

由图论问题浅析算法优化

武钢三中 贾由

图论问题

图论是数学的一个分支，它以图为研究对象，研究节点和边组成的图形的数学理论和方法。

图论问题与信息学竞赛联系紧密，经典的图论模型以及相关算法已成为竞赛中不可或缺的知识。



算法优化

基础图论知识

解决问题

原始算法

优化

优秀算法

针对性化

优化是一个逐步发现并利用问题的特殊之处、使算法更有针对性的过程。

例 · 二分图的最大匹配

■ 图的匹配：

图中任何两条边都没有共同顶点的子图。

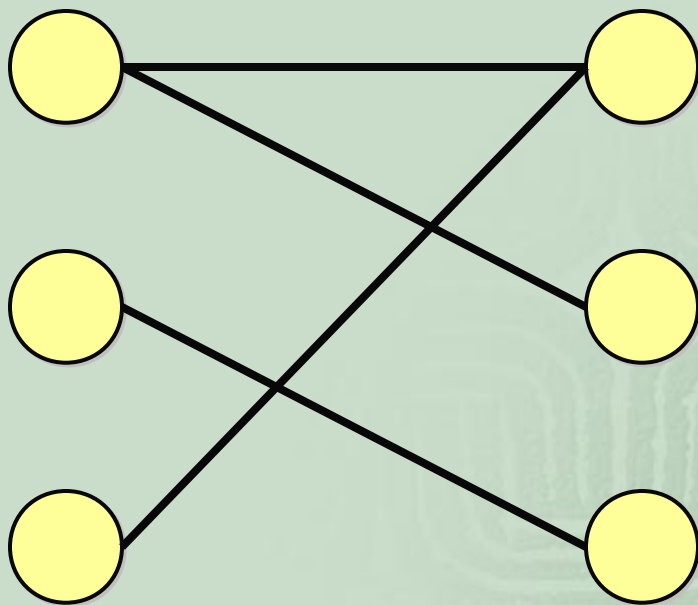
