“分层图思想”及其在信息学竞赛中的应用

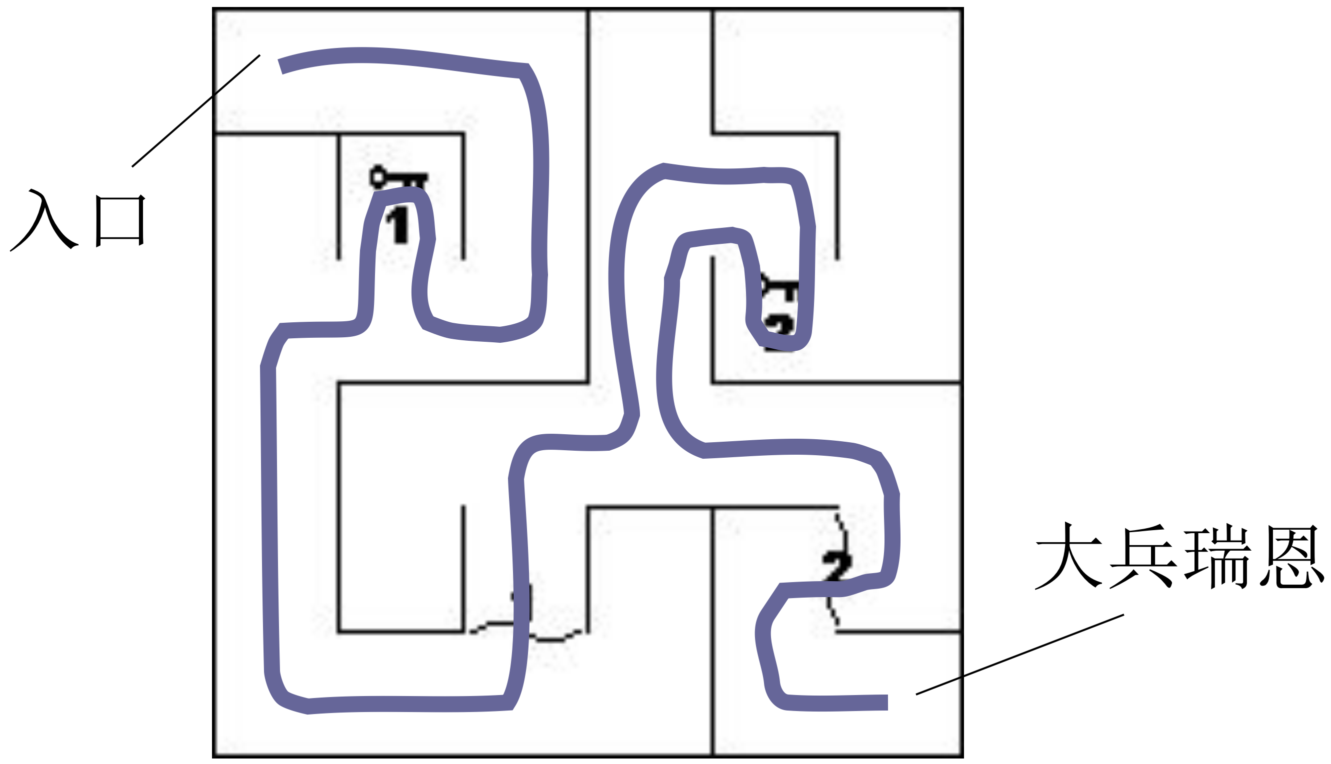
天津市南开中学　肖天

主要内容

- 这不是一个算法，而是一个建模思想
- 通过一个例题介绍该思想，并小结该思想的特点
- 应用该思想解决另一个例题，得到一个高效算法

例1：拯救大兵瑞恩（CTSC'99）

