

研究巡检路线的排班状况及优化问题

摘要

在确保某工厂能正常运行的情况下，以减少人力资源为目的，让工人生产力得到充分发挥且每名工人的工作量尽可能均衡，确定巡检人员数量，制定恰当的工作时间表和工作路线图。

针对问题一：以时间最短为目标函数建立多目标优化模型，采用 0-1 规划进行建立，先利用 excel 对附件的数据进行处理，借助 lingo 软件运行，结合人工对数据的整理，得出要完成该任务每班需要 5 个工人巡检较为理想，该 5 个工人具体巡检时间（见表 6-1 至表 6-5）和巡检路线（如图 6-2 至图 6-6）。

针对问题二：在问题一的基础上满足巡检工人 2 小时左右休息一次，因固定上班时时间，三班倒，则假设三个班在固定时间进餐，不考虑进餐时间，以时间最少为目标函数，增加约束条件，建立 0-1 规划，利用 lingo 软件运行以及对数据的整理，得到每班需要 6 个工人巡检较为理想，其巡检时间（见表 6-7 至表 6-12）和巡检路线（如图 6-9 至图 6-14）。

针对问题三：在问题一和问题二的基础上，采用错时上班，从而增加目标函数，对其建立 0-1 规划模型，对问题一及问题二再次分别进行讨论，得出错时上班的问题一每班安排 4 人巡检合理，错时上班的问题二每班安排 5 人巡检合理，根据相关数据对比，可见错时上班更节省人力资源。

关键词

0-1 规划

排班

LINGO

EXCEL

1. 问题重述

1.1 情况说明

为保障某化工厂正常运行，需对 26 个位号进行巡检，每个巡检点要一个工人巡检，且每个工人巡检起始点为位号 XJ0022，每个班的上班时间可以固定也可错时，在能完成巡检任务的情况下，尽可能减少人力资源且使每个工人达到工作量均衡。

1.2 相关信息

附件 (shift1) :各个位号的周期与巡检耗时的基本信息

附件 (shift2) :两个位点之间的连通以及行走的耗时数据

附件 (shift3) :各个位点间的连通图

1.3 需要解决的问题

(1) 固定上班时间，三班倒，预测每班需要多少人，并呈现出巡检时间表以及巡检路线图。

(2) 不固定上班时间，三班倒，每个工人工作量达到 2 小时左右，需要休息 5 至 10 分钟，并且在中午 12 点及下午 6 点左右进餐，进餐时间为半个小时，预测每班需要多少人，并呈现出巡检时间表以及巡检路线图。

(3) 在问题一和问题二的基础上，错时上班，再分别讨论问题一及问题二，进行比较，分析得出采用哪种上班方式更能减少人力资源。

2. 问题分析

2.1 问题一分析

问题一在固定时间上班，不涉及巡检人员的休息时间的情况下，采用三班倒，每班每天工作 8 小时左右，且尽量保障每名工人工作量平衡，且每个巡检点在 8 小时左右都能按时完成巡检任务，为了让每个工人能充分发挥生产力，结合题意假设每个工人工作量之差的绝对值相差 10 分钟，同时考虑上个巡检点到下个巡检点的时间之和刚好等于下个巡检点的周期，以上均作为限制条件。为降低人力资源消耗，以时间最短（即所用工人最少）建立目标函数^[1]，欲借助 excel、linggo 软件进行数据处理和优化结果。因此为解决此问题，方便讨论，以一个班为基准建立 0-1 规划模型，得出满足限制条件的最优安排工人人数、巡检时间表和巡检路线图。

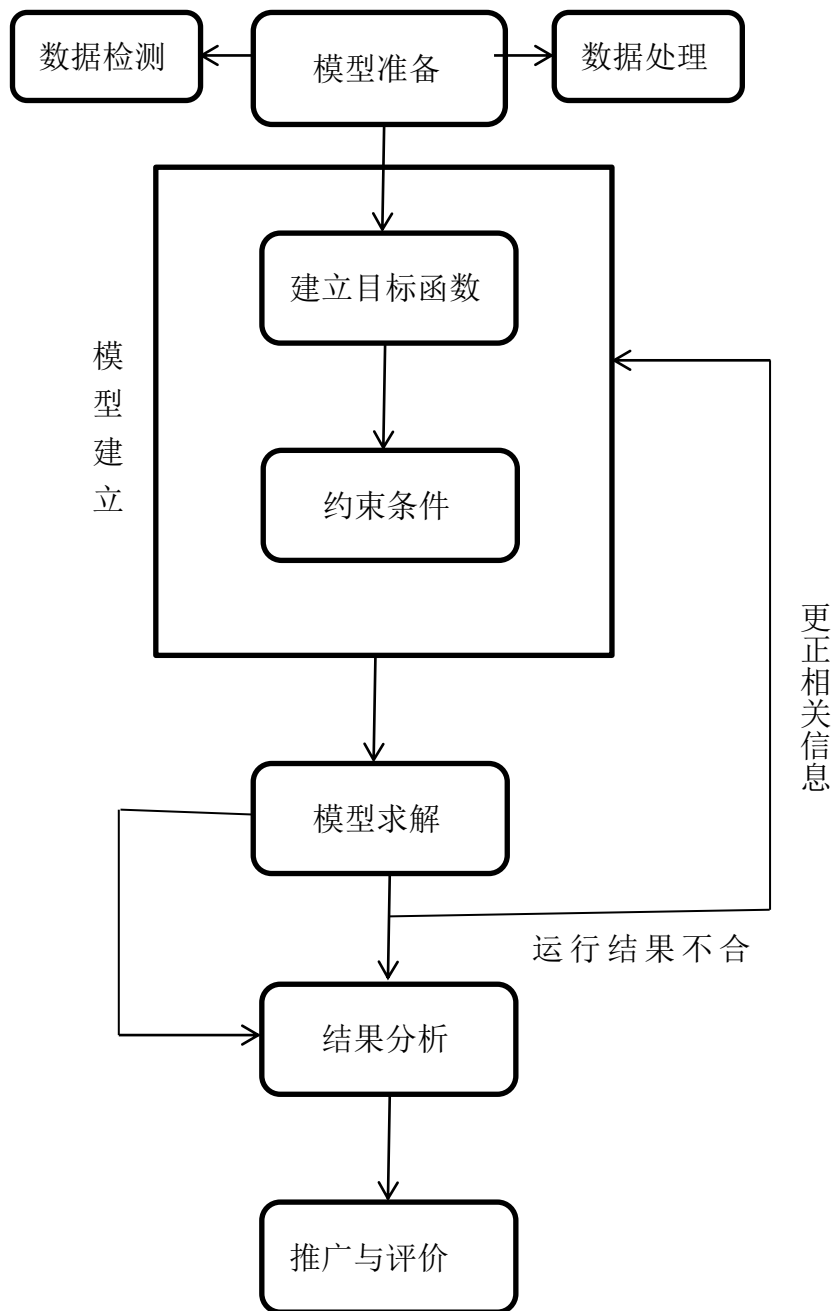
2.2 问题二分析

在问题一的基础上，需满足巡检工人 2 小时左右休息一次，休息时间为 5-10 分钟，并且在中午 12 点及下午 6 点左右进餐，进餐时间为半个小时，为了方便建立优化模型，规定休息时间为 10 分钟，且不考虑进餐时间，类似问题一，以一个班为基准，同样为尽可能减少人力资源，以时间最少建立目标函数，增加了约束条件，利用 lingo 程序进行优化，得出满足限制条件的最优安排工人人数、巡检时间表和巡检路线图。

2.3 问题三分析

在问题二的基础之上，采用错时上班，将问题 1 与问题 2 中的情况再次重新分析，综合考虑人力资源消耗尽可能的少和每名工人在 8 小时左右的工作量均衡等方面因素，为使工作最大化，人力资源最小化，建立合理多目标函数，利用 lingo 软件分别给出错时上班最优化的巡检人数及巡检时间安排，并对问题一、问题二分别进行比对讨论，得出固定上班与错时上班哪一种上班方式更节省人力资源。

2.4 解题思路






3. 问题假设

- (1) 附件中给出的数据来源有效。
- (2) 在巡检过程中，每个巡检人员的技术熟练程度相同，巡检耗时一定，不会出现特殊情况而耽误时间。
- (3) 为减少人力资源且保障工作量均衡的情况下，假设每个班每个人均工作 8 小时左右。
- (4) 每名工人从第 i 处到第 j 处巡检点不考虑速度，则所需路程时间相同。
- (5) 每个巡检点在同一时刻仅需一名工人解决巡检，且同一时刻一名工人只能巡检一个点。
- (6) 每个班每次巡检完不考虑返回时间，只要工作时间达到就可以离开工作岗位。

4. 符号说明

t_{ij}	第 i 处到第 j 处巡检点路途中所消耗的时数（单位：分钟）其中 $i = j$ 时， $t_{ij} = 0$
t_i	第 i 处巡检所消耗时数(分钟)
T_i	第 i 处巡检周期（分钟）
$a_{ki} : 0-1$ 变量	$a_{ki} = \begin{cases} 0 & \text{第 } k \text{ 个人没有到第 } i \text{ 处巡检点} \\ 1 & \text{第 } k \text{ 个人到第 } i \text{ 处巡检点} \end{cases}$
$a_{kj} : 0-1$ 变量	$a_{kj} = \begin{cases} 0 & \text{第 } k \text{ 个人没有到第 } j \text{ 处巡检点} \\ 1 & \text{第 } k \text{ 个人到第 } j \text{ 处巡检点} \end{cases}$
$a_{k+s} : 0-1$ 变量	$a_{k+s} = \begin{cases} 0 & \text{第 } k+n \text{ 个人没有到第 } s \text{ 处巡检点} \\ 1 & \text{第 } k+n \text{ 个人有到第 } s \text{ 处巡检点} \end{cases}$
$a_{22i} : 0-1$ 变量	$a_{22i} = \begin{cases} 1 & \text{从第 } 22 \text{ 处巡检点到第 } i \text{ 处巡检点} \\ 0 & \text{从第 } 22 \text{ 处巡检点没有到第 } i \text{ 处巡检点} \end{cases}$
$a_{kp} : 0-1$ 变量	$a_{kp} = \begin{cases} 1 & \text{从第 } k \text{ 处巡检点到第 } p \text{ 处巡检点} \\ 0 & \text{从第 } k \text{ 处巡检点没有到第 } p \text{ 处巡检点} \end{cases}$

