

数学模型与数学软件

第 10 次作业

1907402030

熊 雄^{*}



2022 年 5 月 27 日

^{*}mrxiong@foxmail.com 苏州大学数学科学学院本科生

Problem 1

(Page 244 Ex.4.)

某货运公司需要从 9 个货运订单中选定一些订单作为一批用一个集装箱发送, 以获得最大利润. 该集装箱的最大装载容积 (不允许重复叠放, 所以这里以底面积表示) 为 $1000m^2$, 最大装载重量为 $1200kg$. 9 个货运订单的相关信息见表 10.5.

表10.5

订单号	1	2	3	4	5	6	7	8	9
利润	71	6	3	6	33	13	110	21	49
空间	67	27	794	53	234	32	792	97	435
重量	774	76	22	42	21	760	818	62	785

Solution.

• 模型建立

设总利润为 W . 由于对于每个订单 x_i , $i = 1, \dots, 9$ 只能选择运送/不运送, 因此本问题显然是一个 0-1 规划问题. 我们需要求得最大利润为:

$$\max W = 71x_1 + 6x_2 + 3x_3 + 6x_4 + 33x_5 + 13x_6 + 110x_7 + 21x_8 + 49x_9. \quad (1)$$

约束条件为

$$\begin{cases} 67x_1 + 27x_2 + 794x_3 + 53x_4 + 234x_5 + 32x_6 + 792x_7 + 97x_8 + 435x_9 \leq 1000, \\ 774x_1 + 76x_2 + 22x_3 + 42x_4 + 21x_5 + 760x_6 + 818x_7 + 62x_8 + 785x_9 \leq 1200. \end{cases} \quad (2)$$

• 代码求解

由于是 0-1 规划问题, 在编写程序时需要对每个 x_i , $i = 1, \dots, 9$, 加上 @bin(xi) 的命令. 输入以下 Lingo 代码:

```

1 model:
2 max = 71 * x1 + 6 * x2 + 3 * x3 + 6 * x4 + 33 * x5 + 13 * x6 + 110 * x7 +
   21 * x8 + 49 * x9 ;
3 67 * x1 + 27 * x2 + 794 * x3 + 53 * x4 + 234 * x5 + 32 * x6 + 792 * x7 + 97
   * x8 + 435 * x9 <= 1000;
4 774 * x1 + 76 * x2 + 22 * x3 + 42 * x4 + 21 * x5 + 760 * x6 + 818 * x7 + 62
   * x8 + 785 * x9 <= 1200;
5 @bin(x1); @bin(x2); @bin(x3); @bin(x4); @bin(x5); @bin(x6); @bin(x7); @bin(
   x8); @bin(x9) ;

```

运行后得到输出如下:

```
1 Global optimal solution found.
2 Objective value: 143.0000
3 Objective bound: 143.0000
4
5
6 Variable Value Reduced Cost
7 X1 0.000000 -71.00000
8 X2 1.000000 -6.000000
9 X3 0.000000 -3.000000
10 X4 1.000000 -6.000000
11 X5 0.000000 -33.00000
12 X6 0.000000 -13.00000
13 X7 1.000000 -110.0000
14 X8 1.000000 -21.00000
15 X9 0.000000 -49.00000
```

• 结果分析

因此, 最佳安排方式为运送订单 2,4,7,8, 从而可以获得最高利润, 最高利润为 143 个单位. ■

Problem 2

(Page 244 Ex.5.)

(指派问题) 考虑指定 n 个人完成 n 项任务 (每人单独承担一项任务), 使所需的总完成时间 (成本) 尽可能短. 已知某指派问题的有关数据 (每人完成各任务所需的时间) 如表 10.6 所示, 试求解该指派问题.

表10.6

工人	任务			
	1	2	3	4
1	15	18	21	24
2	19	23	22	18
3	26	18	16	19
4	19	21	23	17

Solution.