■ 二分图的最大匹配：

二分图中边数最多的匹配。



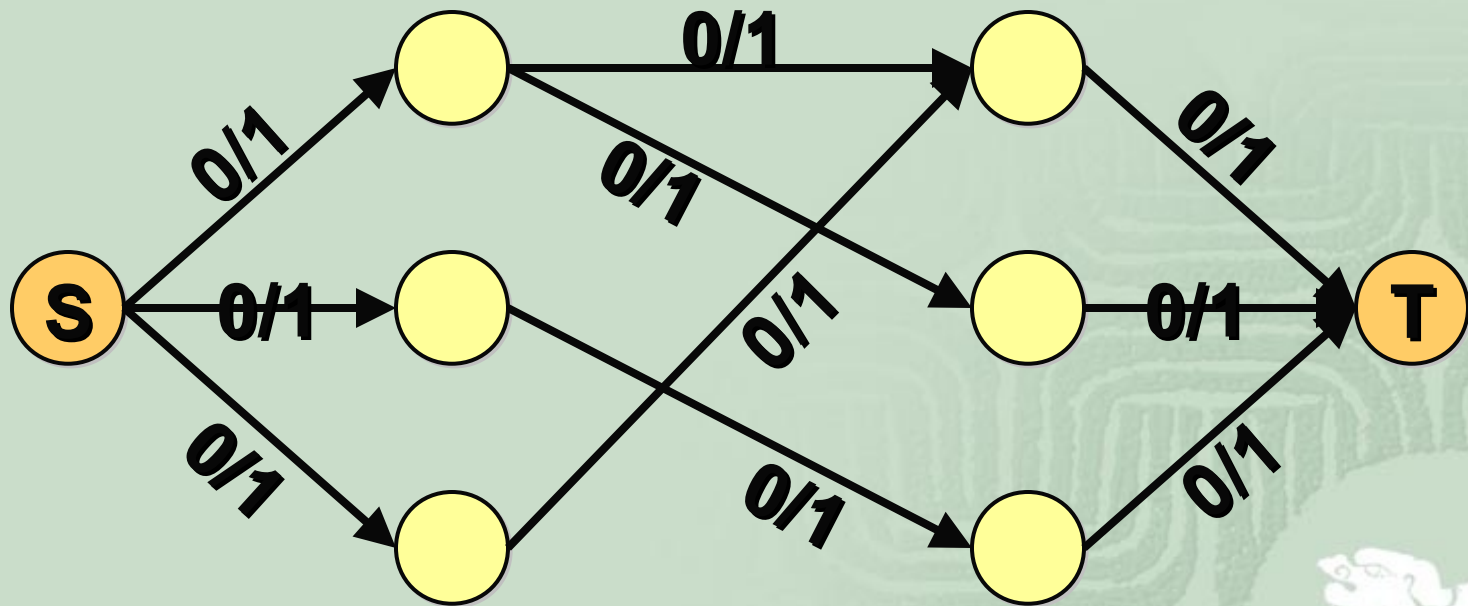
网络流模型

在二分图中加入源点、汇点，改为网络。



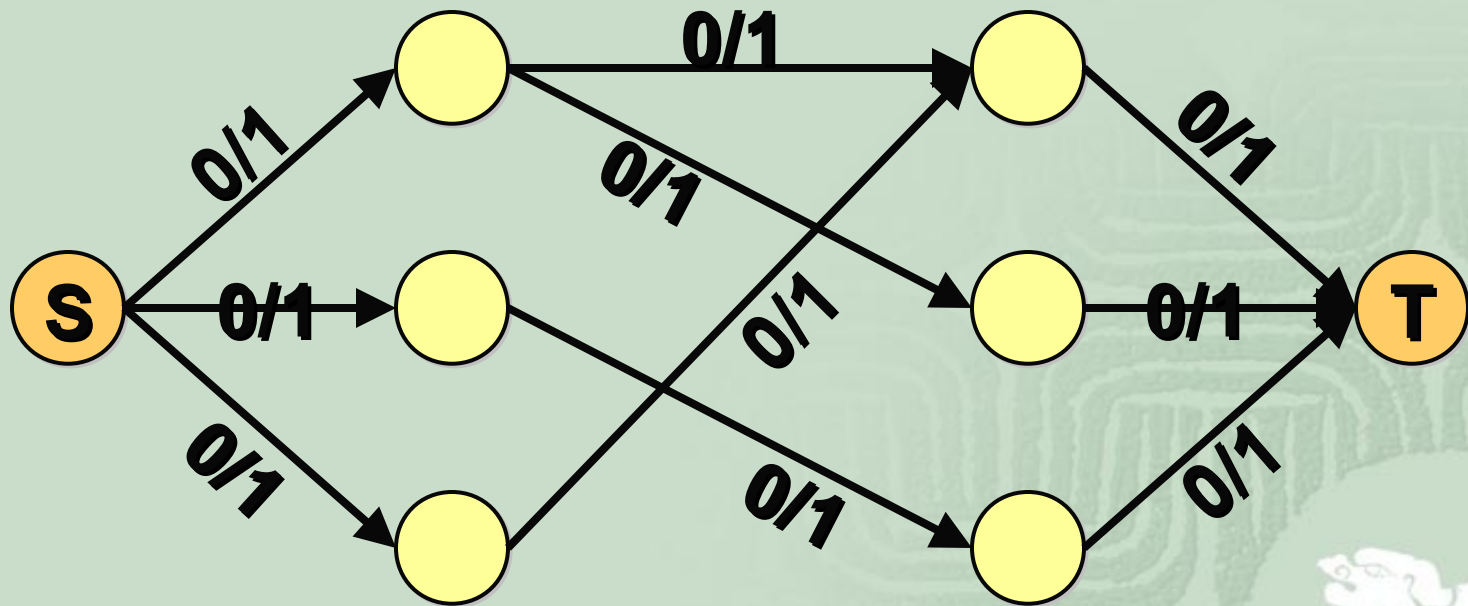
网络流模型

在二分图中加入源点、汇点，改为网络。



网络流算法

广搜可增广链



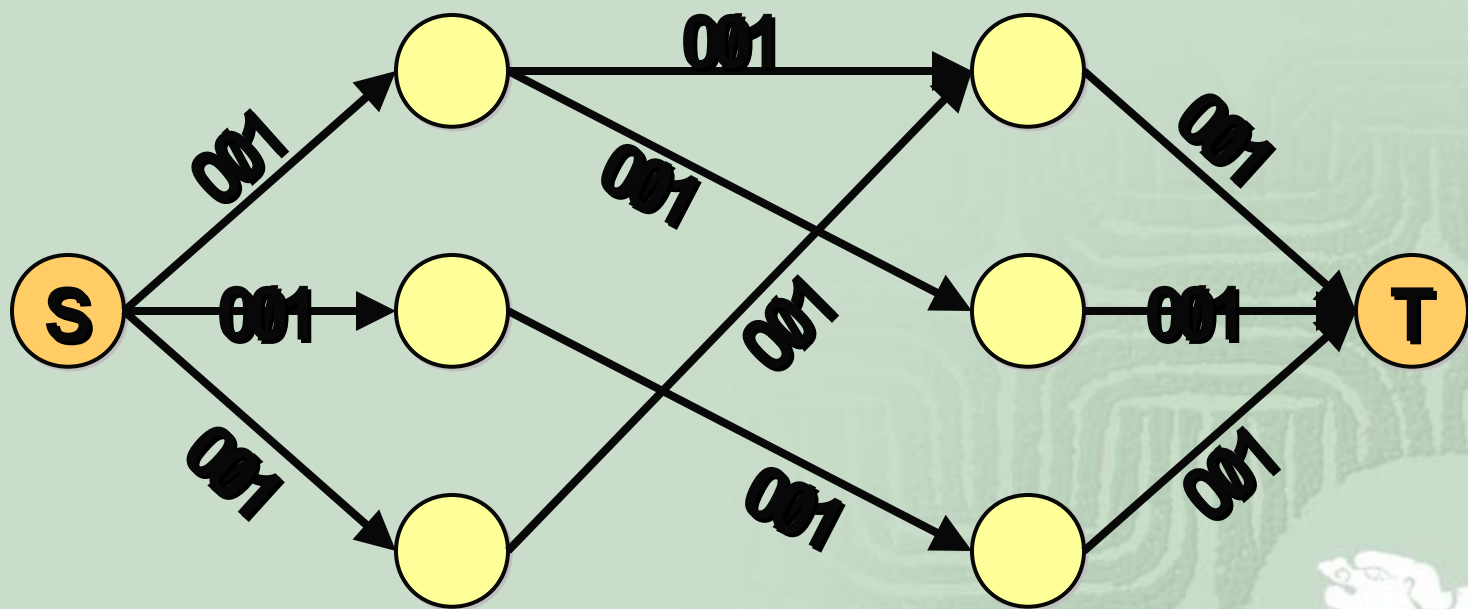
优化方法

a. 所有边的容量均为 1

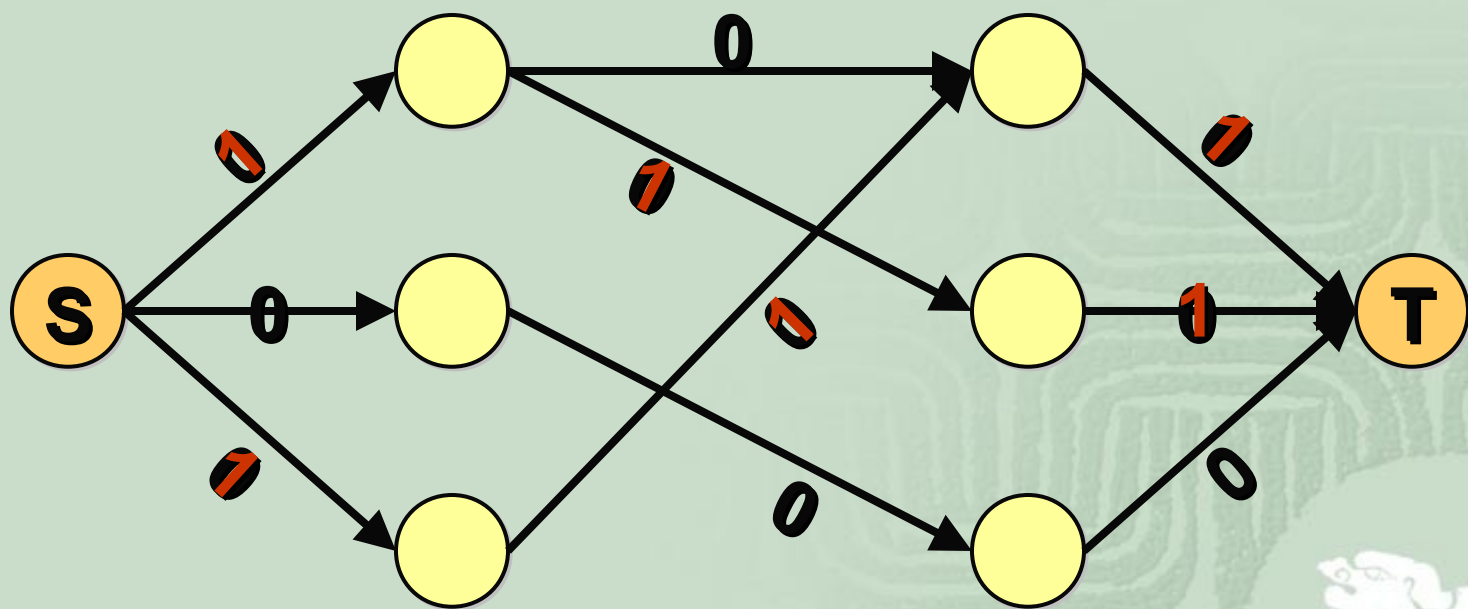