$a_{ki}^{(x)}$: 0-1变量	$a_{ki}^{(x)} = \begin{cases} 1 & \text{X班第k个工人在第i处巡检} \\ 0 & \text{X班第k个工人不在第i处巡检} \end{cases}$
 表示巡检点;	 表示经过巡检点而不巡检
 表示行走路线方向	

5. 模型准备

5.1 数据处理

(1) 为了方便建立下列模型，对附件的巡检点位号进行编号由下表所示：

表 5-1 巡检点位号编号

位号	XJ-0001	XJ-0002	XJ-0003	XJ-0004	XJ-0005	XJ-0006	XJ-0007
编号	1	2	3	4	5	6	7
位号	XJ-0008	XJ-0009	XJ-0010	XJ-0011	XJ-0012	XJ-0013	XJ-0014
编号	8	9	10	11	12	13	14
位号	XJ-0015	XJ-0016	XJ-0017	XJ-0018	XJ-0019	XJ-0020	XJ-0021
编号	15	16	17	18	19	20	21
位号	XJ-0022	XJ-0023	XJ-0024	XJ-0025	XJ-0026		
编号	22	23	24	25	26		

(2) 采用穷举法将第 i 处到第 j 处巡检点所需最短时间计算出来，得出第 i 处到第 j 处巡检点所需最短时间汇总图^[3]（见附件 1）

5.2 有效数据检测

因诸多原因，避免不了数据发生错误，下面分别对给出的周期、巡检耗时的数据中相对特别大的数据视为错误数据，对其进行检测（路程远近不同，因此巡检点之间的路程耗时数不在检测范围之内）^[2]：

在 26 个巡检点的周期中其中有 4 个数据相对平均值特别大，则

周期数据正确率为： $\frac{22}{26} \times \% \approx 85\%$

在 26 个巡检点的巡检耗时数据中有 2 个数据相对平均值特别大，则

巡检耗时数据正确率为： $\frac{24}{26} \times \% = 92\%$

根据数据正确率达 85%以上，在后面数据运用中认为是有效的。

6. 模型的建立与求解

6.1 问题一模型建立与求解

6.1.1 模型建立

在保障某化工厂 26 个巡检点能正常运行的情况下，需安排工人巡检，为了减少消耗的人力资源，使用工人人数尽可能少的情况下完成巡检任务。该问题因为固定时间上班，不考虑休息，采用三班倒的方式上班，故假设一个班在 8 小时左右工作时间内巡检完了之后，下一个班再巡检，则以一个班建立模型即可。下面是一个班从 i 到 j 巡检点之间所有时间之和最小（即所用工人最少）为优化目标^[4]，建立如下模型：

目标函数 1：

$$\min Z = \sum_{k=1}^n \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki} \cdot a_{kj} (t_{ij} + t_i + T_i) + (1 - a_{k22}) \cdot N$$

（一个班中所有工人从 i 到 j 巡检点所有时间，即包括从 i 到 j 路上耗时+巡检耗时+周期）

目标函数 2：

$$\min N = \{b_1, b_2, b_3, \dots, b_{26}\}$$

（第 k 个人从巡检点 22 到巡检点 i 所有路程耗时的时间中取最短的时间）

$$\text{其中 } b_1 = 8a_{ki} \cdot a_{kj} \quad b_2 = 6a_{ki} \cdot a_{kj} \quad b_3 = 7a_{ki} \cdot a_{kj} \quad b_4 = 3a_{ki} \cdot a_{kj}$$

$$b_5 = 8a_{ki} \cdot a_{kj} \quad b_6 = 8a_{ki} \cdot a_{kj} \quad b_7 = 10a_{ki} \cdot a_{kj} \quad b_8 = 9a_{ki} \cdot a_{kj}$$

$$b_9 = 4a_{ki} \cdot a_{kj} \quad b_{10} = 13a_{ki} \cdot a_{kj} \quad b_{11} = 15a_{ki} \cdot a_{kj} \quad b_{12} = 18a_{ki} \cdot a_{kj}$$

.....

（ b_i 表示巡检点 22 到各个巡检点的最短时间）

因为工作时间 8 小时左右即工作量相差在 10 分钟内都属于正常，由此，对 N 进行

修正，取 N 的平均值为 9，由上，得到最终目标函数：

$$\min Z = \sum_{k=1}^n \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki} \cdot a_{kj} (t_{ij} + t_i + T_i) + (1 - a_{k22}) \cdot 9$$

s.t.

$$470 \leq \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki} \cdot a_{kj} (t_{ij} + t_i) + (1 - a_{k22}) \cdot 9 \leq 480 \quad (k = 1, 2, \dots, n)$$

(第 k 个人从巡检点 i 到巡检点 j 所花工作时间之和在 7 时 50 分钟到 8 小时之间
即第 k 个人工作量为 470 分钟至 480 分钟，以保障每个工人工作量在 8 小时左右)

$$\sum_{k=1}^n a_{ki} = \left\lceil \frac{480}{T_i} \right\rceil + 1$$

(一班中所有工人在 i 处的巡检次数之和=8 小时内第 i 处需要巡检的次数, 避免在
8 小时内第 i 处巡检次数不够而导致不能正常运行)

$$a_{ki} \cdot (t_{ij} + t_i) = T_j \quad (i, j = 1, 2, \dots, 26) \quad (k = 1, 2, \dots, n)$$

(第 k 个人在第 i 处巡检点的巡检耗时+从第 i 处到第 j 处巡检点之间路程所用时间=第 j 处的周期，避免时间浪费或不够)^[10]

$$\left| \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki} a_{kj} (t_{ij} + t_i) - \sum_{i=1}^{26} \sum_{j=1}^{26} a_{k+si} a_{k+sj} \cdot (t_{ij} + t_i) \right| \leq 10 \text{ 分钟}$$

$$(k, s = 1, 2, \dots, n \text{ 且 } k \neq s)$$

(第 k 个人工作量与第 k+s 个人工作量相差范围小于等于 10 分钟，以确保每名工人在上班 8 小时内工作量的平衡)

6.1.2 模型求解

利用 LINGO 软件编程^[5] (见附件 2) 对上述模型进行计算，其根据运行结果得出要完成该任务需要 5 个工人，具体巡检情况见图 6-1:

巡检站	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
第一人	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
第二人	0	0	1	0	1	1	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	1	1	0
第三人	0	0	1	0	0	1	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0
第四人	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0
第五人	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1

图 6-1 工人到各巡检点巡检情况图

(注释: 1: 表示工人到巡检点巡检; 0: 表示工人没有到该巡检点巡检)

通过图 6-1 可清楚知道每个工人到各个巡检点的巡检情况，并制定巡检时间表及路线图。假设两次检测的时间间隔的误差在正负 2 分钟以内包括 2 分钟，划分工作区域(按

照路径最短即时间用得最少进行划分：利用 lingo 软件运行可得，程序见附件 6) [4]。在固定时间上班的情况下，进行三班倒，每个巡检工人在固定的区域里巡检，以下分别是不同班五个巡检员在相同巡检点的不同巡检时间安排，且为工作时间 8 小时左右中的一个循环划分的巡检时间情况和巡检路线图：

第一个巡检人员巡检时间表及巡检路线图：

表 6-1 第一个巡检人员巡检时间表

巡检点	一班	二班	三班
22	8:00-8:02	16:00-16:02	0:00-0:02
20	8:04-8:07	16:04-16:07	0:04-0:07
19	8:09-8:11	16:09-16:11	0:09-0:11
2	8:14-8:16	16:14-16:16	0:14-0:16
1	8:18-8:21	16:18-16:21	0:18-0:21
4	8:26-8:28	16:26-16:28	0:26-0:28
21	8:29-8:32	16:29-16:32	0:29-0:32

根据表 6-1 第一个巡检人员巡检时间表得第一个巡检人员巡检路线图^[9](见图 6-2)：

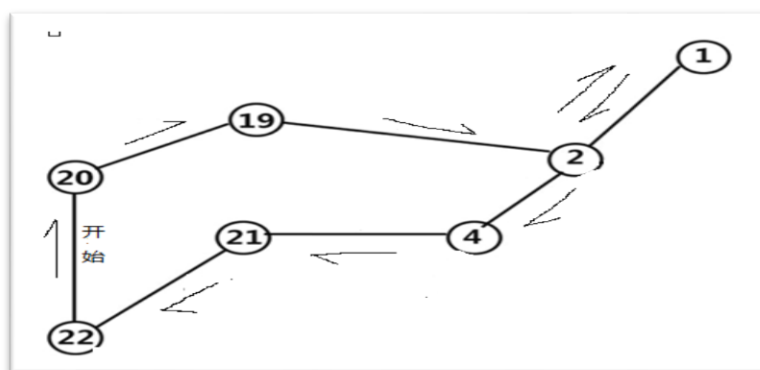


图 6-2 第一个巡检人员巡检路线图

由图 6-2 可知第一个巡检工人从巡检点 22 开始巡检，按箭头指向分别巡检了巡检点 22、巡检点 20、巡检点 19、巡检点 2、巡检点 1，再返回巡检点 2 进行巡检，由箭头指向继续向巡检点 4、巡检点 21 进行巡检，最后回到巡检点 22 下班。

第二个巡检人员巡检时间表及巡检路线图：

表 6-2 第二个巡检人员巡检时间表

巡检点	一班	二班	三班
23	8:01-8:04	16:01-16:04	0:01-0:04
24	8:05-8:07	16:05-16:07	0:05-0:07
9	8:09-8:13	16:09-16:13	0:09-0:13
25	8:16-8:18	16:16-16:18	0:16-0:18
17	8:19-8:21	16:19-16:21	0:19-0:21
8	8:22-8:25	16:22-16:25	0:22-0:25

