



中国科学院大学

University of Chinese Academy of Sciences

## 计算机算法设计与分析

083500M01001H

### Chap 6&7 课程作业解答

2022 年 11 月 8 号

*Professor:* 刘玉贵



学生: 周胤昌

学号: 202228018670052

学院: 网络安全学院

所属专业: 网络安全

方向: 安全协议理论与技术

## Problem 1

假设对称旅行商问题的邻接矩阵如下所示, 试用优先队列式分枝限界算法给出最短环游. 画出状态空间树的搜索图, 并说明搜索过程.

$$\begin{pmatrix} \infty & 20 & 30 & 10 & 11 \\ & \infty & 16 & 4 & 2 \\ & & \infty & 6 & 7 \\ & & & \infty & 12 \\ & & & & \infty \end{pmatrix}$$

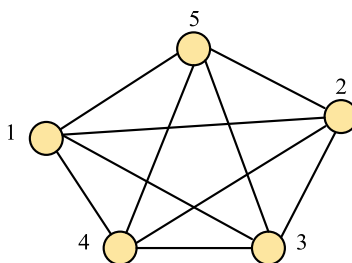


图 1: 旅行商问题

**Solution:** 状态空间树的搜索图如下图2中所示:

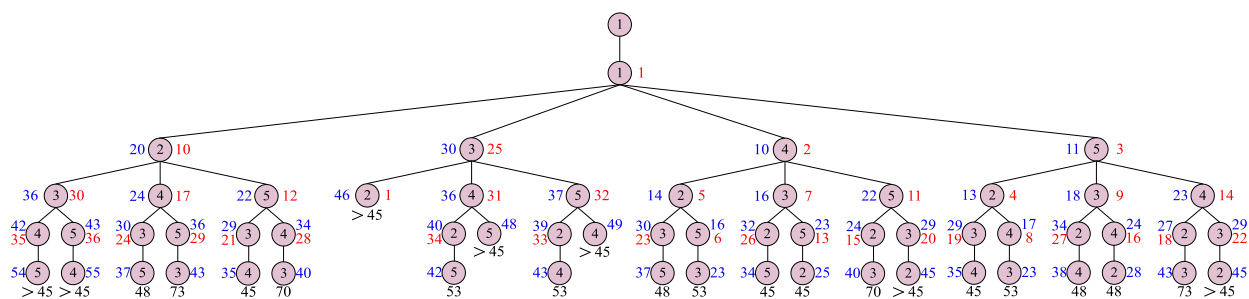


图 2: 解空间树搜索图

图中圆圈内为顶点序号, 蓝色数字为该路径到该节点的总路程, 红色数字表示该节点的搜索序数. 从顶点 1 开始搜索, 将其 4 个儿子节点放入队列. 然后优先访问队列中当前路程最小的节点, 再将其儿子放入队列. 然后重复上述过程, 优先访问队列中当前路径最小的节点, 将其儿子放入队列. 第一个被访问到的叶子节点为 1-4-2-5-3-1 这条路径, 总路程为 53, 记录当前最短路径. 之后若某节点的路径大于最短路径, 则不再搜索该节点的子树. 若某叶子节点的总路程小于当前最短路径, 则更新之. 如此搜索, 当访问到 1-4-3-5-2-1 这条路径的叶子节点时, 其总路程为  $45 < 53$ , 将 45 更新为当前最短路径, 继续搜索. 最后得出最短环游为 1-2-5-3-4-1、1-4-3-2-5-1、1-4-3-5-2-1、1-5-2-3-4-1, 总路程均为 45.

## Problem 2

最佳调度问题: 假设有  $n$  个任务要由  $k$  个可并行工作的机器来完成, 完成任务  $i$  需要的时间为  $t_i$ . 试设计一个分枝限界算法, 找出完成这  $n$  个任务的最佳调度, 使得完成全部任务的时间 (从机器开始加工任务到最后停机的时间) 最短.

**Solution:** 设计的算法步骤如下:

- 先将任务由大到小排序;
- 计算  $n$  个任务需要的总时间和平均到  $k$  个机器上的时间;
- 将大于平均时间的任务各分配一个机器, 找到最大完成时间;
- 将其他任务顺序安排在一台机器上, 如果时间超出最大时间, 则把该任务交给下一个机器, 下一个任务继续在这台机器上试安排, 直到所有任务都不能在小于最大完成时间的情况下安排;
- 安排下一台机器直到所有任务安排完;
- 或有可能安排某些任务找不到小于最大完成时间, 那么重新扫描各台机器使再加上该任务后时间最小, 按此方法安排完所有任务.

