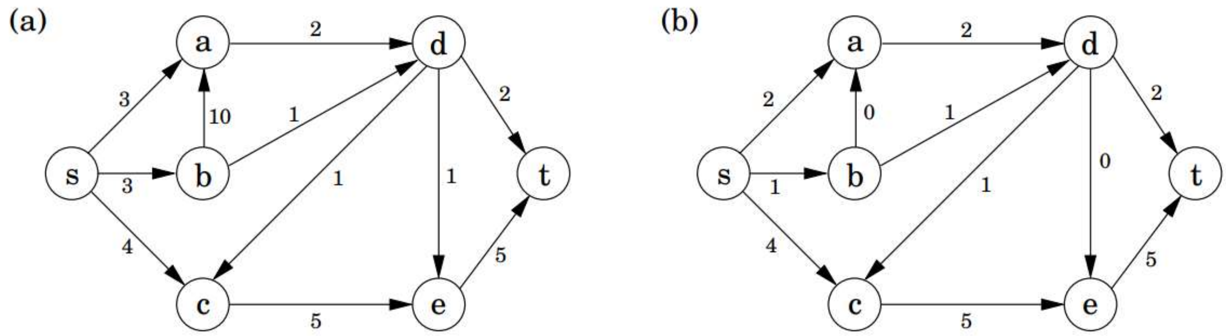
网络流

> 喻勃洋 2023.4.21 算分小班

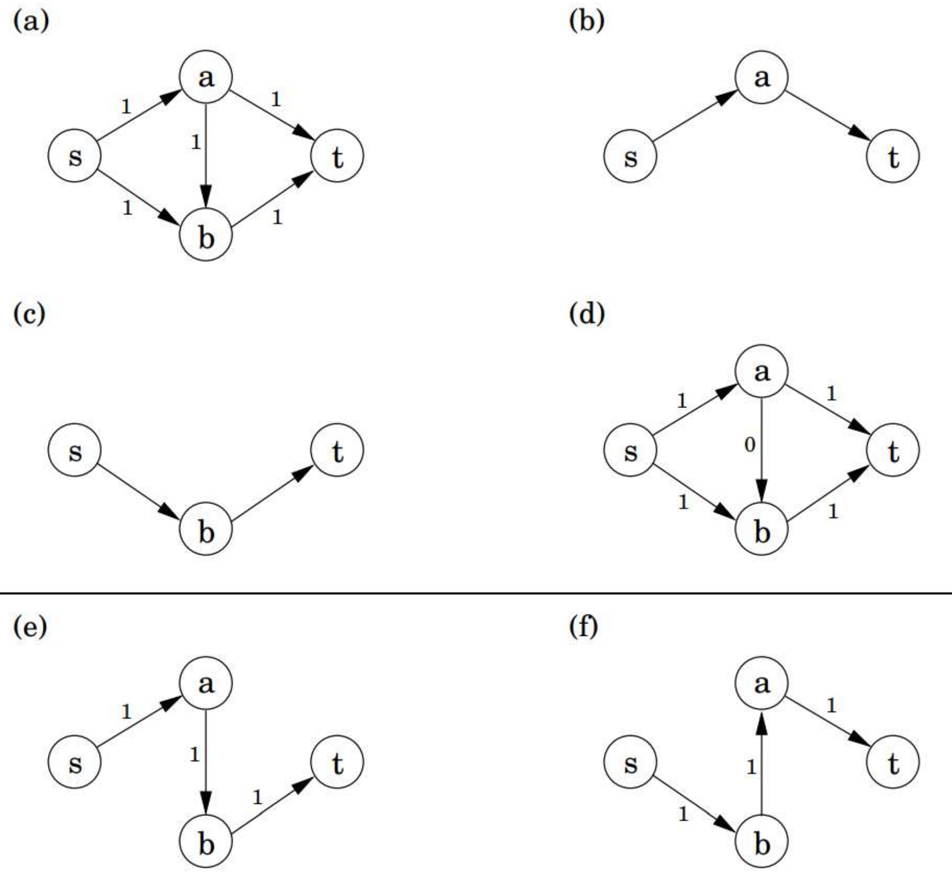
- 问题的起源： Shipping oil
  - 表现流的运输网络：
    - 有向
    - 承载能力有上限
    - 不能存储，要满足流的连续性原理
    - > 注意这与物理意义上的流不同，水流等的流量是速度和截面的函数，但我们这里相当于规定了速度为1，故不总能满足截面的上界
    - 