■ 不记录容量



优化方法



优化方法



优化方法

a. 所有边的容量均为 1

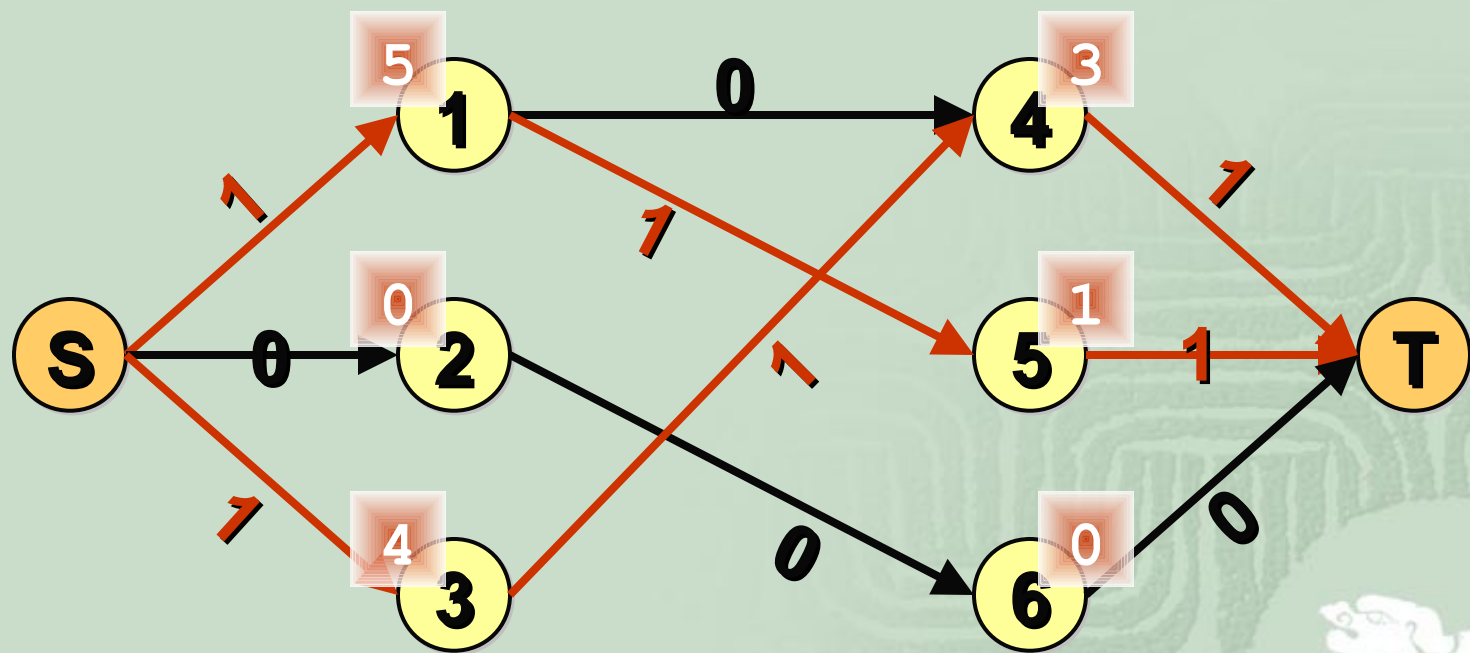
- 不记录容量

b. 与源点、汇点相连的边的流量可由其他边的流量推出

- 改为记录点的匹配情况



优化方法



优化方法

a. 所有边的容量均为 1

- 不记录容量

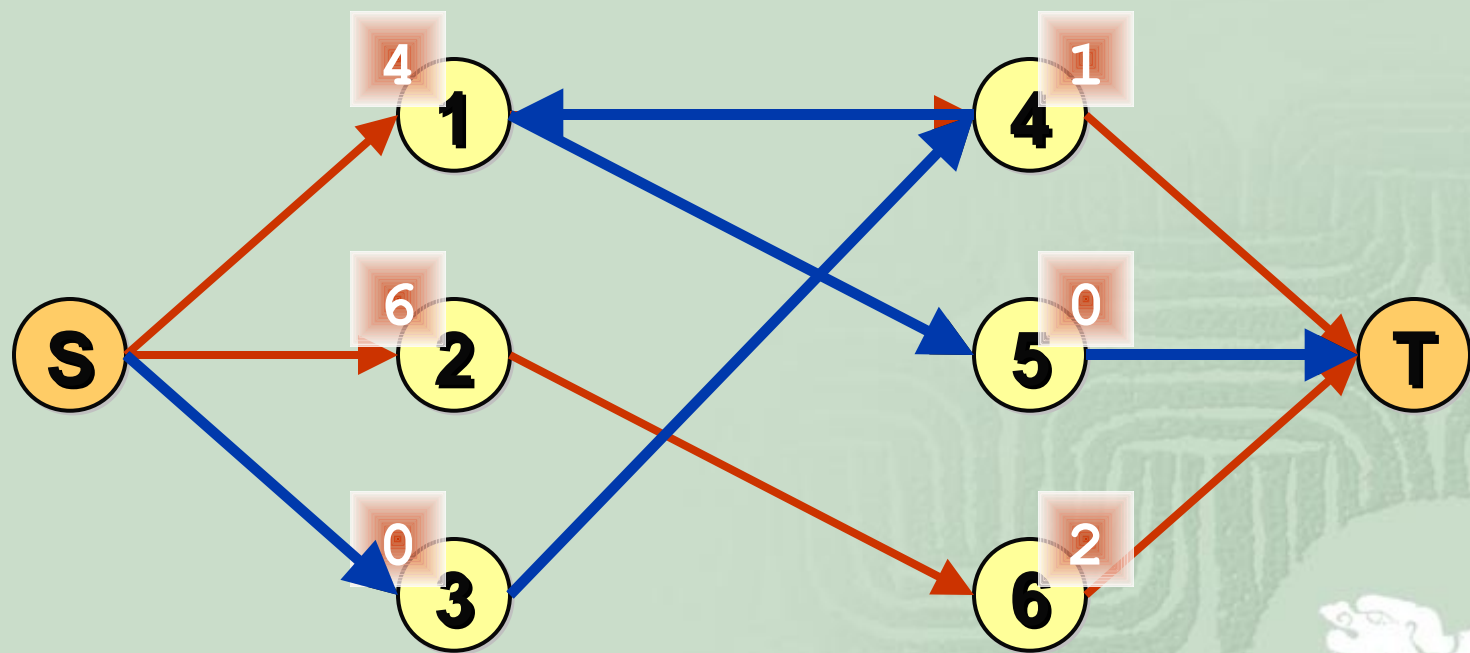
b. 与源点、汇点相连的边的流量可由其他边的流量推出

- 改为记录点的匹配情况