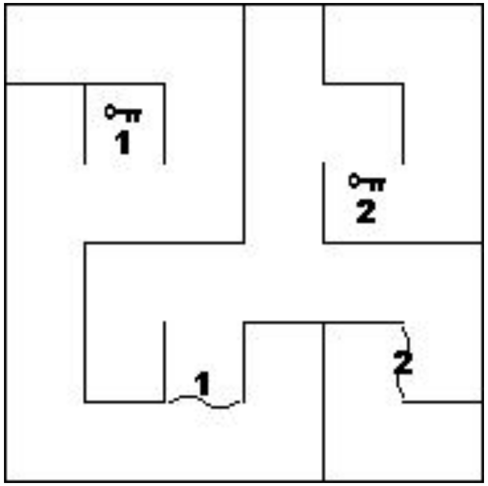
求从地图左上角到右下角的最少步数



例1：拯救大兵瑞恩（CTSC'99）

几点说明：

- 地图中共有P种钥匙（门），$P \le 10$
- 同种钥匙（或门）可能有多个



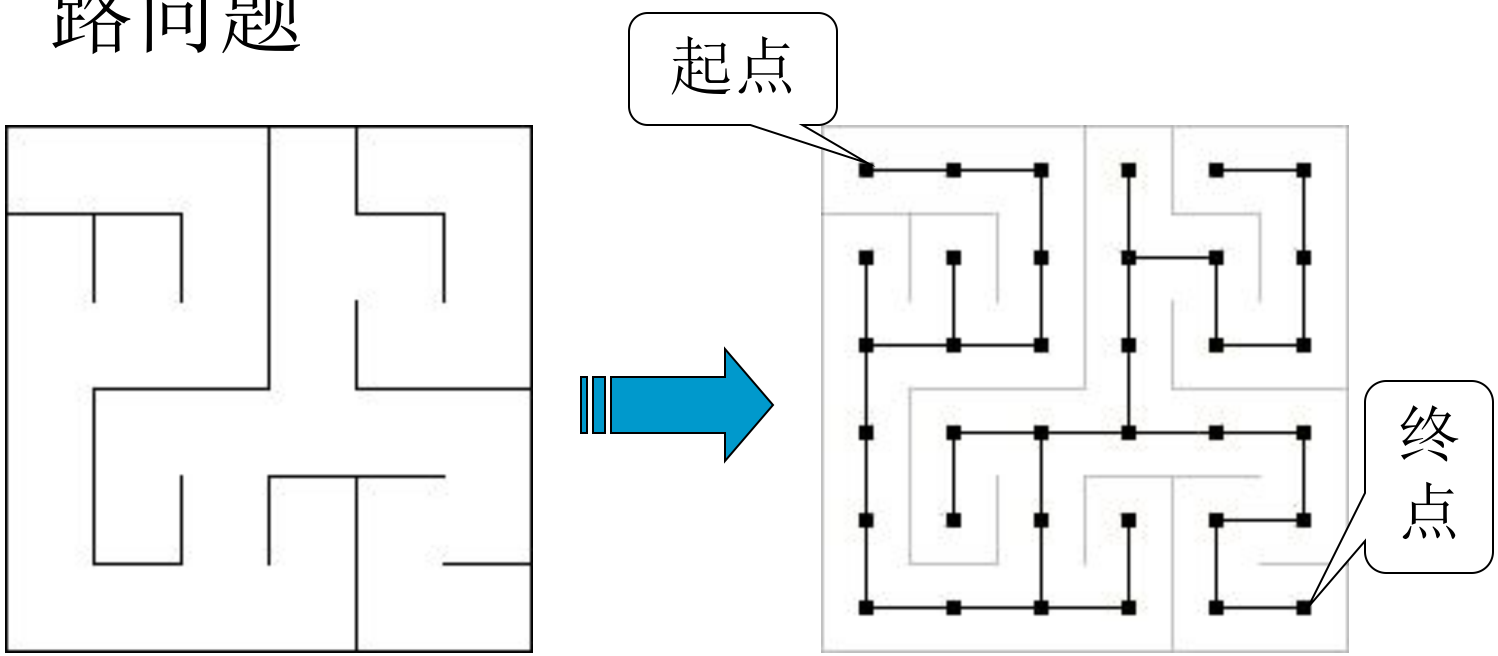
Rescue（CTSC’99）

问题的简化