- 最大流
  - 基本假设：
    - $G=(V,E)$; $s,t\in V$; $c_e>0$ （不妨假设s只出，t只进，便于后面讨论）
    - 对所有边e:$0\le f_e\le c_e$
    - 对于所有非s,t的点u:

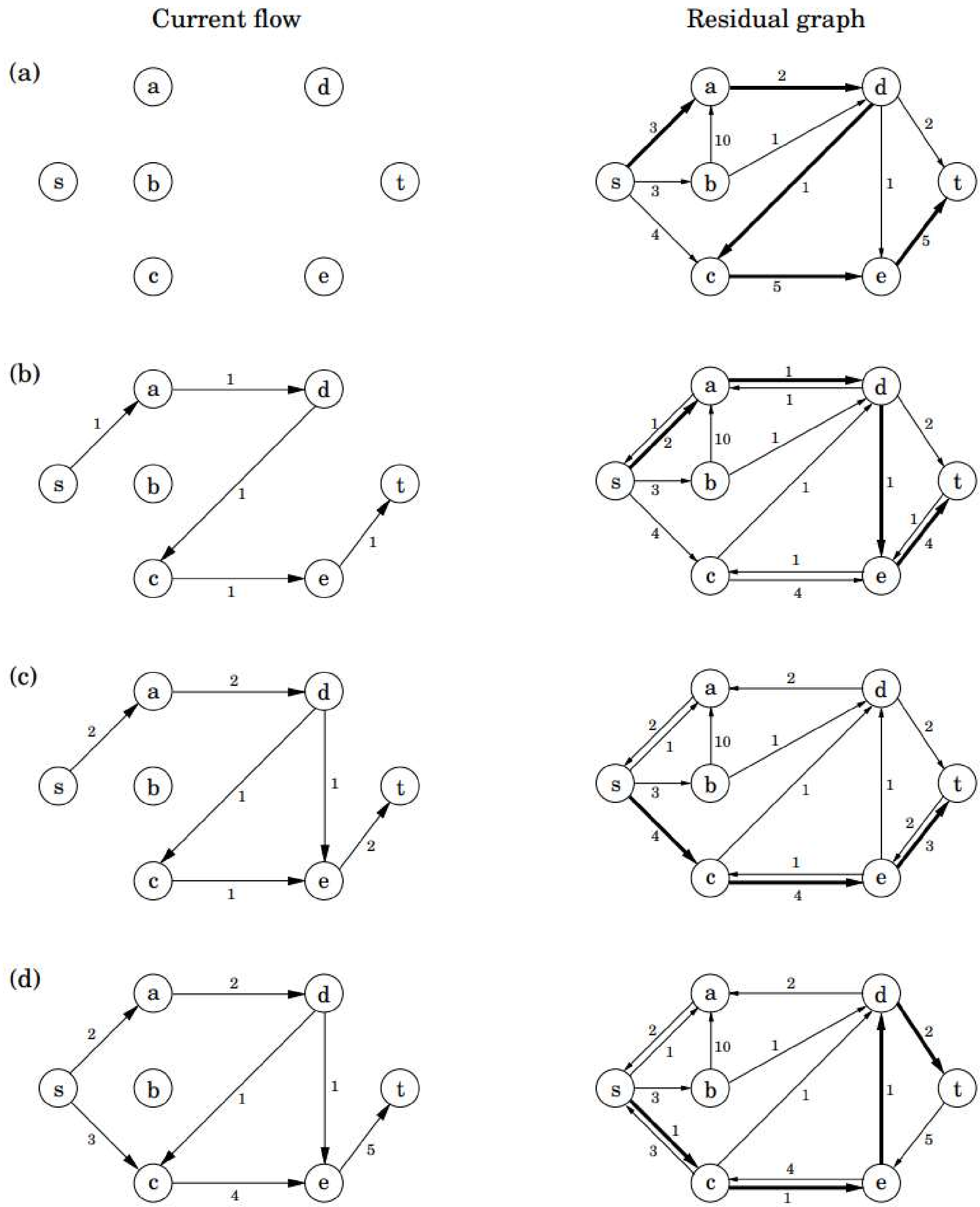
$$\sum_{(w,u)\in E} f_{wu}=\sum_{(u,z)\in E} f_{uz}$$