根据表 6-2 第二个巡检人员巡检时间表得第二个巡检人员巡检路线图(见图 6-3):

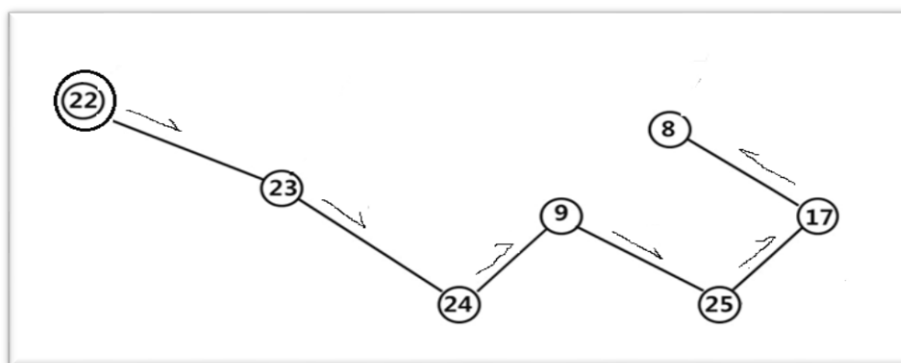


图 6-3 第二个巡检人员巡检路线图

由图 6-3 可知第二个巡检工人从巡检点 22 出发, 经过巡检点 22 且不巡检, 按箭头指向分别巡检了巡检点 23、巡检点 24、巡检点 9、巡检点 25、巡检点 17, 巡检点 8, 最后下班。

第三个巡检人员巡检时间表及巡检路线图:

表 6-3 第三个巡检人员巡检时间表

巡检点	一班	二班	三班
3	8:07-8:10	16:07-16:10	0:07-0:10
5	8:11-8:13	16:11-16:13	0:11-0:13
7	8:15-8:17	16:15-16:17	0:15-0:17
6	8:27-8:30	16:27-16:30	0:27-0:30
14	8:31-8:34	16:31-16:34	0:31-0:34

根据表 6-3 第三个巡检人员巡检时间表得第三个巡检人员巡检路线图(见图 6-4):

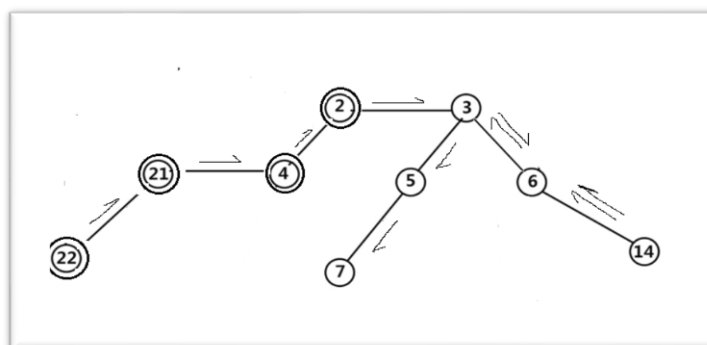


图 6-4 第三个巡检人员巡检路线图

由图 6-4 可知第三个巡检工人按箭头指向从巡检点 22 出发, 经过巡检点 22、巡检点 21、巡检点 4、巡检点 2 且不巡检, 到达巡检点 3, 再以箭头指向分别对巡检点 3、巡检点 6、巡检点 14 进行巡检, 再原路返回到巡检点 3, 又对巡检点 5、巡检点 7 进行巡检, 最后下班。

第四个巡检人员巡检时间表及巡检路线图:

表 6-4 第四个巡检人员巡检时间表

巡检点	一班	二班	三班
18	8:18-8:20	16:18-16:20	0:18-0:20
16	8:23-8:26	16:23-16:26	0:23-0:26
13	8:28-8:33	16:28-16:33	0:28-0:33
11	8:35-8:38	16:35-16:38	0:35-0:38

根据表 6-4 第四个巡检人员巡检时间表得第四个巡检人员巡检路线图(见图 6-5):

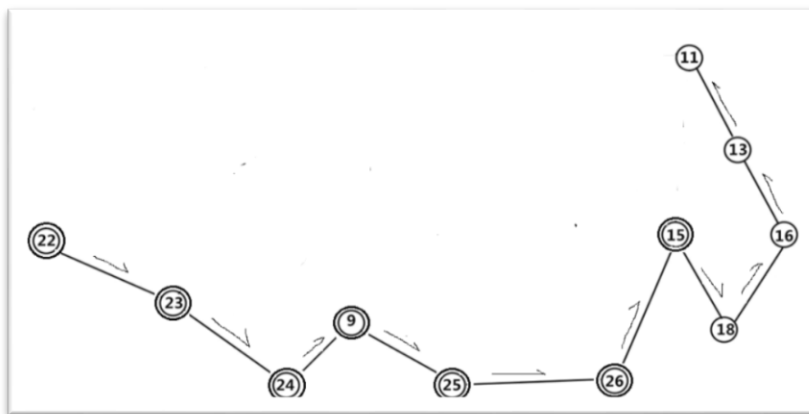


图 6-5 第四个巡检人员巡检路线图

由图 6-5 可知第四个巡检工人按箭头指向从巡检点 22 出发, 分别经过巡检点 22、巡检点 23、巡检点 24、巡检点 9、巡检点 25、巡检点 26、巡检点 15 且不巡检, 到了巡检点 18 分别对巡检点 18、巡检点 16、巡检点 13 和巡检点 11 进行巡检, 最后下班。

第五个巡检人员巡检时间表及巡检路线图:

表 6-5 第五个巡检人员巡检时间表

巡检点	一班	二班	三班
10	8:13-8:15	16:13-16:15	0:13-0:15
12	8:21-8:23	16:21-16:23	0:21-0:23
15	8:25-8:27	16:25-16:27	0:25-0:27
26	8:33-8:35	16:33-16:35	0:33-0:35

根据表 6-5 第五个巡检人员巡检时间表得第五个巡检人员巡检路线图(见图 6-6):

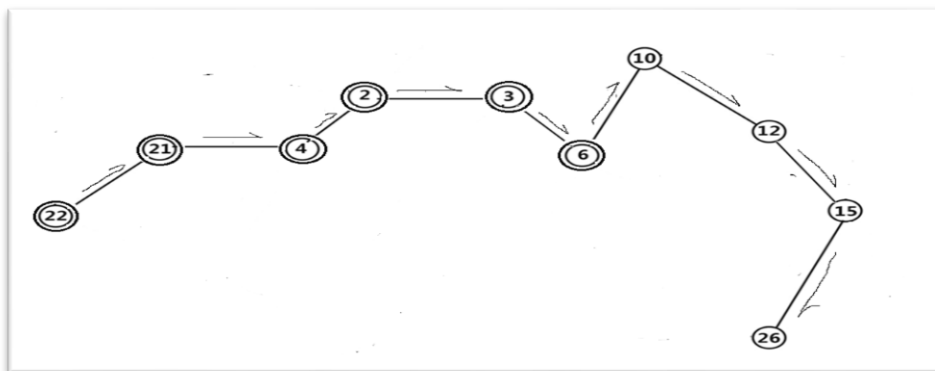


图 6-6 第五个巡检人员巡检路线图

由图 6-6 可知第五个巡检工人按箭头指向从巡检点 22 出发，分别经过巡检点 22、巡检点 21、巡检点 4、巡检点 2、巡检点 3、巡检点 6 且不巡检，到了巡检点 10 开始对巡检点 10、巡检点 12、巡检点 15 及巡检点 26 进行巡检，最后下班。

6.1.3 结果分析

借助 excel 分别对五个人 8 小时左右的工作量进行统计（见表 6-6）。

表 6-6 五个人一天工作量汇总表^[7]

人员	k1	k2	k3	k4	k5
工作量（分钟）	480	475	476	472	478

由表 6-6 绘制出其工作量饱和情况的饼形图，如图 6-7 所示：

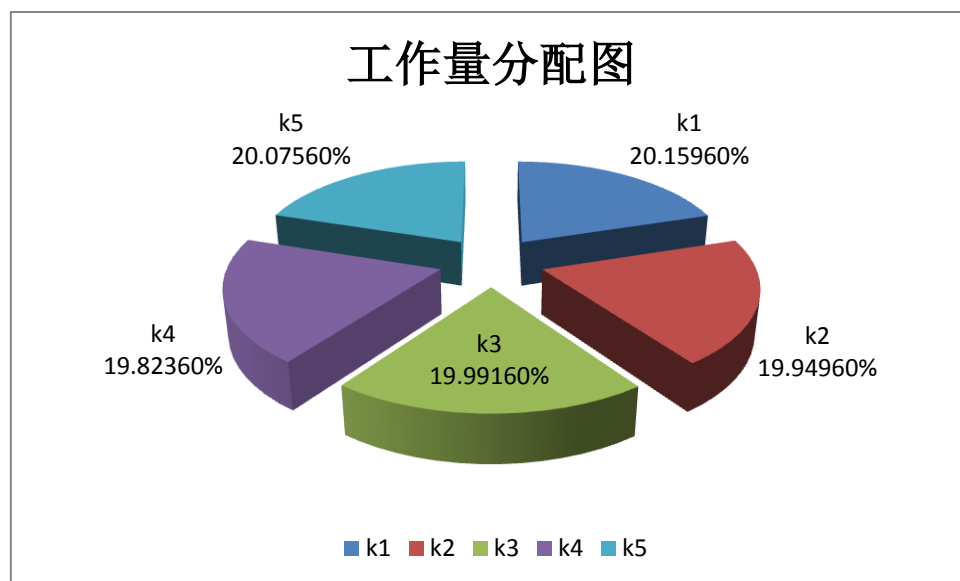


图 6-7 五个工人工作量饱和状态图^[6]