c. 可增广链结构特殊



优化方法



优化方法

a. 所有边的容量均为 1

- 不记录容量

b. 与源点、汇点相连的边的流量可由其他边的流量推出

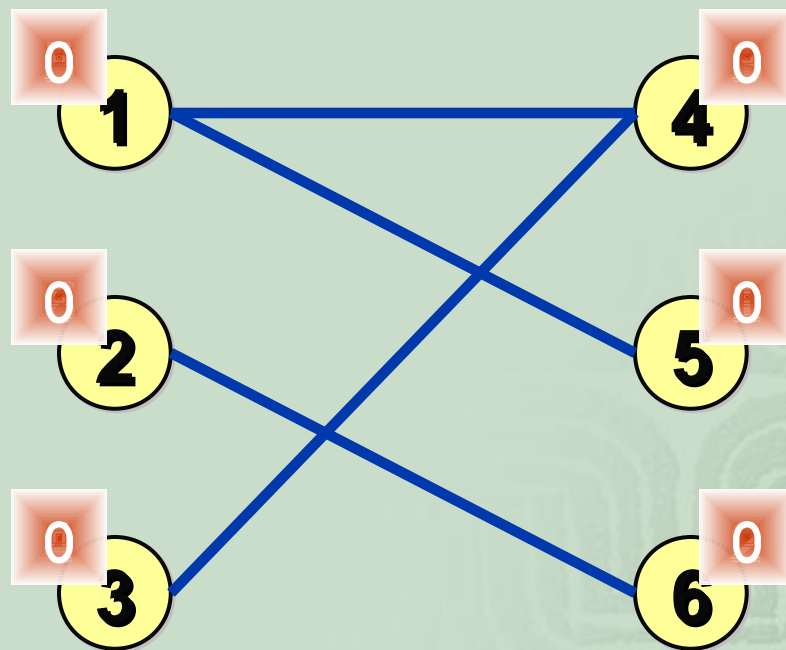
- 改为记录点的匹配情况

c. 可增广链结构特殊

- 只搜索中间部分



匈牙利算法



回顾

■ 容量特殊

○ 调整存储方式

减少存储空间

■ 网络结构特殊

□ 改进搜索算法

减少运行时间

设计网络流算法时的两个实用优化手段

小结

寻找特别之处 – 优化的根本途径

经验

耐心

灵感



优化方向

- 速度慢 时间复杂度
- 占用空间过大 空间复杂度
- 难以实现 编写难度
- 难以记忆 思维难度
- 不能完全解决问题 正确率



应对有缺陷的算法

- 面对难题时，我们难免会有意或无意地设计出有漏洞的算法
 - 简单处理、提高正确率
 - 分析问题、纠正错误
- 3. 放弃算法、另寻他解



快速处理

- 加入特殊判断过程
- 随机化 + 重复求解
 - 最优化问题
 - 取重复求解得到的最优值
 - 判定性问题
 - 正确率大于 50 %
 - 可以正确判断“是”、“否”中的一个方面
-