> flow is conserved

    - 定义 $size(f)=\sum_{(s,u)\in E} f_s u$
  - 综上假设，可以看出线性约束+线性目标函数——线性规划问题
  - 因此我们可以很简单的用单纯形法求解
- 能更进一步吗?
  - 思考单纯形法的本质
    - 在约束规定的区域内，在边界上跑（改变变量的值），且要求变化之后的目标函数值增大，直到找到使目标函数最大的位置
  - 所以实际上单纯形法在最大流中的操作
    - 从无流开始
    - 重复的寻找$s\to t$ 的通路(how?)，并在其上加上最大可能的flow，作为一次优化
  - 每次添加新的$s\to t$ 通路 需要满足什么要求呢
    - 目前全图已经有了一个flow的分配：f_{uv}，可以维护一个$G^f=(V,E^f)$来储存
    - 对于找到的通路的每个边$u\to v$ ，如果符合有向，则最多能添加$c_{uv}-f_{uv}$的flow
    - 而如果与本来的有向相反，则可以用来抵消一部分原来的flow，故给出最多f_{vu}的flow

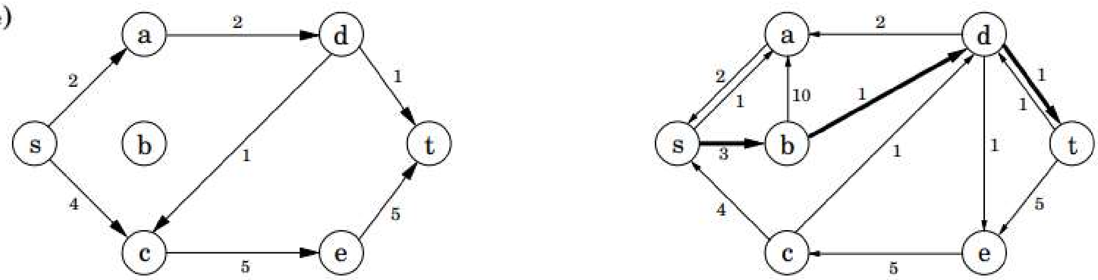
- 

- 以此规律，我们得到了一个每条边有确定上限的通路，则显然添加的最大flow为最小的边上限
- 添加flow后，要更新G^f，从而迭代的寻找下一个优化

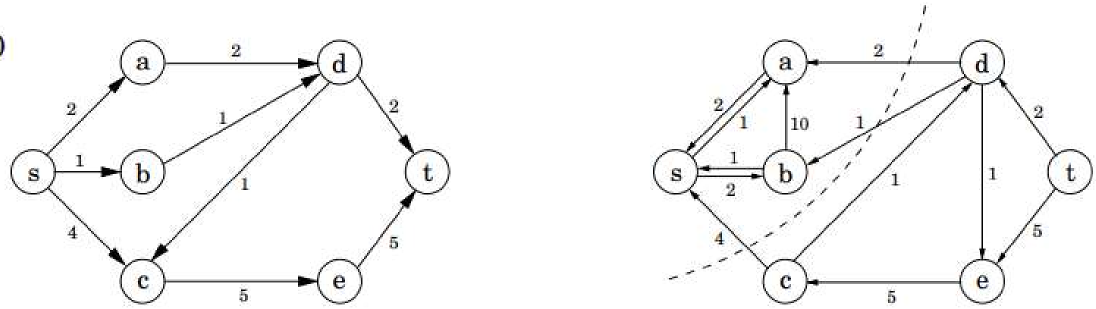

Current flow
Residual graph
(a)
(b)
(c)
(d)

