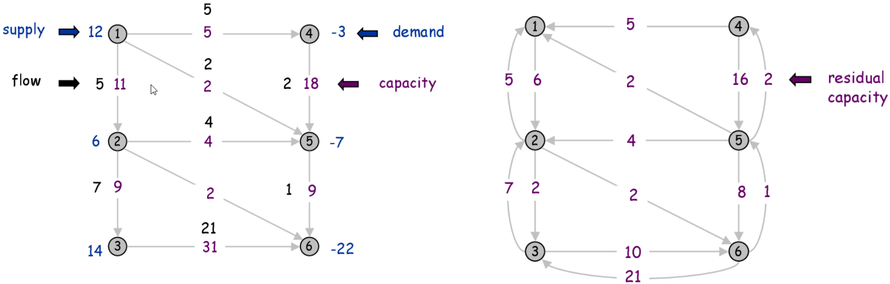

给定带供需节点的网络，求可行的流量

- 加入两个虚拟节点 $s$, $t$，求解最大流问题（此处暂时忽略供需节点）

# 对应的最大流问题


supply
demand
capacity
flow
s
t
1
2
3
4
5
6
12
11
5
2
18
3
6
4
7
14
9
22
31
21

# 流量剩余网络



supply
12
5
5
-3
demand
flow
5 11
2
2
2 18
capacity
6
4
4
-7
7 9
2
1 9
14
21
31
-22
5
5 6
2
16 2
residual
capacity
4
7 2
2
8 1
10
21

# 费用剩余网络

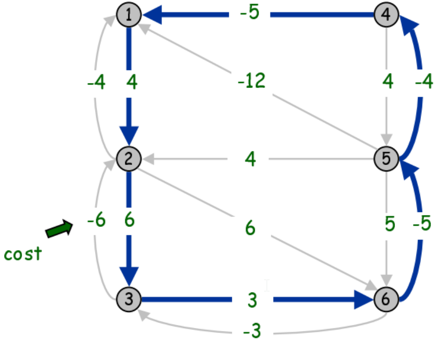



Figure: 图中 1-2-3-6-5-4-1 为负环，单位流量的费用为-1，容量为 $\min = \{6, 2, 10, 1, 2, 5\} = 1$

# 本节目录

# 环消除算法

- 如何在保证可行性的情况下，降低费用?
  - 增量环：费用剩余网络中的负费用环（总和为负）
  - 为什么一定需要是环？→ 保证可行性 (参考最小费用流问题的约束条件，只有当形成环时，每个节点处的流量才守恒)
  - 为什么费用为负？→ 保证费用下降

# 环消除算法

1. 创建可行解 $x$，并求解对应的剩余网络 $G(x)$
2. while 剩余网络中存在负费用环
   - 使用某些算法识别一个负环 $C$（例如最短路算法）
   - 计算剩余流量 $\delta(P) = \min\{r_{ij} : (i,j) \in C\}$
   - 沿着环 $C$ 增加 $\delta(P)$ 单位流量
   - 更新剩余网络 $G(x)$

# 完整例子–第 1 次迭代

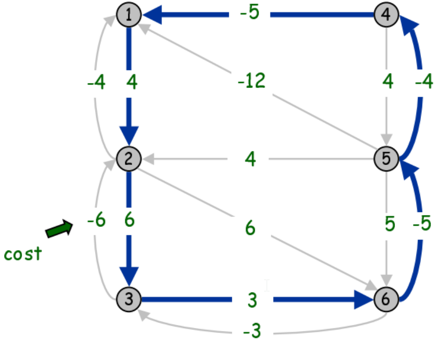



Figure: 图中 1-2-3-6-5-4-1 为负环，单位流量的费用为 4+6+3-5-4-5=-1，容量为 min $= \{6, 2, 10, 1, 2, 5\} = 1$

# 第 2 次迭代

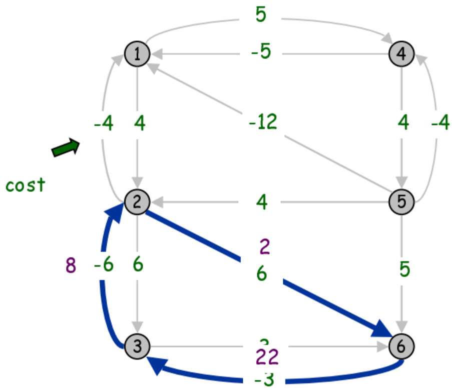



Figure: 图中 2-6-3 为负环，单位流量的费用为 6-3-6=-3，容量为 min $= \{2, 22, 8\} = 2$

# 第 3 次迭代

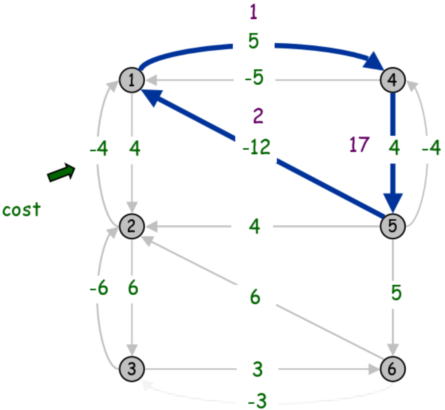



Figure: 图中 1-4-5 为负环，单位流量的费用为 5+4-12=-3，容量为 $\min = \{1, 17, 2\} = 1$

# 本节目录

1. 网络优化简介
2. 最短路问题回顾
3. 最大流问题
   - Ford Fulkerson 算法
4. 最小费用流问题
   - 环消除 (Cycle Canceling) 算法
   - 网络单纯形法 (Network Simplex)
   - 原-对偶法 (Primal-Dual)

# 本节目录

谢谢!