据饼形图非常清楚的显示出五个工人的工作量均达到饱和状态。根据表 6-6 和图 6-7 反应出：工人在 8 小时左右的工作量，没有造成时间浪费或不够。工作量分配在 19%-20%之间，满足了每名工人在 8 小时左右工作量的均衡，因此针对该问题每班安排 5 个工人巡检达到了优化人力资源的目的。

由于在求巡检点 22 到各巡检点最短时间进行了取平均值的处理，因此巡检人数可能存在合理的误差。

6.2 问题二模型建立与求解

6.2.1 模型建立

在问题一基础之上，增加约束条件即每个工人在工作量达到 2 小时左右必须休息，并假设每个工人休息时间都为 10 分钟，且在中午 12 点和下午 6 点左右同时进餐，不考虑进餐时间，固定时间三班倒，以时间最少为优化目标，建立如下模型^[8]：

$$\min Z = \sum_{k=1}^n \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki} \cdot a_{kj} (t_{ij} + t_i + T_i) + (1 - a_{k22}) \cdot 9 - \sum_{k=1}^n \sum_{i=1}^{26} \sum_{p=1}^{26} a_{ki} \cdot a_{kp} \cdot 10$$

(一个班所有工人以巡检点 22 为起点到 j 巡检点所有时间-所有工人在各个休息点的休息时间, 即包括从 i 到 j 路上耗时+巡检耗时+周期-休息时间)

s.t.

$$440 \leq \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki} \cdot a_{kj} (t_{ij} + t_i) + (1 - a_{k22}) \cdot 9 \leq 450 \quad (k = 1, 2, \dots, n)$$

(两个小时休息一次, 在 8 小时内需休息 3 次, 则第 k 个人工作量为 440 分钟至 450 分钟)

$$110 \leq \sum_{p=1}^{26} \sum_{i=1}^{26} a_{ki} \cdot a_{kp} (t_{ip} + t_i) + (1 - a_{k22}) \cdot 9 \leq 120 \quad (p \neq j)$$

(110=<第 k 个人从巡检点 p 到巡检点 i 所有工作量之和<=120)

$$\sum_{k=1}^n a_{ki} = \left\lceil \frac{480}{T_i} \right\rceil + 1$$

(一班中所有工人在 i 处的巡检次数之和=8 小时内第 i 处需要巡检的次数)

$$a_{ki} \cdot (t_{ij} + t_i) = T_j \quad (i, j = 1, 2, \dots, 26) \quad (k = 1, 2, \dots, n)$$

(第 k 个人在第 i 处巡检点的巡检耗时+从第 i 处到第 j 处巡检点之间路程所用时间=第 j 处的周期)

$$a_{k+sm} \cdot (t_{mj} + t_m) = T_j \quad (m \neq i)$$

(第 $k+s$ 个人在巡检点 m 处的巡检时间 m 到 j 路程上的耗时刚好等于第 j 个巡检点周期, 则第 $k+s$ 个人正好巡检第 j 处巡检点)

$$\left| \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki} a_{kj} (t_{ij} + t_i) - \sum_{i=1}^{26} \sum_{j=1}^{26} a_{k+si} a_{k+sj} \cdot (t_{ij} + t_i) \right| \leq 10$$

$$(k, s = 1, 2, \dots, n)$$

(第 k 个人工作量与第 $k+s$ 个人工作量相差范围<=10 分钟, 确保每个工人工作量均衡)

6. 2. 2 模型求解

利用 LINGO 软件编程，对上述模型进行计算，部分程序（部分程序见附件 3），根据运行结果结合人工整理筛选出有用数据，得出要完成该任务需要 6 个工人，具体巡检情况（见图 6-8）。

巡检站	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
第一人	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	0	0
第二人	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
第三人	0	0	1	0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
第四人	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
第五人	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
第六人	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	1

图 6-8 工人到各巡检点巡检情况表

（注释：1：表示工人到巡检点巡检；0：表示工人没有到该巡检点巡检）

通过图 6-8 可清楚知道每个工人到各个巡检点是否巡检情况，制定巡检时间表及路线图，分别如下：

第一个巡检人员巡检时间表及巡检路线图：

表 6-7 第一个巡检人员巡检时间表

巡检点	一班	二班	三班
24	8:02-8:04	16:02-16:04	0:02-0:04
23	8:05-8:08	16:05-16:08	0:05-0:08
22	8:09-8:11	16:09-16:11	0:09-0:11
20	8:13-8:16	16:13-16:16	0:13-0:16
19	8:18-8:20	16:18-16:20	0:18-0:20

根据表 6-7 第一个巡检人员巡检时间表得第一个巡检人员巡检路线图(见图 6-9)：

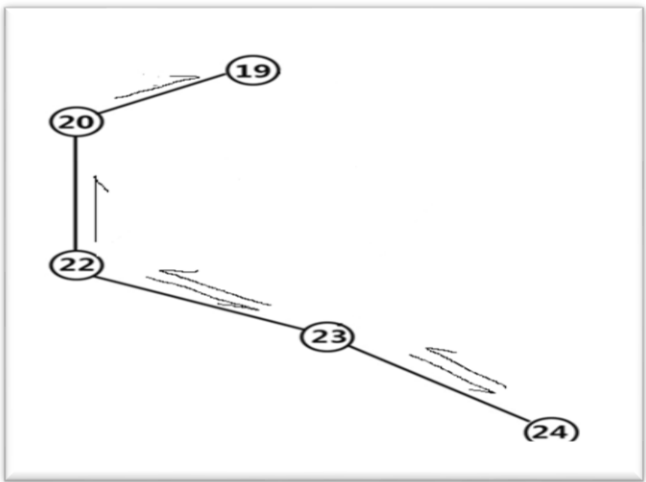


图 6-9 第一个巡检人员巡检路线图

由图 6-8 可知第一个巡检工人按箭头指向从巡检点 22 出发，分别对巡检点 22、巡检点 23、巡检点 24 进行巡检，再返回巡检点 23 和巡检点 22，又按箭头指向对巡检点

20 及巡检点 19 进行巡检，最后下班。

第二个巡检人员巡检时间表及巡检路线图：

表 6-8 第二个巡检人员巡检时间表

巡检点	一班	二班	三班
21	8:02-8:05	16:02-16:05	0:02-0:05
4	8:06-8:08	16:06-16:08	0:06-0:08
2	8:11-8:13	16:11-16:13	0:11-0:13
1	8:15-8:18	16:15-16:18	0:15-0:18

根据表 6-8 第二个巡检人员巡检时间表得第二个巡检人员巡检路线图(见图 6-10)：

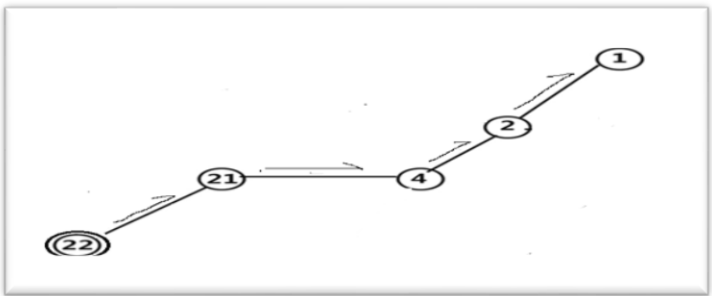


图 6-10 第二个巡检人员巡检路线图

由图 6-10 可知第二个巡检工人按箭头指向从巡检点 22 出发，经过巡检点 22 且不巡检，分别对巡检点 21、巡检点 4、巡检点 2 和巡检点 1 进行巡检，最后下班。

第三个巡检人员巡检时间表及巡检路线图：

表 6-9 第三个巡检人员巡检时间表

巡检点	一班	二班	三班
14	8:09-8:12	16:09-16:12	0:09-0:12
6	8:13-8:16	16:13-16:16	0:13-0:16
3	8:17-8:20	16:17-16:20	0:17-0:20
5	8:21-8:23	16:21-16:23	0:21-0:23
7	8:25-8:27	16:25-16:27	0:25-0:27

根据表 6-9 第三个巡检人员巡检时间表得第三个巡检人员巡检路线图(见图 6-11)：

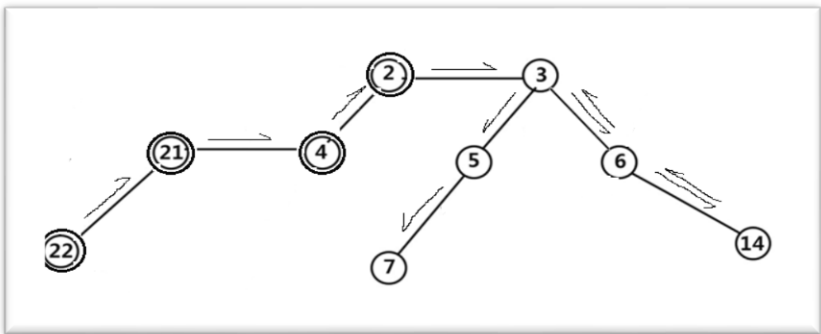


图 6-11 第三个巡检人员巡检路线图

由图 6-11 可知第三个巡检工人按箭头指向从巡检点 22 出发且不巡检，分别经过巡检点 21、巡检点 4、巡检点 2 都不巡检，到达巡检点 3，分别对巡检点 3、巡检点 6、巡检点 14 进行巡检，然后原路返回到巡检点 3，再对巡检点 5 和巡检点 7 进行巡检，最后下班。

第四个巡检人员巡检时间表及巡检路线图：

表 6-10 第四个巡检人员巡检时间表

巡检点	一班	二班	三班
9	8:04-8:08	16:04-16:08	0:04-0:08
25	8:11-8:13	16:11-16:13	0:11-0:13
17	8:14-8:16	16:14-16:16	0:14-0:16
8	8:17-8:20	16:17-16:20	0:17-0:20