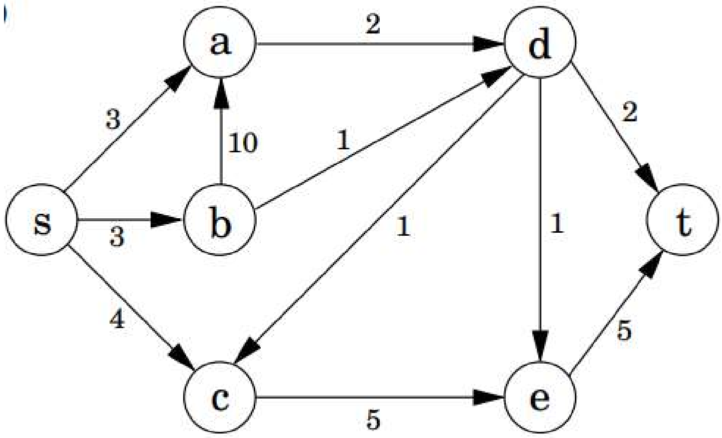
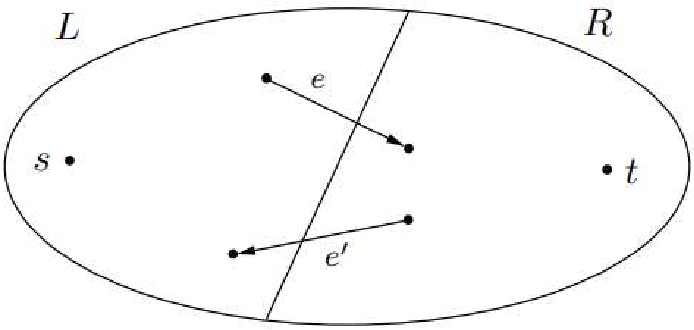
▪

(e)



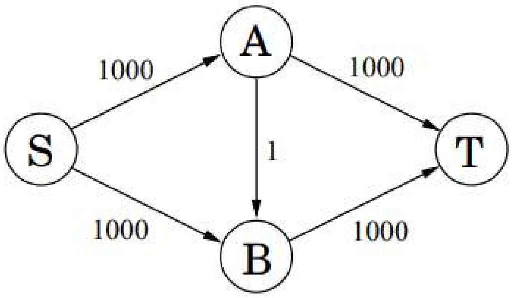
(f)



- 最优化条件？（A certificate of optimality）
  - 为什么我们认为我们算法的终止即是最优解
  - 单纯形法的终止：
    - 在当下，我们再也找不到任何一种变换使得目标函数更优
    - 其来源是约束方程与目标函数的关系
  - 观察网络流的直观上限
    - 对于任何将s,t分开的分割L和R，跨越分割的$R \to L$ 的线的总和的最大流的上界（但不总是紧的）,记为$capacity(L, R)$
    - 
  - 何时是紧的上界
    - 算法终止——无$s \to t$ 通路——从s能走到的点中没有t
    - 因此可以以此来做一个分割，所有从s能走到的点为L，其余的点为R
    - 故，任何从L到R的边都是满的，任何从R到L的边都是0
    - 

    - 故，此时有$size(f) = capacity(L, R)$
      - 此上界即为紧的上界，此解即为最优解
    - 有定理：

    - 最小的分割上界即为最大流的最大值
    - > Max-flow min-cut theorem The size of the maximum flow in a network equals the capacity of the smallest (s,t)-cut.
  - 例题（18年期末）：
    - > 如果 f 是最大流，那么不存在关于 f 的 s-t 增广链。（√）

- 算法效率
  - 简单分析
    - 每次搜索到一个通路：$O(|E|)$
    - 总共的迭代次数:
      - 假设边都是整数的承载量，且最大值为C，则最大流的上界总是小于$C\,|E|$
      - 且每次找到新通路我们至少增添了1的总流量
    - 因此总复杂度在$O(C|E|^2)$
  - 这是紧的吗
    - C可能很大，边值可能都很大，如果能一次添加很大的流量，则迭代次数可以降低
  - 通路的搜索策略
    - 看一个例子
    - 

    - 我们尽量寻找可能的最大流量来作为迭代——贪心
      1. 用Dijkstra的变式算法来解出，图中具有最大流量的通路
      2. 最大流的最大值总是小于图中最多$|E|$条边的和
      3. 以每次都取最大流量的通路来迭代，迭代次数的上限为$O(|E|\,logF)$,F是最大流的结果
    - 证明：每次贪心至少可以减少$\frac{F_{last}}{|E|}$
      - 因此$F_t \le F(1-\frac{1}{E})^t < F(e^{-1/k})^t = Fe^{-t/k}$