先忽略钥匙和门的问题

BFS！

问题转化为在一个给定隐式图中的最短路问题



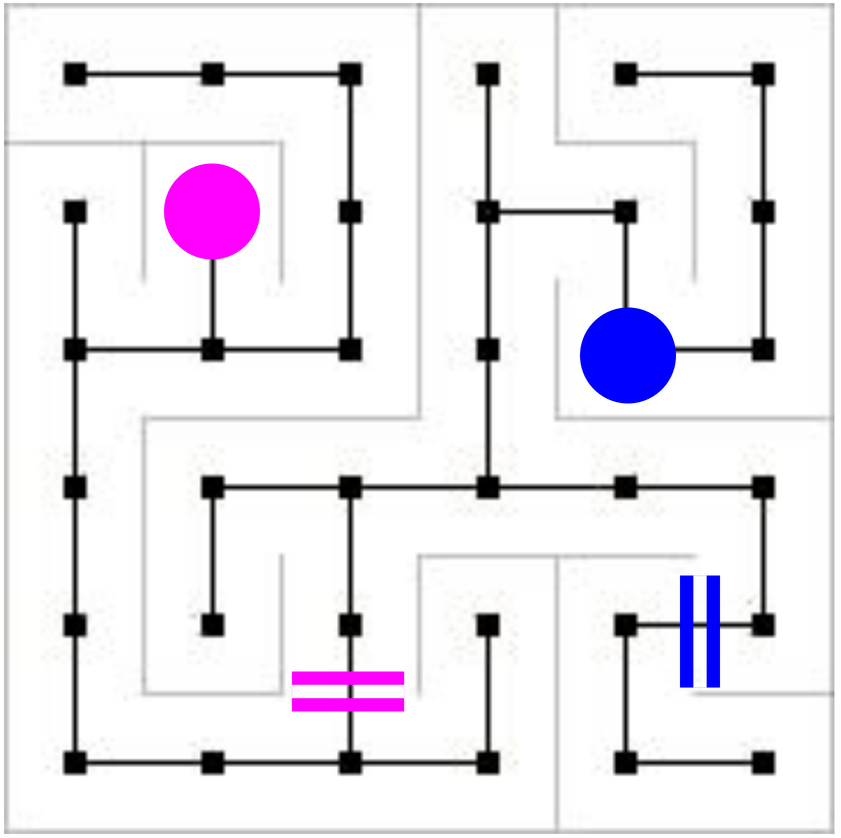
分析

加入钥匙和门的因素

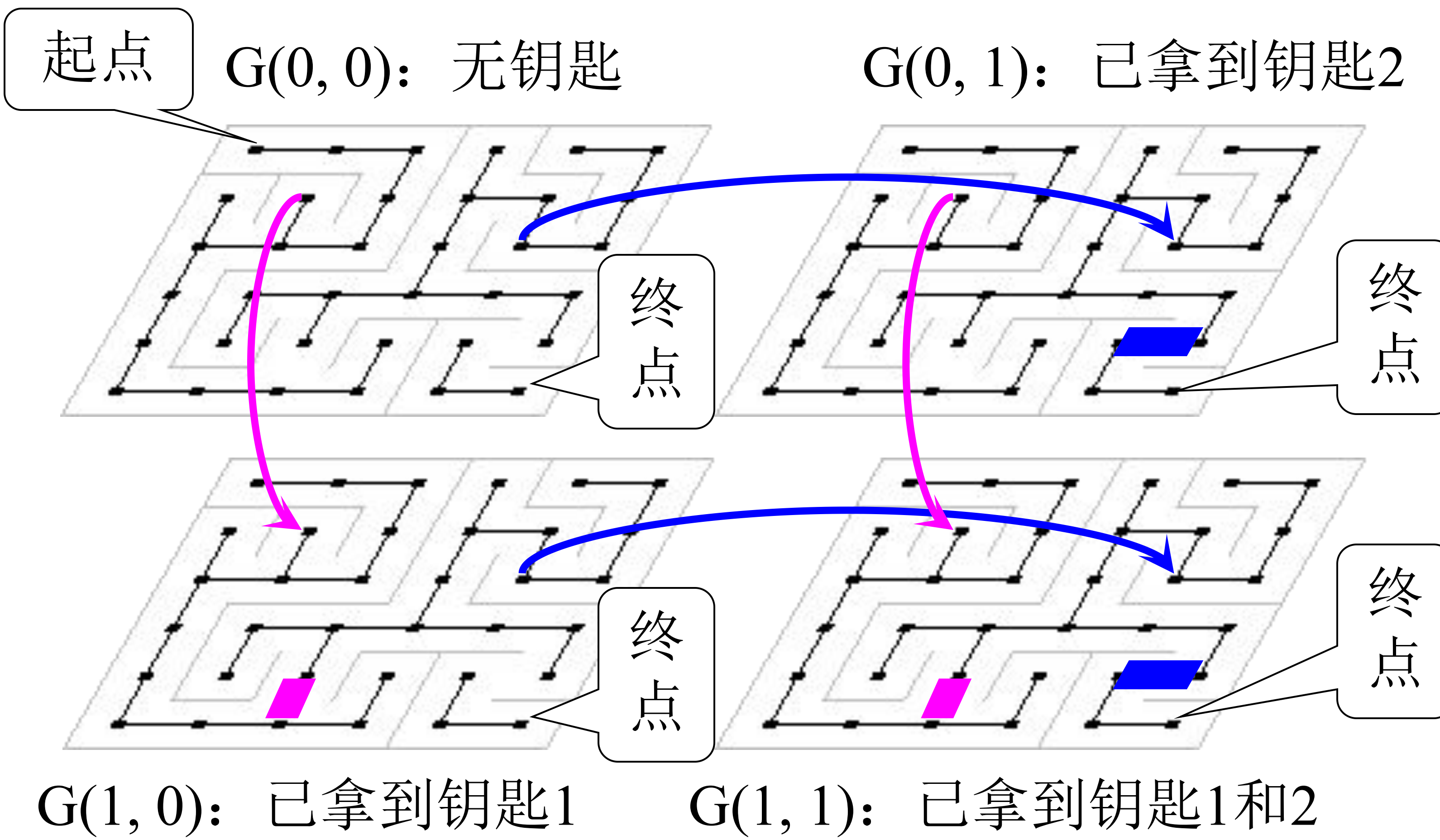
- 不能再简单地求最短路，因为通过某些边是有条件的（拿到相应钥匙）
- 需要考虑钥匙状态：