根据表 6-10 第四个巡检人员巡检时间表得第四个巡检人员巡检路线图(见图 6-12)：

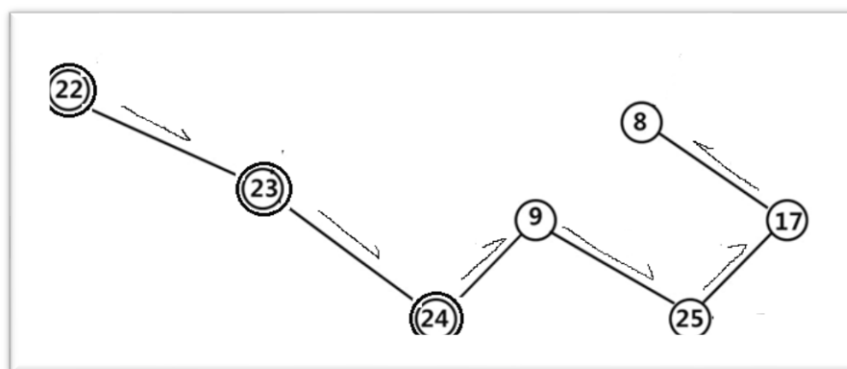


图 6-12 第四个巡检人员巡检路线图

由图 6-12 可知第四个巡检工人按箭头指向从巡检点 22 出发且不巡检，经过巡检点 23、巡检点 24 都不巡检，到达巡检点 9，按箭头指向分别对巡检点 9、巡检点 25、巡检点 17 和巡检点 8 进行巡检，最后下班。

第五个巡检人员巡检时间表及巡检路线图：

表 6-11 第五个巡检人员巡检时间表

巡检点	一班	二班	三班
10	8:13-8:15	16:13-16:15	0:13-0:15
11	8:17-8:20	16:17-16:20	0:17-0:20
13	8:22-8:27	16:22-16:27	0:22-0:27
16	8:29-8:32	16:29-16:32	0:29-0:32

根据表 6-11 第五个巡检人员巡检时间表得第五个巡检人员巡检路线图(见图 6-13)：

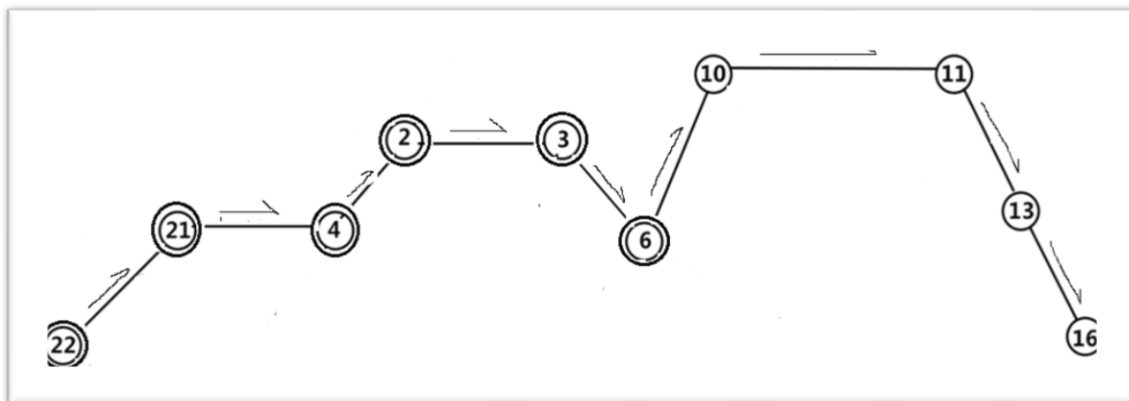


图 6-13 第五个巡检人员巡检路线图

由图 6-13 可知第五个巡检工人按箭头指向从巡检点 22 出发且不巡检，经过巡检点 21、巡检点 4、巡检点 2、巡检点 3、巡检点 6 都不巡检，到达巡检点 10，按箭头指向分别对巡检点 10、巡检点 11、巡检点 13 和巡检点 16 进行巡检，最后下班。

第六个巡检人员巡检时间表及巡检路线图：

表 6-12 第六个巡检人员巡检时间表

巡检点	一班	二班	三班
26	8:10-8:12	16:10-16:12	0:10-0:12
15	8:18-8:20	16:18-16:20	0:18-0:20
18	8:22-8:24	16:22-16:24	0:22-0:24
12	8:28-8:30	16:28-16:30	0:28-0:30

根据表 6-12 第六个巡检人员巡检时间表得第六个巡检人员巡检路线图(见图 6-14)：

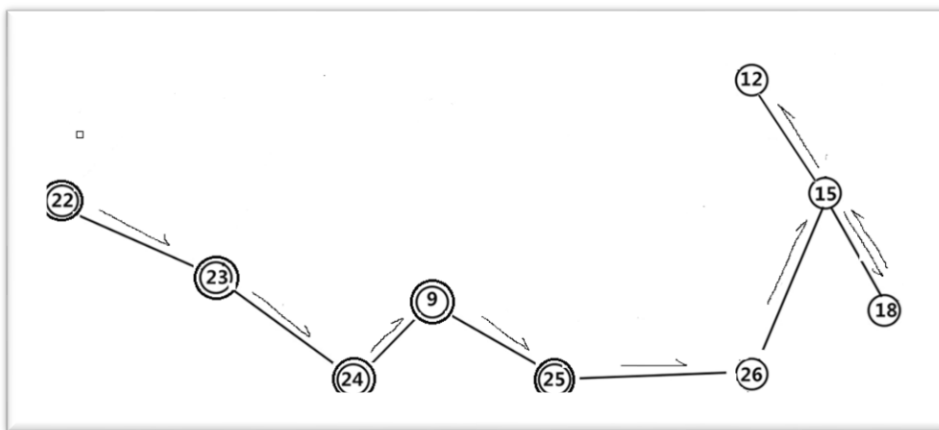


图 6-14 第五个巡检人员巡检路线图

由图 6-14 可知第六个巡检工人按箭头指向从巡检点 22 出发且不巡检，经过巡检点 23、巡检点 24、巡检点 9、巡检点 25 都不巡检，到达巡检点 26，按箭头指向分别对巡检点 26、巡检点 15、巡检点 18 进行巡检，再原路返回到巡检点 15，对巡检点 12 进行巡检，最后下班。

6.2.3 结果分析

借助 excel 分别对六个人 8 小时左右的工作量进行统计（见表 6-13）。

表 6-13 工人休息时间节点表

巡检次数	第一次	第二次	第三次	第四次	总工作时间(分钟)
第一人巡检时间点	9:51	11:52	14:21	16:22	444
第二人巡检时间点	9:52	11:54	14:22	16:24	448
第三人巡检时间点	9:57	12:00	14:27	16:24	455
第四人巡检时间点	9:50	11:50	14:20	16:20	440
第五人巡检时间点	10:03	12:00	14:33	16:26	456
第六人巡检时间点	10:03	12:00	14:33	16:26	456

由表 6-13,利用 excel 将得到的六个工人巡检时间点由柱形图表现出来如图 6-15:

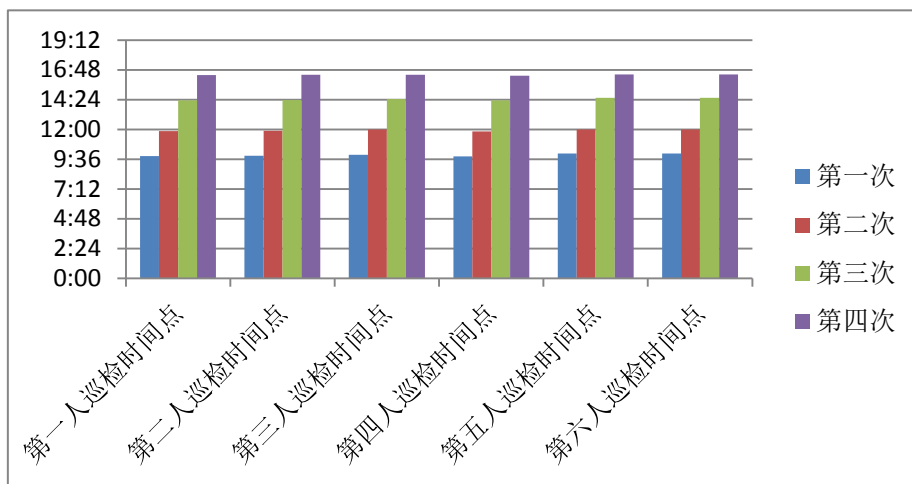


图 6-15 六个工人休息时间节点图

根据图 6-15,可清晰的观察出每个工人在工作量达到 2 小时左右均达到了 10 分钟休息时间。

由表 6-13 绘制出其工作量饱和情况的饼形图, 如图 6-16 所示:

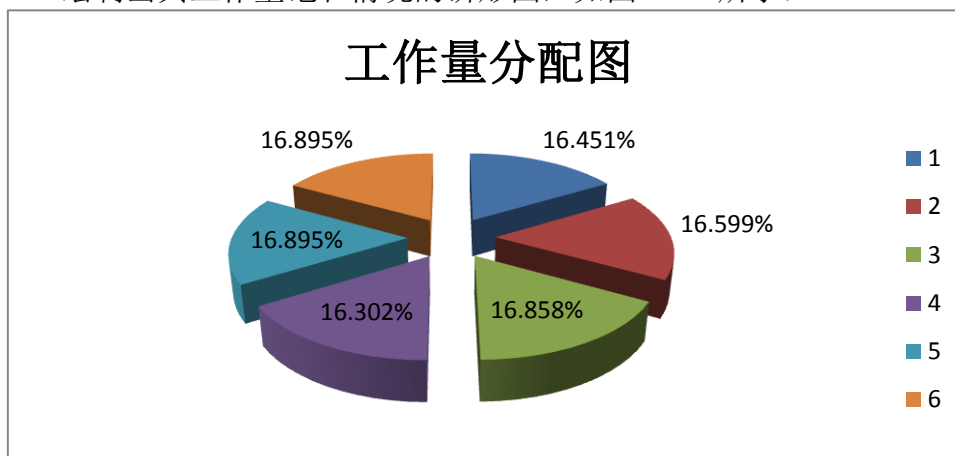


图 6-16 六个人工作量饱和状态图

据饼形图非常清楚的显示出六个工人在每 2 小时左右, 必须休息 10 分钟情况下的工作量达到饱和状态。根据表 6-14 和图 6-14 反应出: 工人在 8 小时左右的工作量, 没