导致错误的原因

■ 误解模型的性质

《 Fishing Net 》

■ 猜想错误

《 Flying Right 》

■ 忽略算法细节

《 Cow Patterns 》



例 · Flying Right

一条航线上有 N 个机场，编号 1 到 N 。有 K 群牛等待乘坐飞机，第 i 群牛中有 M_i 头牛，它们要从 S_i 机场飞到 E_i 机场（ S_i 小于 E_i ）。

请问一架可承载 C 头牛的飞机从 1 号机场飞到 N 号机场最多可以把多少头牛送到目的地？

这当中，可以将一群牛拆散，只将其中的一部分带到目的地。

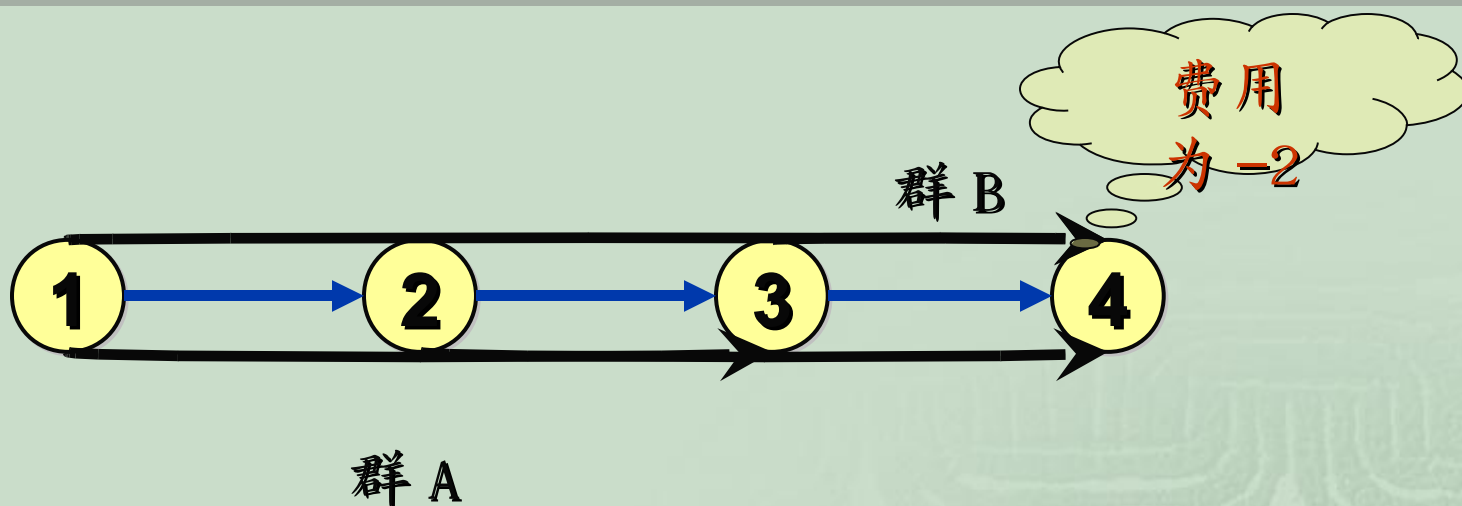
输入数据的规模

- 机场数： $1 \leq N \leq 10,000$
- 牛群数： $1 \leq K \leq 50,000$
- 每群中的牛数： $1 \leq M_i \leq C$
- 飞机容量： $1 \leq C \leq 100$