  0：未拿到该钥匙

  1：已拿到该钥匙



图的分层



小结：分层图的特点

- 分层消耗时间少
- 所有层都极为相似
- 所有的层是拓扑有序的

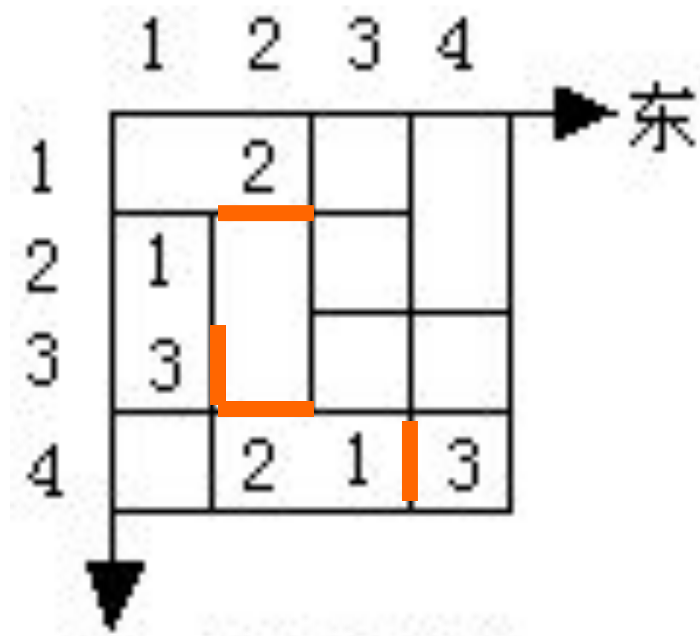
- 问题的规模并没有增大，而数学模型更清晰了

例2：迷宫改造（Winter Camp'99）

有一个$N*M$的长方形迷宫，其中假定有P个人，他们分别从P个指定的起点出发，要求他们只能向南或向东移动，分别到达P个指定的终点

问至少拆掉多少堵墙

（这是原问题的一部分）



迷宫示例

例2：迷宫改造（Winter Camp'99）

- 参数限定

  $N = M\ (\le 20)$

  $\cancel{1 \le P \le 3}$