## Problem 3

分派问题: 给  $n$  个人分派  $n$  件工作, 给第  $i$  人分派第  $j$  件工作的成本是  $C(i, j)$ , 试用分枝限界法求成本最小的工作分配方案.

**Solution:** 设  $n$  个人的集合是  $\{1, 2, \dots, n\}$ ,  $n$  项工作的集合是  $\{1, 2, \dots, n\}$ , 每个人恰好 1 项工作. 于是有

$$\text{把工作 } j \text{ 分配给 } i \Leftrightarrow x_i = j, \quad i, j = 1, 2, \dots, n$$

设解向量为  $X = \langle x_1, x_2, \dots, x_n \rangle$ , 分配成本为  $C(X) = \sum_{i=1}^n C(i, x_i)$ . 搜索空间是排列树. 部分向量  $\langle x_1, x_2, \dots, x_k \rangle$  表示已经考虑了人  $1, 2, \dots, k$  的工作分配. 节点分支的约束条件为:

$$x_{k+1} \in \{1, 2, \dots, n\} \setminus \{x_1, x_2, \dots, x_k\}$$

可以设立代价函数:

$$F(x_1, x_2, \dots, x_k) = \sum_{i=1}^k C(i, x_i) + \sum_{i=k+1}^n \min \{C(i, t) : t \in \{1, 2, \dots, n\} \setminus \{x_1, x_2, \dots, x_k\}\}$$

界  $B$  是已得到的最好可行解的分配成本. 如果代价函数大于界, 则回溯. 算法的时间复杂度为  $O(n \cdot n!)$ .

## Problem 4

如图3所示, 一个4阶 Latin 方是一个  $4 \times 4$  的方格, 在它的每个方格内填入 1, 2, 3 或 4, 并使得每个数字在每行、每列都恰好出现一次. 用回溯法求出所有第一行为 1, 2, 3, 4 的所有4阶 Latin 方. 将每个解的第2行到第4行的数字从左到右写成一个序列. 如图3中的解是  $\langle 3, 4, 1, 2, 4, 3, 2, 1, 2, 1, 4, 3 \rangle$ . 给出所有可能的4阶 Latin 方.

1	2	3	4
3	4	1	2
4	3	2	1
2	1	4	3

图 3: Latin 方

通过 PPT 中的回溯算法可以求出共有 24 个 Latin 方, 具体如下图4所示:

1 2 3 4 2 1 4 3 3 4 1 2 4 3 2 1	1 2 3 4 2 1 4 3 3 4 2 1 4 3 1 2	1 2 3 4 2 1 4 3 4 3 1 2 3 4 2 1	1 2 3 4 2 1 4 3 4 3 2 1 3 4 1 2	1 2 3 4 2 3 4 1 3 4 1 2 4 1 2 3	1 2 3 4 2 3 4 1 4 1 2 3 3 4 1 2
1 2 3 4 2 4 3 1 3 1 4 2 4 3 2 1	1 2 3 4 2 4 1 3 4 3 2 1 3 1 4 2	1 2 3 4 3 1 4 2 2 4 1 3 4 3 2 1	1 2 3 4 3 1 4 2 4 3 2 1 2 4 1 3	1 2 3 4 3 4 1 2 2 1 4 3 4 3 2 1	1 2 3 4 3 4 1 2 2 3 4 1 4 1 2 3
1 2 3 4 3 4 1 2 4 1 2 3 2 3 4 1	1 2 3 4 3 4 1 2 4 3 2 1 2 1 4 3	1 2 3 4 3 4 2 1 2 1 4 3 4 3 1 2	1 2 3 4 3 4 2 1 4 3 1 2 2 1 4 3	1 2 3 4 4 1 2 3 2 3 4 1 3 4 1 2	1 2 3 4 4 1 2 3 3 4 1 2 2 3 4 1
1 2 3 4 4 3 1 2 2 1 4 3 3 4 2 1	1 2 3 4 4 3 1 2 3 4 2 1 2 1 4 3	1 2 3 4 4 3 2 1 2 1 4 3 3 4 1 2	1 2 3 4 4 3 2 1 2 4 1 3 3 1 4 2	1 2 3 4 4 3 2 1 3 1 4 2 2 4 1 3	1 2 3 4 4 3 2 1 3 4 1 2 2 1 4 3

图 4: Latin 方结果