> To learn more, see $cs170\ note\ \S7.2$

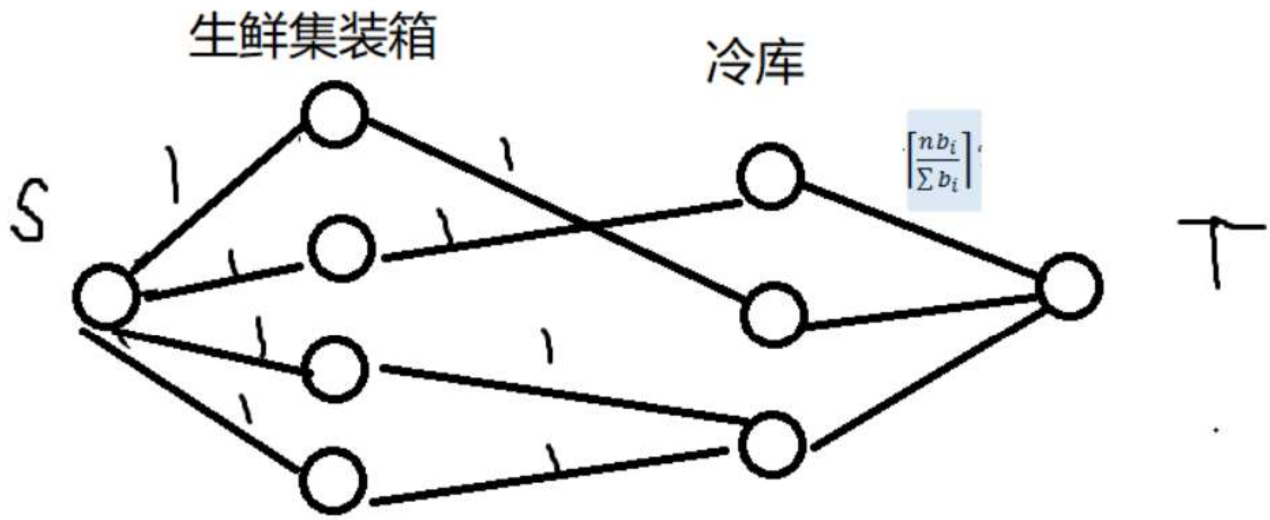
- 大题例题：18年期末

得分

五、网络流应用（20 分）

生鲜商品的物流问题，只有尽快把生鲜商品及时送到冷库才能最大限度的减少生鲜商品的变质。假设共有 k 个冷库，每个冷库 i 最多可以储存b_i个生鲜集装箱。现共有 n 个生鲜集装箱，每个集装箱 j 距离每个冷库 i 的距离以运送时间计算记为 t_{ij}。我们希望集装箱既能在 1 小时内送到冷库，又不能把一个冷库的储存空间用完导致无法储存后续出现的生鲜集装箱，因此需要尽可能地保持冷库空间占用率平均，不要过于集中。

1. 请设计一个最大流算法，判断是否可以将 n 个生鲜集装箱在 1 小时内送往 k 个冷库，并保证每个冷库 i 的存储空间占用率不超过$\left\lceil\frac{nb_i}{\sum b_i}\right\rceil$? 画出流网络图，说明源点、汇点、中间节点、边、容量、流量分别代表的含义。（10 分）



答案：如下图所示，建立第一层节点为生鲜集装箱，第二层节点为冷库。对于每个集装箱所对应的顶点，如果该集装箱能够在 1 小时内被送达某个冷库，那么将该集装箱顶点连边到该冷库节点。源点到集装箱节点的边的容量为 1，集装箱节点到冷库节点的边的容量也为 1，冷库节点到汇点的边的容量为第一问中所要求的那个比例，即$\left\lceil\frac{nb_i}{\sum b_i}\right\rceil$。然后运行网络流算法，如果求得的最大流值为 n，那么证明可以把所有集装箱在 1 小时内送到冷库，并且保证冷库的空间利用率绝对平均。如果小于 n，则不行。