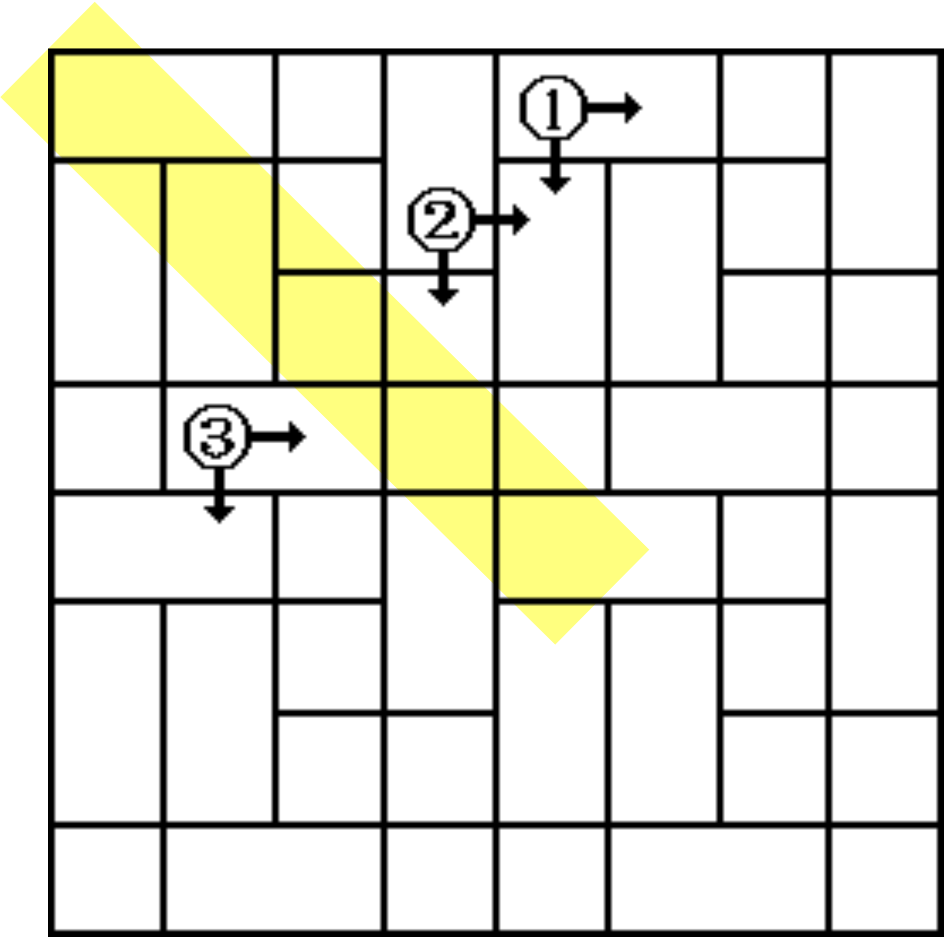
  增加起点与终点重合的人使 $P = 3$

解法1：动态规划

- 以平行于副对角线的斜线划分阶段
- 状态描述

  斜线位置

  三个人的位置
- 细节



解法1：动态规划

- 状态数：$O(N^4)$
- 每个状态的状态转移方案：≤ 8

- 时间复杂度：$O(N^4)$
- 空间复杂度：$O(N^3)$

好像大了点儿……

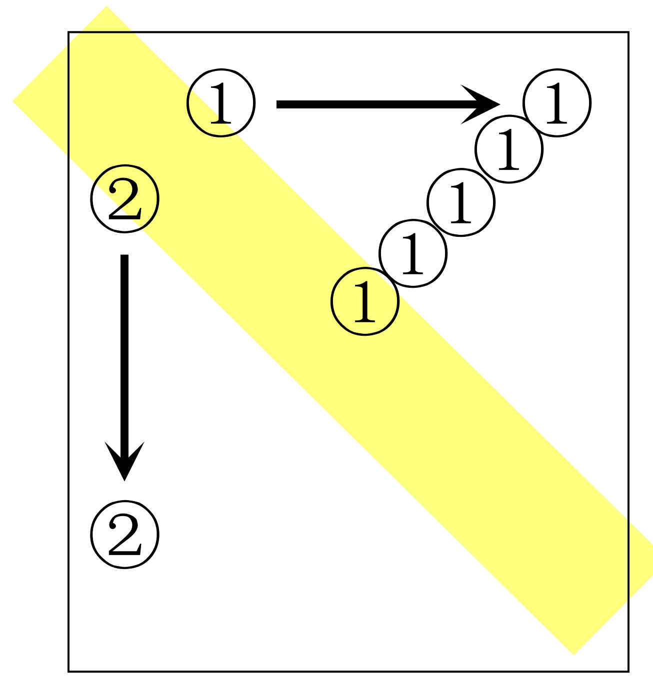
解法1：动态规划 — 分析

如右图：只考虑两次P=1的情况即可

②的情况被重复计算了

仍有冗余计算！

- 原因
  - 两人之间只在有公共路线时才有关系
  - 其它时候，原则上只需分别考虑