逐一考虑每个座位？



最小费用流



- 容量：边所对应的群中牛的个数
- 费用：-1（为了适应最小费用流）
- 为空闲的座位加入辅助边
 - 会容量无穷大、费用为零

每个单位流对应一个座位

最小费用流

- 算法的时间复杂度为 $O(K*N*C)$ ，无法承受题目给出的数据规模
- 另一方面，在这个十分特殊的图上直接套用最小费用流的算法未免有些浪费

如何优化？

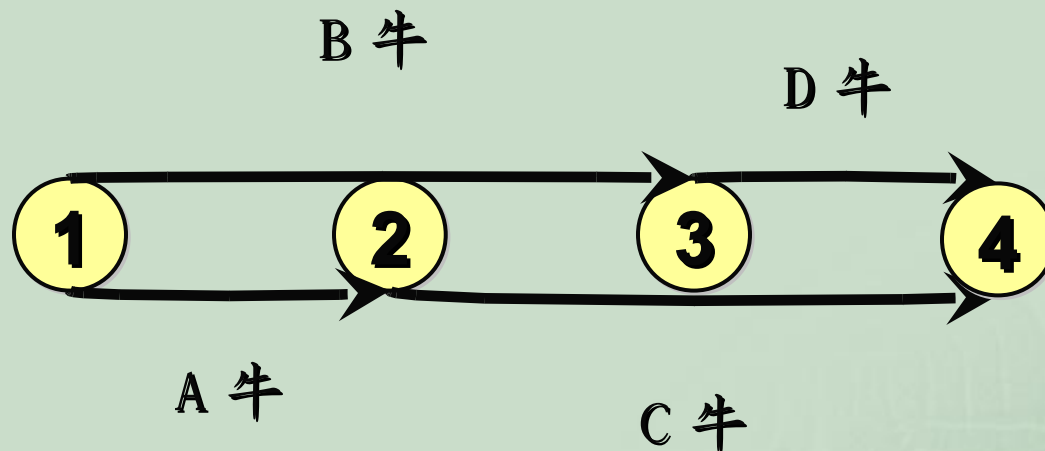


忽略后向边

- 寻找可增广链时考虑后向边使得后续过程可以调整已扩展的流，解决了后效性问题
- 忽略后向边相当于逐一为每个座位选择运送牛数最多的方案，选择后不再改动
- 用动态规划求解，时间复杂度为 $O(K \cdot C)$ ，可以接受