一、利用动态规划方法计算求解

• 建立模型

设 $f_k(i)$ 为工人 i 完成任务 k 需要的时间, 其中 $k = 1, 2, 3, 4, i = 1, 2, 3, 4$. 设

$$g_k(i) = \begin{cases} 1, & \text{若工人 } i \text{ 参与第 } k \text{ 个任务,} \\ 0, & \text{若工人 } i \text{ 不参与第 } k \text{ 个任务.} \end{cases}$$

设完成任务的总时间 (成本) 为 T . 则该问题可以用整数规划描述如下:

$$\begin{cases} \min T = \sum_{k=1}^4 \sum_{i=1}^4 f_k(i) g_k(i), \\ \text{s. t. } \sum_{k=1}^4 g_k(i) = 1, \quad i = 1, 2, 3, 4. \end{cases} \quad (3)$$

- 共 4 个任务, 可以把问题分解为 **4 个阶段**: 在任意阶段 $k (= 4, 3, 2, 1)$, 把任务 k 分配给第 $i (= 4, 3, 2, 1)$ 个工人.
- **状态变量** x_k 可以选为: x_k 表示第 k 个阶段还未参加任务的工人. 由题意知 $x_1 = \{1, 2, 3, 4\}, x_5 = \emptyset$.
- **决策变量** u_k 可以选为: u_k 表示第 k 个阶段使得 $g_k(i) = 1$ 的工人 i . 因此状态转移方程为 $x_{k+1} = x_k - u_k$.
- 阶段 k 的**准则函数**为 $v(x_k, u_k) = \sum_{i=1}^4 f_k(i) g_k(i)$.

用 $F_k(x_k)$ 表示将工人集合 x_k 依次分配给任务 4, 3, 2, 1 时的最短时间, 则有如下的

动态规划基本方程

$$\begin{cases} F_k(x_k) = \min_{0 \leq u_k \leq x_k} \left[\sum_{i=1}^4 f_k(i) g_k(i) + F_{k+1}(x_{k+1}) \right], & x_{k+1} = x_k - u_k, \\ F_5(x_5) = \emptyset, & k = 4, 3, 2, 1. \end{cases} \quad (4)$$

边界条件: $F_5(x_5) = F_5(\emptyset) = 0$.

• 计算求解

— $k = 4$ 时,

$$F_4(x_4) = \min_{0 \leq u_4 \leq x_4} \left[\sum_{i=1}^4 f_4(i) g_4(i) + F_5(x_5) \right] = \sum_{i=1}^4 f_4(i) g_4(i).$$

$$F_4(\{1\}) = 19, F_4(\{2\}) = 21, F_4(\{3\}) = 19, F_4(\{4\}) = 17.$$

— $k = 3$ 时,

$$F_3(x_3) = \min_{0 \leq u_3 \leq x_3} \left[\sum_{i=1}^4 f_3(i) g_3(i) + F_4(x_4) \right] = \min_{0 \leq u_3 \leq x_3} \left[\sum_{i=1}^4 f_3(i) g_3(i) + F_4(x_3 - u_3) \right].$$

$$F_3(\{1, 2\}) = 37, F_3(\{1, 3\}) = 35, F_3(\{1, 4\}) = 38,$$

$$F_3(\{2, 3\}) = 37, F_3(\{2, 4\}) = 35, F_3(\{3, 4\}) = 33.$$

— $k = 2$ 时,

$$F_2(x_2) = \min_{0 \leq u_2 \leq x_2} \left[\sum_{i=1}^4 f_2(i) g_2(i) + F_3(x_3) \right] = \min_{0 \leq u_2 \leq x_2} \left[\sum_{i=1}^4 f_2(i) g_2(i) + F_3(x_2 - u_2) \right].$$

$$F_2(\{1, 2, 3\}) = 56, F_2(\{1, 2, 4\}) = 54, F_2(\{1, 3, 4\}) = 52, F_2(\{2, 3, 4\}) = 55.$$

— $k = 1$ 时,

$$F_1(x_1) = \min_{0 \leq u_1 \leq x_1} \left[\sum_{i=1}^4 f_1(i) g_1(i) + F_2(x_2) \right] = \min_{0 \leq u_1 \leq x_1} \left[\sum_{i=1}^4 f_1(i) g_1(i) + F_2(x_1 - u_1) \right]$$

$$F_1(\{1, 2, 3, 4\}) = 70.$$

• 结果分析

由以上分析求解知, 总时间至少为 **70 分钟**, 此时可以有两个方案:

a) 工人 1 完成任务 1, 工人 2 完成 4, 工人 3 完成任务 3, 工人 4 完成任务 2;

b) 工人 1 完成任务 2, 工人 2 完成 1, 工人 3 完成任务 3, 工人 4 完成任务 4.