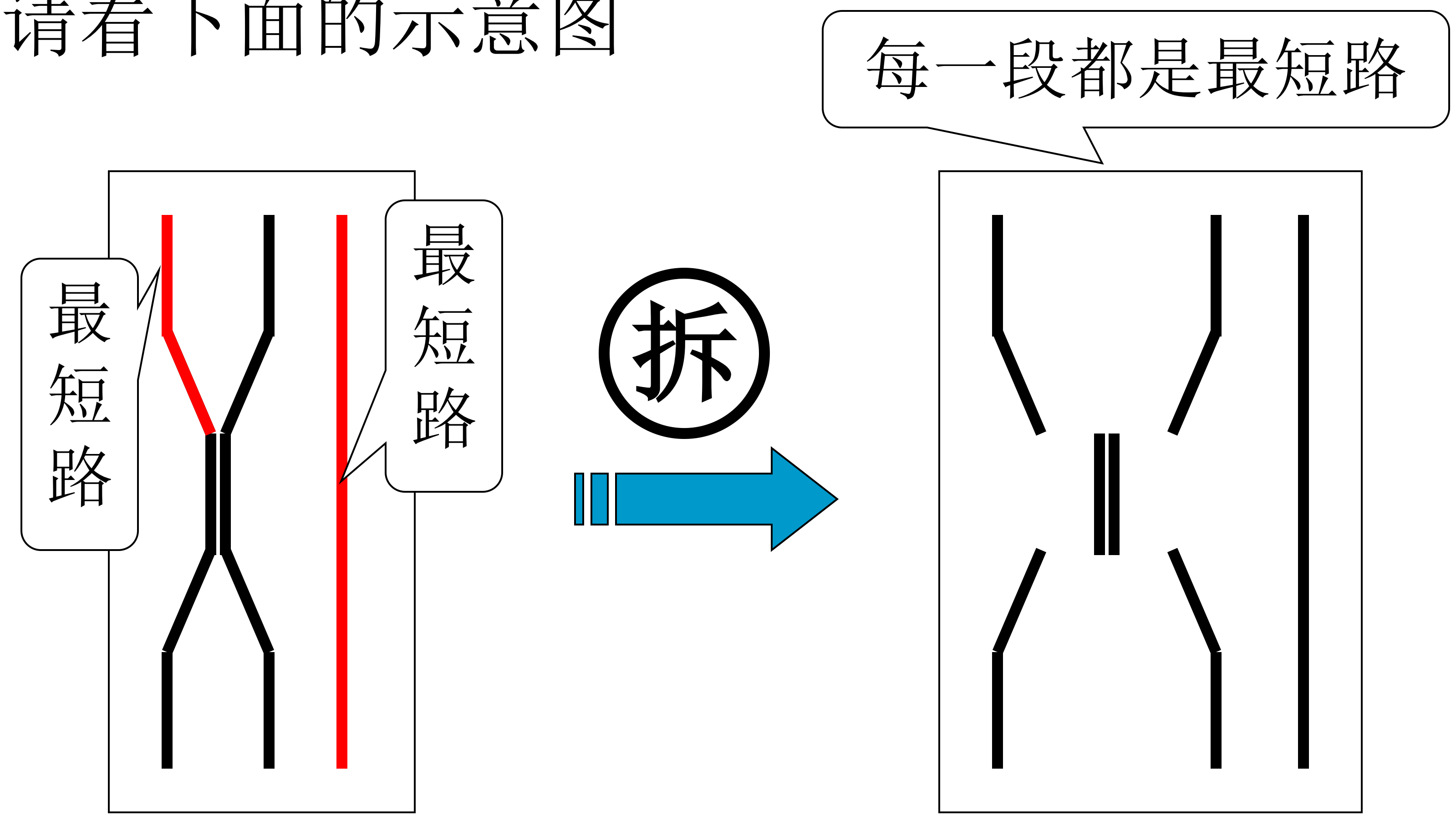


挖掘问题特点

- 某人路线的**最优性**受其他人影响，但是
- 如果某人与其他人没有公共路线，那么他的路线是最优的当且仅当此路线是他起点与终点之间的最短路
- 如果某人路线中*A*点与*B*点之间的部分与其他人没有公共部分，且该路线是最优的，那么*AB*段一定是*A*点与*B*点之间的最短路

挖掘问题特点

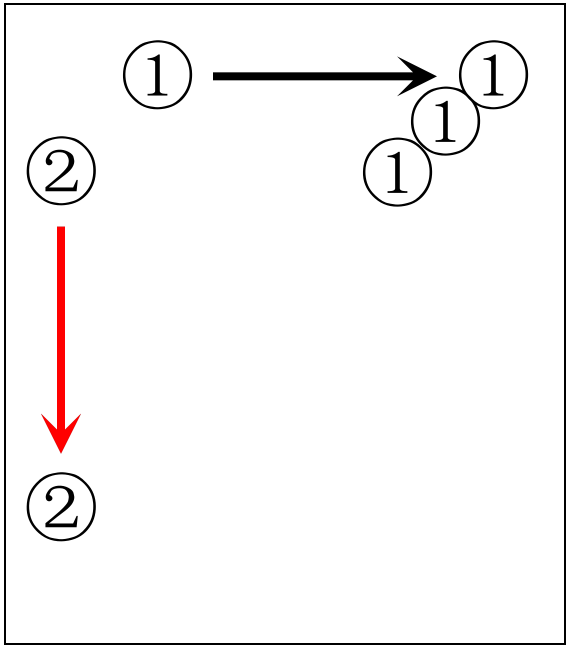
请看下面的示意图



试图消除冗余

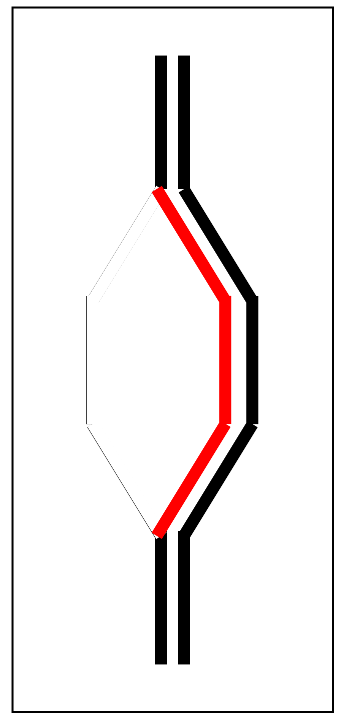
- ②的路线是最短路，只需计算一次
- 通过预处理求出任意两点间的最短路
- 时间复杂度O(N^4)