有造成时间浪费或不够。每名工人在 8 小时左右工作量分配在 16.3%-16.9%之间，达到了工作量均衡，因此针对该问题每班安排 6 个工人巡检达到了优化人力资源的目的。

由于在求巡检点 22 到各巡检点最短时间进行了取平均值的处理，及休息时间固定在 10 分钟，因此巡检人数可能存在合理的误差。

6.3 问题三模型建立与求解

6.3.1 错时上班的问题一

6.3.1.1 模型建立

在问题一的基础上，采用错时上班，对其重新进行优化，建立如下模型：

令

$$Z_x = \sum_{k=1}^n \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki}^{(x)} \cdot a_{kj}^{(x)} (t_{ij} + t_i + T_i) + (1 - a_{k22}) \cdot 9 \quad (x=1,2,3)$$

（第 x 班 k 个工人以巡检点 22 为起点到各个巡检点所有时间之和）得目标函数 1：

$$\min z = \sum_{x=1}^3 Z_x$$

（三班所有工人以巡检点 22 为起点到各个巡检点所有时间之和最短，达到减少人力）

令

$$g_x = \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki}^{(x)} \cdot a_{kj}^{(x)} (t_{ij} + t_i) + (1 - a_{k22}^{(x)}) \cdot 9 \quad (x=1,2,3)$$

（第 x 班第 k 个工人所有工作量）

$$G_x = \sum_{k=1}^n g_x$$

（第 x 班 k 个工人所有工作量之和）得目标函数 2 为：

$$\min G = |G_1 - G_2| + |G_2 - G_3| + |G_3 - G_1|$$

（任意两个班中 k 个工人工作量之差最小，使得在错时上班的情况下，每日变动不能很大）

方便模型求解，进行加权平均处理，得最终目标函数为：

$$\min Y = 0.7Z + 0.3T$$

s.t.

$$1410 \leq \sum_{x=1}^3 t_x \leq 1440$$

(三个班工作量之和范围控制在 1410 分钟至 1440 分钟)

$$\sum_{k=1}^n a_{ki}^{(x)} = \left\lceil \frac{480}{T_i} \right\rceil + 1$$

(x 班中所有工人在 i 处的巡检次数之和=8 小时内第 i 处需要巡检的次数)

$$|G_s - G_m| \leq 35 \quad (s, m = 1, 2, 3)$$

(任意两个班 k 个工人工作量之和相差不超过 35 分钟)

$$a_{ki}^{(x)} \cdot (t_{ij} + t_i) = T_j \quad (i, j = 1, 2, \dots, 26, x = 1, 2, 3)$$

(x 班第 k 个人在第 i 处巡检点的巡检耗时+从第 i 处到第 j 处巡检点之间路程所用时间=第 j 处的周期)

$$\left| \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki}^{(x)} a_{kj}^{(x)} (t_{ij} + t_i) - \sum_{i=1}^{26} \sum_{j=1}^{26} a_{k+si}^{(x)} a_{k+sj}^{(x)} (t_{ij} + t_i) \right| \leq 10$$

(k, s = 1, 2, \dots, n)

(x 班第 k 个人工作量与第 $k+s$ 个人工作量相差范围 ≤ 10 分钟)

6.3.1.2 模型求解与结果对照分析

利用 LINGO 软件 (程序见附件 4) 对上述模型进行运行, 并将结果结合人工对数据进行整理得出要完成该任务需要 4 个工人, 具体巡检情况如图 6-17 所示:

巡检站	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
第一人	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
第二人	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
第三人	0		1	0	1	1	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
第四人	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	1	0	1	0	0	0	0	0	0	0	0

图 6-17 工人到各巡检点的巡检情况

下面是通过数据整理得到错时上班中一班与二班的错时对照表 (见表 6-14) (三班没有呈现出来, 因为在其循环之中):

表 6-14 错时上班中一班与二班时间节点对照表

人员巡检时间	一班	二班
第 1 人巡检时间	8:00-16:01	15:54-23:55
第 2 人巡检时间	8:00-15:57	16:00-23:57
第 3 人巡检时间	8:00-16:00	15:49-23:51
第 4 人巡检时间	8:00-16:01	15:52-23:55

根据表 6-14, 在满足约束条件的情况下, 不仅能按时完成任务还可达到提前几分钟下班的可能 (因为约束条件中两个班工作量绝对值之差小于 10 分钟, 所以只能提前几分钟下班)。

以问题一中二班上班时间为例，与错时上班的问题一进行比对（见表 6-15）：

表 6-15 固定上班与错时上班对照表

人员巡检时间	固定上班	错时上班
第 1 人巡检时间	16:00-00:00	15:54-23:55
第 2 人巡检时间	16:00-23:55	16:00-23:57
第 3 人巡检时间	16:00-23:56	15:49-23:51
第 4 人巡检时间	16:00-23:52	15:52-23:55
第 5 人巡检时间	16:00-23:58	

据表 6-15 的对照，可见在问题一中固定上班需要 5 人减少到了 4 人，充分说明了同等限制条件下错时上班更优化，满足人力资源的优化。

6.3.2 错时上班的问题二

6.3.2.1 模型建立

在问题二的基础上，采用错时上班，对其重新进行优化，建立如下模型：

令

$$Z_x = \sum_{k=1}^n \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki}^{(x)} \cdot a_{kj}^{(x)} (t_{ij} + t_i + T_i) + (1 - a_{k22}^{(x)}) \cdot 9 - \sum_{k=1}^n \sum_{i=1}^{26} \sum_{p=1}^{26} a_{ki}^{(x)} \cdot a_{kp}^{(x)} \cdot 10$$

（第 x 班 k 个工人以巡检点 22 为起点到各个巡检点所有时间之和-k 个工人的休息时间之后）得目标函数 1：

$$\min z = \sum_{x=1}^3 Z_x$$

（三班所有工人以巡检点 22 为起点到各个巡检点所有时间之和最短，达到减少人力）

令

$$g_x = \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki}^{(x)} \cdot a_{kj}^{(x)} (t_{ij} + t_i) + (1 - a_{k22}^{(x)}) \cdot 9 \quad (x=1,2,3)$$

（第 x 班第 k 个工人所有工作量）

$$G_x = \sum_{k=1}^n g_x$$

（第 x 班 k 个工人所有工作量之和）得目标函数 2 为：

$$\min G = |G_1 - G_2| + |G_2 - G_3| + |G_3 - G_1|$$

（任意两个班中 k 个工人工作量之差最小，使得在进行错时上班的情况下，每日变动

不能很大)

方便模型求解，进行加权平均处理，得最终目标函数为：

$$\min Y = 0.7Z + 0.3G$$

s.t.

$$1410 \leq \sum_{x=1}^3 t_x \leq 1440$$

(三个班工作量之和范围控制在 1410 分钟至 1440 分钟)

$$110 \leq \sum_{p=1}^{26} \sum_{i=1}^{26} a_{ki}^{(x)} \cdot a_{kj}^{(x)} (t_{ij} + t_i) + (1 - a_{k22}^{(x)}) \cdot 9 \leq 120 \quad (p \neq j)$$

(110=<x 班第 k 个人从巡检点 p 到巡检点 i 所有工作量之和<=120)

$$\sum_{k=1}^n a_{ki}^{(x)} = \left\lceil \frac{480}{T_i} \right\rceil + 1$$

(x 班中所有工人在 i 处的巡检次数之和=8 小时内第 i 处需要巡检的次数)

$$|T_s - T_m| \leq 35 \quad (s, m = 1, 2, 3)$$

(任意两个班 k 个工人工作量之和相差不超过 35 分钟)

$$a_{ki}^{(x)} \cdot (t_{ij} + t_i) = T_j \quad (i, j = 1, 2, \dots, 26, x = 1, 2, 3)$$

(x 班第 k 个人在第 i 处巡检点的巡检耗时+从第 i 处到第 j 处巡检点之间路程所用时间=第 j 处的周期)

$$a_{k+sm}^{(x)} \cdot (t_{mj} + t_m) = T_j \quad (m \neq i)$$

(x 班第 k+s 个人在巡检点 m 处的巡检时间 m 到 j 路程上的耗时刚好等于第 j 个巡检点周期，则第 k+s 个人正好巡检第 j 处巡检点)

$$\left| \sum_{i=1}^{26} \sum_{j=1}^{26} a_{ki}^{(x)} a_{kj}^{(x)} (t_{ij} + t_i) - \sum_{i=1}^{26} \sum_{j=1}^{26} a_{k+si}^{(x)} a_{k+s+j}^{(x)} \cdot (t_{ij} + t_i) \right| \leq 10$$

$$(k, s = 1, 2, \dots, n)$$

(x 班第 k 个人工作量与第 $k+s$ 个人工作量相差范围<=10 分钟)

6.3.2.2 模型求解与结果对照分析

利用 LINGO 软件 (程序见附件 5) 对上述模型进行计算，将运行结果结合人工对数据进行整理得出要完成该任务需要 5 个工人，具体巡检情况如图 6-18:

巡检站	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
第一人	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
第二人	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1	0
第三人	0	0	1	0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
第四人	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	1
第五人	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	1		0	0	0	0	0	0	0	0	0

图 6-18 工人在各巡检点的巡检情况

以问题二中一班上班时间为例，与错时上班的问题二进行比对（见表 6-16）：

表 6-16 固定上班与错时上班对照表

人员巡检时间	固定上班	错时上班
第 1 人巡检时间	8:00-16:22	8:00-16:27
第 2 人巡检时间	8:00-16:24	7:52-16:21
第 3 人巡检时间	8:00-16:23	7:54-16:32
第 4 人巡检时间	8:00-16:20	7:52-16:26
第 5 人巡检时间	8: 00-16:26	7:59-16:30
第 6 人巡检时间	8: 00-16:26	