二、利用 0-1 规划与 Lingo 求解

• 模型建立

每个人只能有两种状态: 安排工作/不安排工作, 因此本问题是一个 0-1 规划问题. 需要的最小总时间是:

$$\min T = 15x_{11} + 18x_{12} + 21x_{13} + 24x_{14} + 19x_{21} + 23x_{22} + 22x_{23} + 18x_{24} + 26x_{31} + 18x_{32} + 16x_{33} + 19x_{34} + 19x_{41} + 21x_{42} + 23x_{43} + 17x_{44}. \quad (5)$$

由于一个人只能从事一个工作, 并且一个工作只能由一个人完成. 因此还有如下约束条件:

$$\begin{cases} x_{11} + x_{21} + x_{31} + x_{41} = 1, \\ x_{12} + x_{22} + x_{32} + x_{42} = 1, \\ x_{13} + x_{23} + x_{33} + x_{43} = 1, \\ x_{14} + x_{24} + x_{34} + x_{44} = 1, \\ x_{11} + x_{12} + x_{13} + x_{14} = 1, \\ x_{21} + x_{22} + x_{23} + x_{24} = 1, \\ x_{31} + x_{32} + x_{33} + x_{34} = 1, \\ x_{41} + x_{42} + x_{43} + x_{44} = 1. \end{cases} \quad (6)$$

• 代码求解

由于是 0-1 规划问题, 在编写程序时需要对每个 x_{ij} , $i, j = 1, \dots, 4$, 加上 @bin(xi) 的命令.

输入以下 Lingo 代码:

```

1 model:
2  min=15 * x11 + 18 * x12 + 21 * x13 + 24 * x14 + 19 * x21 + 23 * x22 + 22 *
      X23 + 18 * x24 + 26 * x31 + 18 * x32 + 16 * x33 + 19 * x34 + 19 *
      x41 + 21 * x42 + 23 * x43 + 17 * x44;
3  x11 + x21 + x31 + x41 = 1;
4  x12 + x22 + x32 + x42 = 1;
5  x13 + x23 + x33 + x43 = 1;
6  x14 + x24 + x34 + x44 = 1;
7  x11 + x12 + x13 + x14 = 1;
8  x21 + x22 + x23 + x24 = 1;
9  x31 + x32 + x33 + x34 = 1;
10 x41 + x42 + x43 + x44 = 1;
11 @bin(x11); @bin(x12); @bin (x13); @bin(x14); @bin(x21); @bin(x22); @bin(x23);
      @bin(x24);
12 @bin(x31); @bin (x32); @bin(x33); @bin(x34); @bin(x41); @bin(x42); @bin(x43)
      ; @bin(x44);

```

运行后得到输出如下:

1	Global optimal solution found.		
2	Objective value:	70.00000	
3	Objective bound:	70.00000	
4			
5	Variable	Value	Reduced Cost
6	X11	0.000000	0.000000
7	X12	1.000000	18.00000
8	X13	0.000000	21.00000
9	X14	0.000000	24.00000
10	X21	1.000000	4.000000
11	X22	0.000000	23.00000
12	X23	0.000000	22.00000
13	X24	0.000000	18.00000
14	X31	0.000000	11.00000
15	X32	0.000000	18.00000
16	X33	1.000000	16.00000
17	X34	0.000000	19.00000
18	X41	0.000000	4.000000
19	X42	0.000000	21.00000
20	X43	0.000000	23.00000
21	X44	1.000000	17.00000
22	XL1	0.000000	0.000000

• 结果分析

因此, 工人 1 完成任务 2, 工人 2 完成 1, 工人 3 完成任务 3, 工人 4 完成任务 4, 这样安排可以使得总时间最小, 最小值为 70 分钟.

Remark: 利用 0-1 规划和 Lingo 程序求解所得的解决方案只有一种, 不如利用动态规划法求解. ■