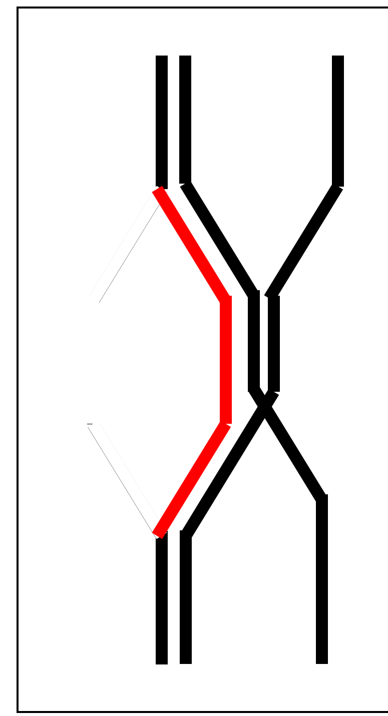
失败



继续挖掘问题特点

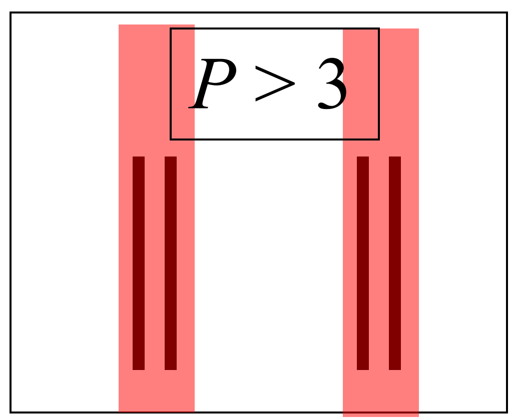
- 两人有**公共路线**时，代价只计算一次
- 两人路线之并集没有环
- 三人路线之并集没有环

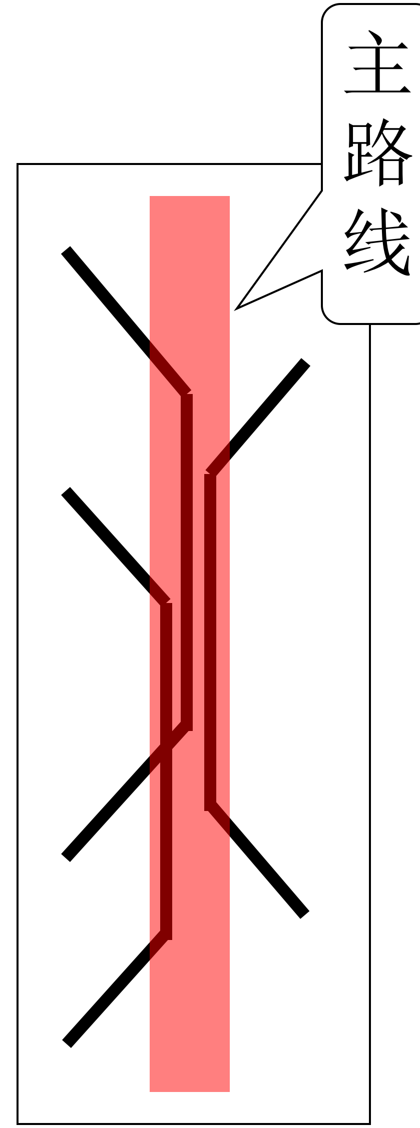




继续挖掘问题特点

- 假想有一个人SuperMan要从左上角**走**到右下角，要求是他要走过所有的公共路线
- 这个SuperMan一定能找到满意的路线，我们称该路线为主路线
- 反证法：