据表 6-16 的对照，可见在问题二中固定上班需要 6 人减少到了 5 人，充分说明了同等限制条件下错时上班更优化，满足人力资源的优化。

7. 模型的评价与推广

7.1 模型的优点

（1）模型中采用 0-1 规划模型与本文研究问题相符吻合，具有很好的合理性。

（2）运用 0-1 规划模型针对此问题通过对位号、周期、巡检时间以及路程的耗时数据分析整理，借助 lingo 软件，以减少人力资源为目的，在工人工作量平衡问题得到一定程度解决的基础上，迅速掌握了数据特点，为建立更合理的类似模型提供了参考的经验。

（3）因为建立的优化模型与实际生活紧密联系，结合实际的情况对相应问题进行求解并整理，所以使得模型具有通用性和推广性；

（4）根据此模型结果呈现出：在相同限制条件下，错时上班比固定上班更加优化，由此可将其结果在生活中的类似案例进行推广。

7.2 模型的缺点

（1）在模型建立过程中，对限制条件存在没考虑全面的地方，加之人工对数据进行了一定的处理，因此可能存在与最优结果的差距。

（2）在该问题中影响模型的约束条件诸多，程序运行存在一定难度。

8. 参考文献

- [1]. 数学建模案例分析 白其峥主编 北京 海洋出版社 2000
- [2]. 数学建模 原理与方法 蔡锁章主编 北京 海洋出版社 2000
- [3]. 数学建模竞赛赛题简析与论文点评 西安交大近年参赛论文选编 赫孝良等
[选编] 西安西
- [4]. 建模、变换、优化——结构综合方法新进展, 隋允康著, 大连理工大学出版社,
(1986)
- [5]. 数学建模案例精选 朱道元等编著 北京: 科学出版社, 2003
- [6]. 数学建模: 原理与方法 蔡锁章主编 北京: 海洋出版社, 2000
- [7]. 数学建模的理论与实践 吴翊, 吴孟达, 成礼智编著 长沙: 国防科技大学出版社, 1999
- [8]. 数学建模作者: 沈继红 施久玉 高振滨 张晓威 出版社: 出版日期: 1996
年 5 月第 1 版 页数: 351
- [9]. Cezik, T. Staffing multiskill call centers via linear programming and
simulation. Management science, 2006, 01
- [10]. Fukunaga, A. Staff scheduling for inbound call centers and customer
contact centers. 2002

9. 附录

附件 1:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
1	0	2	3	5	4	4	6	6	11	9	11	15	13	5	17	15	7	18	7	9	6	8	9	10	8	11
2	2	0	1	3	2	2	4	4	9	7	9	13	11	3	15	13	5	15	5	7	4	6	7	8	6	9
3	3	1	0	4	1	1	3	3	8	6	8	12	10	2	14	12	4	16	6	8	5	7	8	9	5	8
4	5	3	4	0	5	5	7	7	7	10	12	16	14	6	18	16	8	19	8	5	1	3	4	5	9	12
5	4	2	1	5	0	2	2	4	9	7	9	13	11	3	15	13	5	17	7	9	6	8	9	10	6	9
6	4	2	1	5	2	0	4	2	7	5	7	11	9	1	13	11	3	14	7	9	6	8	9	9	4	7
7	6	4	3	6	2	4	0	6	11	8	10	15	13	6	17	15	7	18	9	11	8	10	11	12	8	11
8	6	4	3	7	4	2	6	0	5	7	9	13	11	3	11	13	1	13	9	11	8	9	8	7	2	5
9	11	9	8	7	9	7	11	5	0	12	14	14	16	8	12	17	4	14	8	6	6	4	3	2	3	6
10	9	7	6	10	7	5	9	7	12	0	2	6	4	6	8	6	8	9	12	14	11	13	14	14	9	12
11	11	9	8	12	9	7	11	9	14	2	0	8	2	8	7	4	10	7	14	16	13	15	16	16	11	13
12	15	13	12	16	13	11	15	13	14	6	8	0	10	12	2	12	12	4	18	20	17	18	17	16	11	8
13	13	11	10	14	11	9	13	11	16	4	2	9	0	10	7	2	12	5	16	18	15	17	18	19	13	13
14	5	3	2	6	3	1	5	3	8	6	8	12	10	0	14	12	4	16	8	10	7	9	10	10	5	8
15	17	15	14	18	15	13	17	11	9	8	7	2	7	14	0	5	10	2	20	18	18	16	15	14	9	6
16	15	13	12	16	13	12	16	13	18	6	4	7	2	12	5	0	14	3	18	20	17	19	20	19	14	11
17	7	5	4	8	5	3	7	1	4	8	10	14	12	4	10	15	0	12	10	10	9	8	7	6	1	4
18	18	15	16	19	17	14	18	13	14	9	9	4	5	16	2	3	12	0	21	19	20	18	17	16	11	8
19	7	5	6	8	7	7	9	9	14	12	14	18	14	8	20	18	10	21	0	2	6	4	5	6	11	14
20	9	7	8	5	9	9	11	11	6	14	16	20	18	10	18	20	10	19	2	0	4	2	3	4	9	12
21	6	4	5	1	6	6	8	8	6	11	13	17	15	7	18	17	9	20	6	4	0	2	3	4	9	12
22	8	6	7	3	8	8	10	9	4	13	15	18	17	9	16	19	8	18	4	2	2	0	1	2	7	10
23	9	7	8	4	9	9	11	8	3	14	16	17	18	10	15	20	7	17	5	3	3	1	0	1	6	9
24	10	8	9	5	10	9	12	7	2	14	16	16	19	10	14	19	6	16	6	4	4	2	1	0	5	8
25	8	6	5	9	6	4	7	2	3	9	11	11	13	5	9	14	1	11	11	9	9	7	6	5	0	3
26	11	9	8	12	9	7	11	5	6	12	13	18	13	8	6	11	4	8	14	12	12	10	9	8	3	0

图 5-1 第 i 处到第 j 处巡检点所需最短时间汇总表

附件 2: (问题一 lingo 程序)

```

LINGO 11.0 - [LINGO Model - xj]
File Edit LINGO Window Help

model:
SETS:
jiedian/1..26/:a,qz,qz1,ti,T;
ren/1..26/:rs;
link(jiedian,jiedian)::tt;
ENDSETS
DATA:
tt=0 2 3 5 4 4 6 6 11 9 11 15 13 5 17 15 7 18 7 9 6 8 9 10 8 11
2 0 1 3 2 2 4 4 9 7 9 13 11 3 15 13 5 15 5 7 4 6 7 8 6 9
3 1 0 4 1 1 3 3 8 6 8 12 10 2 14 12 4 16 6 8 5 7 8 9 5 8
5 3 4 0 5 5 7 7 7 10 12 16 14 6 18 16 8 19 8 5 1 3 4 5 9 12
4 2 1 5 0 2 2 4 9 7 9 13 11 3 15 13 5 17 7 9 6 8 9 10 6 9
4 2 1 5 2 0 4 2 7 5 7 11 9 1 13 11 3 14 7 9 6 8 9 9 4 7
6 4 3 6 2 4 0 6 11 8 10 15 13 6 17 15 7 18 9 11 8 10 11 12 8 11
6 4 3 7 4 2 6 0 5 7 9 13 11 3 11 13 1 13 9 11 8 9 8 7 2 5
11 9 8 7 9 7 11 5 0 12 14 14 16 8 12 17 4 14 8 6 6 4 3 2 3 6
9 7 6 10 7 5 9 7 12 0 2 6 4 6 8 6 8 9 12 14 11 13 14 14 9 12
11 9 8 12 9 7 11 9 14 2 0 8 2 8 7 4 10 7 14 16 13 15 16 16 11 13
15 13 12 16 13 11 15 13 14 6 8 0 10 12 2 12 12 4 18 20 17 18 17 16 11 8
13 11 10 14 11 9 13 11 16 4 2 9 0 10 7 2 12 5 16 18 15 17 18 19 13 13
5 3 2 6 3 1 5 3 8 6 8 12 10 0 14 12 4 16 8 10 7 9 10 10 5 8
17 15 14 18 15 13 17 11 9 8 7 2 7 14 0 5 10 2 20 18 18 16 15 14 9 6
15 13 12 16 13 12 16 13 18 6 4 7 2 12 5 0 14 3 18 20 17 19 20 19 14 11
7 5 4 8 5 3 7 1 4 8 10 14 12 4 10 15 0 12 10 10 9 8 7 6 1 4
18 15 16 19 17 14 18 13 14 9 9 4 5 16 2 3 12 0 21 19 20 18 17 16 11 8
7 5 6 8 7 7 9 9 14 12 14 18 14 8 20 18 10 21 0 2 6 4 5 6 11 14
9 7 8 5 9 9 11 11 6 14 16 20 18 10 18 20 10 19 2 0 4 2 3 4 9 12
6 4 5 1 6 6 8 8 6 11 13 17 15 7 18 17 9 20 6 4 0 2 3 4 9 12
8 6 7 3 8 8 8 10 9 4 13 15 18 17 9 16 19 8 18 4 2 2 0 1 2 7 10
9 7 8 4 9 9 11 8 3 14 16 17 18 10 15 20 7 17 5 3 3 1 0 1 6 9
10 8 9 5 10 9 12 7 2 14 16 16 19 10 14 19 6 16 6 4 4 2 1 0 5 8
8 6 5 9 6 4 7 2 3 9 11 11 13 5 9 14 1 11 11 9 9 7 6 5 0 3
11 9 8 12 9 7 11 5 6 12 13 18 13 8 6 11 4 8 14 12 12 10 9 8 3 0;
ti=3 2 3 2 2 3 2 3 4 2 3 2 5 3 2 3 2 2 2 3 3 2 3 2 2 2;
T=35 50 35 35 720 35 80 35 35 120 35 35 80 35 35 480 35 35 80 35 35 120 35;
ENDDATA
min=@sum(jiedian(i):@sum(jiedian(j):@sum(ren(k):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+T(i)+(1-a(k,22)*9)))));
@for(ren(k):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+(1-a(k,22))))>=470);
@for(ren(k):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+(1-a(k,22))))<=480);
@for(jiedian(i):@for(ren(k):a(k,i)=480/T(i));
@for(ren(k):@sum(jiedian(i):a(i,j))=qz(i)+1);
@for(ren(k):@for(jiedian(i):@for(jiedian(j):a(k,i)*(tt(i,j)+ti(i))=T(j)));
@for(ren(k):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)+ti(i))-@for(ren(k):@sum(jiedian(i):@sum(jiedian(j):a((k+5),i)*a((k+5),i)*(tt(1,j)+ti(i))))<=10;
@bin(a(i));@gin(qz(i));@gin(qz1(i));