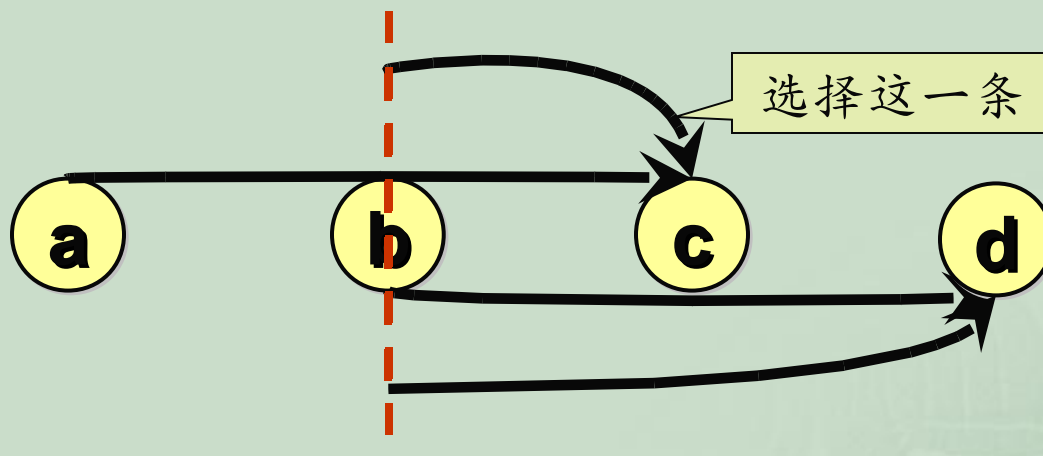
反例



- 假设四条边都只代表一头牛，而飞机上有两个座位
- 对当前座位“同样优”的两条边可能会对最终结果有不同的影响

需要进一步判断边的优劣关系

边的优劣关系



- 一条边含有两个参数： 设法减少一个
 起点、终点

----- 换个角度看问题 -----

- 假设飞机正停在某个机 □ 选择目的地最近的！
 场

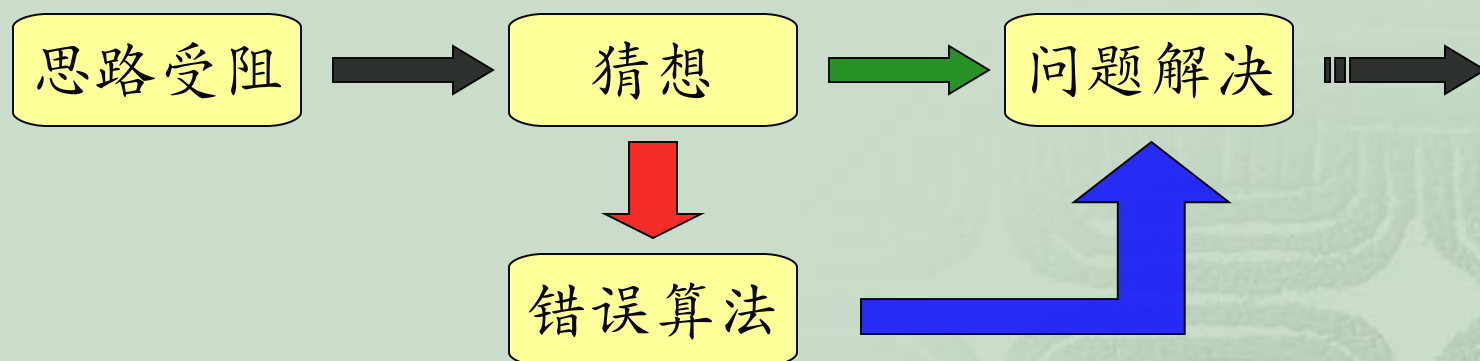
最终算法

- 将所有边按照终点排序
- 在每个机场携带目的地最近的 C 头牛继续飞行
- 直接实现的时间复杂度为 $O(N * C)$

问题解决



小结



分析错误的能力为大胆猜想提供了保障

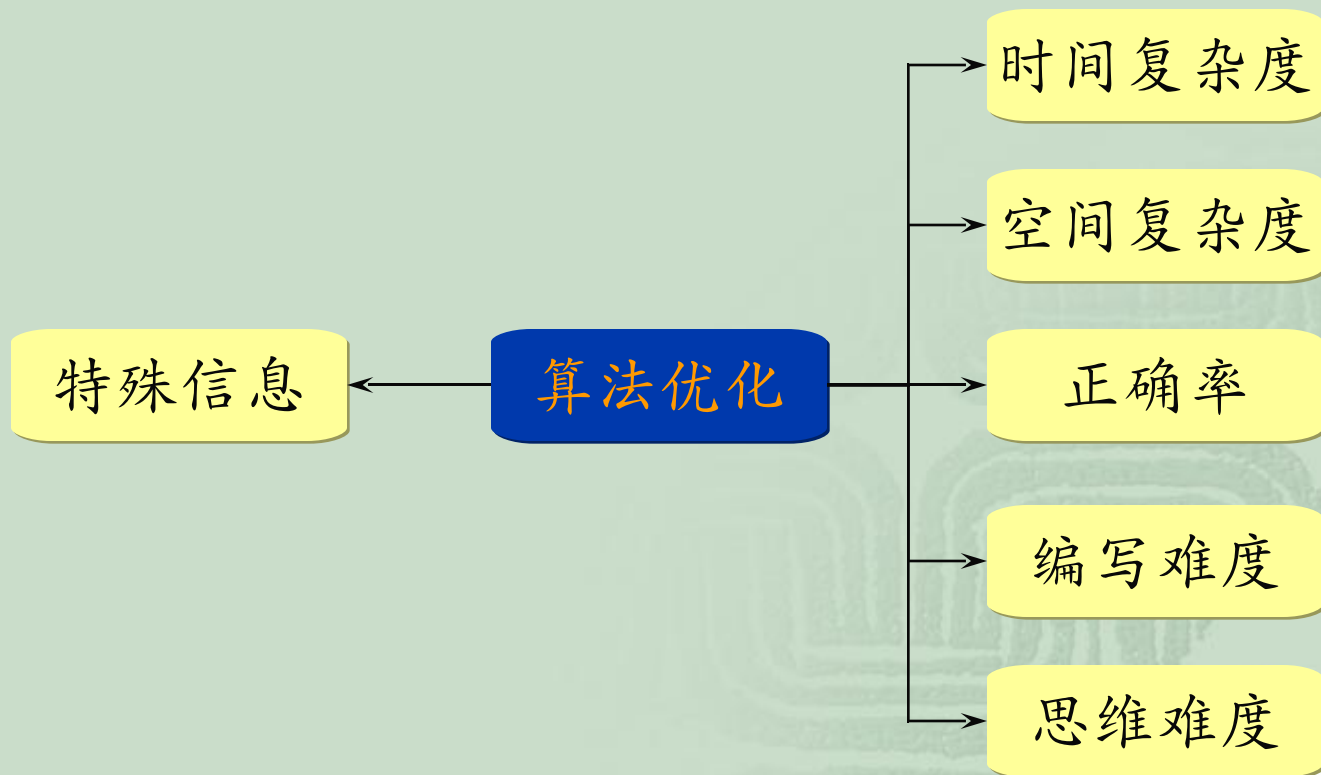
小结

提高算法的正确率是一种十分灵活的优化

解题过程更流畅



总结



结束语

每一次算法优化都是一次思维的旅程
无论结果怎样，思维都会有所收获





谢 谢