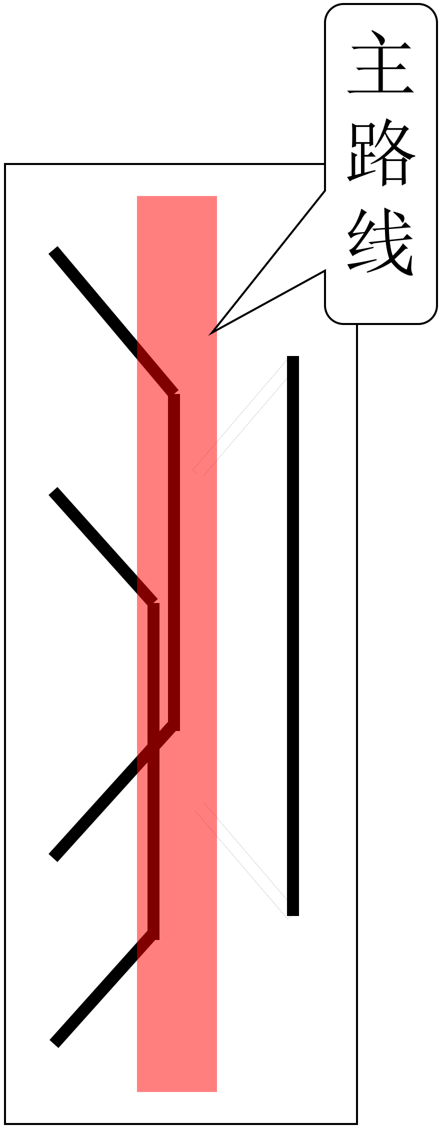




继续挖掘问题特点

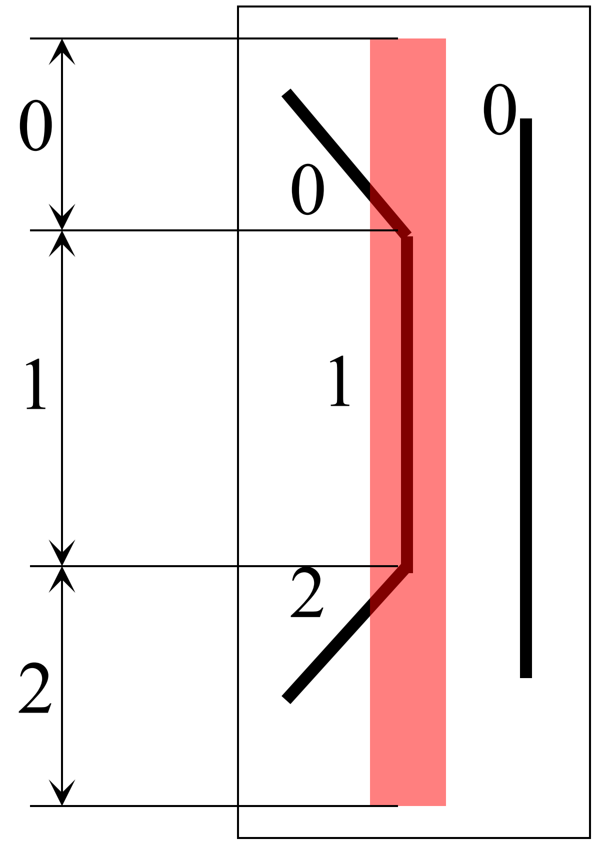
- 有人可能根本不进入主路线
- 主路线之外的部分都是最短路，预处理时间复杂度O(N^2)

求“最短”的主路线



解法2：应用“分层图思想”

- 每个人有三个状态：
  - 0：未进入主路线
  - 1：正在主路线中
  - 2：已离开主路线
- 换一个角度，主路线对每个人同样有三个状态：0，1，2

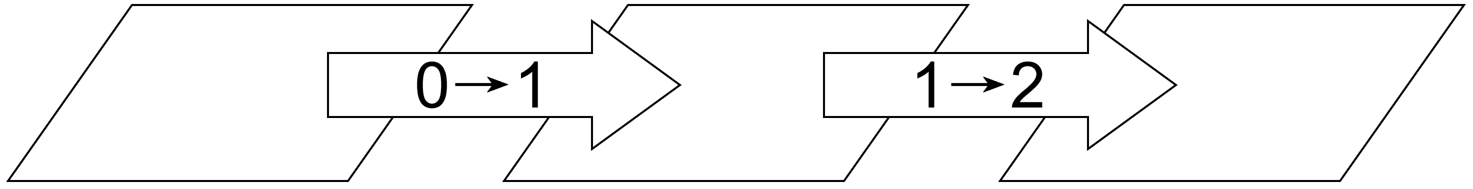


解法2：应用“分层图思想”

- 将原图复制$3^3=27$份，记为$G(s_1, s_2, s_3)$，其中$s_i \in \{0, 1, 2\}$表示第 i 个人的状态
- 在相邻层的所有对应顶点间加边，权为：

  $0 \to 1$：从起点到该点的最短路长度

  $1 \to 2$：从该点到终点的最短路长度



解法2：应用“分层图思想”

- 求解以$G(0, 0, 0)$左上角顶点为起点的单源最短路问题。
- 终点不是$G(2, 2, 2)$右下角顶点，而是所有满足$s_i \in \{0, 2\}$的层$G(s_1, s_2, s_3)$的右下角顶点

解法2：应用“分层图思想”

$$Ans = \min\begin{cases} d(0,0,0) & +p(1) & +p(2) & +p(3) \\ d(0,0,2) & +p(1) & +p(2) & \\ d(0,2,0) & +p(1) & & +p(3) \\ d(0,2,2) & +p(1) & & \\ d(2,0,0) & & +p(2) & +p(3) \\ d(2,0,2) & & +p(2) & \\ d(2,2,0) & & & +p(3) \\ d(2,2,2) & & & \end{cases}$$

总结

- 干扰因素使问题的模型变得模糊
- 将干扰因素细化为若干状态——分层
- 将状态联系起来——层的连接
- 找到算法

谢

谢！