```

附件 3: (问题二 lingo 程序)

```

LINGO 11.0 - [LINGO Model - xj]
File Edit LINGO Window Help

odel:
ETS:
iedian/1..26/:a,qz,qz1,ti,T,p;
en/1..26/:rs;
ink(jiedian,jiedian)//:tt,a;
NDSETS
ATA:
t=0 2 3 5 4 4 6 6 11 9 11 15 13 5 17 15 7 18 7 9 6 8 9 10 8 11
0 1 3 2 2 4 4 9 7 9 13 11 3 15 13 5 15 5 7 4 6 7 8 6 9
1 0 4 1 1 3 3 8 6 8 12 10 2 14 12 4 16 6 8 5 7 8 9 5 8
3 4 0 5 5 7 7 7 10 12 16 14 6 18 16 8 19 8 5 1 3 4 5 9 12
2 1 5 0 2 2 4 9 7 9 13 11 3 15 13 5 17 7 9 6 8 9 10 6 9
2 1 5 2 0 4 2 7 5 7 11 9 1 13 11 3 14 7 9 6 8 9 9 4 7
4 3 6 2 4 0 6 11 8 10 15 13 6 17 15 7 18 9 11 8 10 11 12 8 11
4 3 7 4 2 6 0 5 7 9 13 11 3 11 13 1 13 9 11 8 9 8 7 2 5
1 9 8 7 9 7 11 5 0 12 14 14 16 8 12 17 4 14 8 6 6 4 3 2 3 6
7 6 10 7 5 9 7 12 0 2 6 4 6 8 6 8 9 12 14 11 13 14 14 9 12
1 9 8 12 9 7 11 9 14 2 0 8 2 8 7 4 10 7 14 16 13 15 16 11 13
5 13 12 16 13 11 15 13 14 6 8 0 10 12 2 12 12 4 18 20 17 18 17 16 11 8
3 11 10 14 11 9 13 11 16 4 2 9 0 10 7 2 12 5 16 18 15 17 18 19 13 13
3 2 6 3 1 5 3 8 6 8 12 10 0 14 12 4 16 8 10 7 9 10 10 5 8
7 15 14 18 15 13 17 11 9 8 7 2 7 14 0 5 10 2 20 18 18 16 15 14 9 6
5 13 12 16 13 12 16 13 18 6 4 7 2 12 5 0 14 3 18 20 17 19 20 19 14 11
5 4 8 5 3 7 1 4 8 10 14 12 4 10 15 0 12 10 10 9 8 7 6 1 4
8 15 16 19 17 14 18 13 14 9 9 4 5 16 2 3 12 0 21 19 20 18 17 16 11 8
5 6 8 7 7 9 9 14 12 14 18 14 8 20 18 10 21 0 2 6 4 5 6 11 14
7 8 5 9 9 11 11 6 14 16 20 18 10 18 20 10 19 2 0 4 2 3 4 9 12
4 5 1 6 6 8 8 6 11 13 17 15 7 18 17 9 20 6 4 0 2 3 4 9 12
6 7 3 8 8 10 9 4 13 15 18 17 9 16 19 8 18 4 2 2 0 1 2 7 10
7 8 4 9 9 11 8 3 14 16 17 18 10 15 20 7 17 5 3 3 1 0 1 6 9
0 8 9 5 10 9 12 7 2 14 16 16 19 10 14 19 6 16 6 4 4 2 1 0 5 8
6 5 9 6 4 7 2 3 9 11 11 13 5 9 14 1 11 11 9 9 7 6 5 0 3
1 9 8 12 9 7 11 5 6 12 13 18 13 8 6 11 4 8 14 12 12 10 9 8 3 0;
i=3 2 3 2 2 3 2 3 4 2 3 2 5 3 2 3 2 2 2 3 3 2 3 2 2 2;
=35 50 35 35 720 35 80 35 35 120 35 35 80 35 35 35 480 35 35 35 80 35 35 120 35;
NDDATA
in=@sum(jiedian(i):@sum(jiedian(j):@sum(ren(k):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+T(i)+(1-a(k,22)*9)))));
for(ren(k):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+(1-a(k,22))))>=440;
for(ren(k):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+(1-a(k,22))))<=450;
for(ren(k):@sum(jiedian(i):@sum(jiedian(p):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+(1-a(k,22))))>=110;
for(ren(k):@sum(jiedian(i):@sum(jiedian(p):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+(1-a(k,22))))<=120;
for(jiedian(i):@for(ren(k):a(k,i)=480/T(i));
for(ren(k):@sum(jiedian(i):a(i,j))=qz(i)+1;
for(ren(k):@for(jiedian(i):@for(jiedian(j):a(k,i)*(tt(i,j)+ti(i))=T(j));
for(ren(k+s):@for(jiedian(m):(tt(m,j)+ti(m)=Tj;
for(ren(k):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)+ti(i))-@for(ren(k):@sum(jiedian(i):@sum(jiedian(j):a((k+s),i)*a((k+s),i)*(tt(l,j)+ti(i))))<=10;
bin(a(i));@gin(qz(i));@gin(qz1(i));m#ne#i;p#ne#j;

```

附件 4: (问题三-----lingo 程序 1)

```

LINGO 11.0 - [LINGO Model - xj]
File Edit LINGO Window Help

model:
SETS:
jiedian/1..26/:a,qz,qz1,ti,Te,T;
ren/1..26/:rs;
banci/1,2,3/:x;
link(jiedian,jiedian)::tt;
ENDSETS
DATA:
tt=0 2 3 5 4 4 6 6 11 9 11 15 13 5 17 15 7 18 7 9 6 8 9 10 8 11
2 0 1 3 2 2 4 4 9 7 9 13 11 3 15 13 5 15 5 7 4 6 7 8 6 9
3 1 0 4 1 1 3 3 8 6 8 12 10 2 14 12 4 16 6 8 5 7 8 9 5 8
5 3 4 0 5 5 7 7 7 10 12 16 14 6 18 16 8 19 8 5 1 3 4 5 9 12
4 2 1 5 0 2 2 4 9 7 9 13 11 3 15 13 5 17 7 9 6 8 9 10 6 9
4 2 1 5 2 0 4 2 7 5 7 11 9 1 13 11 3 14 7 9 6 8 9 9 4 7
6 4 3 6 2 4 0 6 11 8 10 15 13 6 17 15 7 18 9 11 8 10 11 12 8 11
6 4 3 7 4 2 6 0 5 7 9 13 11 3 11 13 1 13 9 11 8 9 8 7 2 5
11 9 8 7 9 7 11 5 0 12 14 14 16 8 12 17 4 14 8 6 6 4 3 2 3 6
9 7 6 10 7 5 9 7 12 0 2 6 4 6 8 6 8 9 12 14 11 13 14 14 9 12
11 9 8 12 9 7 11 9 14 2 0 8 2 8 7 4 10 7 14 16 13 15 16 16 11 13
15 13 12 16 13 11 15 13 14 6 8 0 10 12 2 12 12 4 18 20 17 18 17 16 11 8
13 11 10 14 11 9 13 11 16 4 2 9 0 10 7 2 12 5 16 18 15 17 18 19 13 13
5 3 2 6 3 1 5 3 8 6 8 12 10 0 14 12 4 16 8 10 7 9 10 10 5 8
17 15 14 18 15 13 17 11 9 8 7 2 7 14 0 5 10 2 20 18 18 16 15 14 9 6
15 13 12 16 13 12 16 13 18 6 4 7 2 12 5 0 14 3 18 20 17 19 20 19 14 11
7 5 4 8 5 3 7 1 4 8 10 14 12 4 10 15 0 12 10 10 9 8 7 6 1 4
18 15 16 19 17 14 18 13 14 9 9 4 5 16 2 3 12 0 21 19 20 18 17 16 11 8
7 5 6 8 7 7 9 9 14 12 14 18 14 8 20 18 10 21 0 2 6 4 5 6 11 14
9 7 8 5 9 9 11 11 6 14 16 20 18 10 18 20 10 19 2 0 4 2 3 4 9 12
6 4 5 1 6 6 8 8 6 11 13 17 15 7 18 17 9 20 6 4 0 2 3 4 9 12
8 6 7 3 8 8 10 9 4 13 15 18 17 9 16 19 8 18 4 2 2 0 1 2 7 10
9 7 8 4 9 9 11 8 3 14 16 17 18 10 15 20 7 17 5 3 3 1 0 1 6 9
10 8 9 5 10 9 12 7 2 14 16 16 19 10 14 19 6 16 6 4 4 2 1 0 5 8
8 6 5 9 6 4 7 2 3 9 11 11 13 5 9 14 1 11 11 9 9 7 6 5 0 3
11 9 8 12 9 7 11 5 6 12 13 18 13 8 6 11 4 8 14 12 12 10 9 8 3 0;
ti=3 2 3 2 2 3 2 3 4 2 3 2 5 3 2 3 2 2 2 3 3 2 3 2 2 2;
T=35 50 35 35 720 35 80 35 120 35 35 80 35 35 480 35 35 80 35 35 120 35;
ENDDATA
min=@for(banci(x)):@sum(jiedian(i):@sum(jiedian(j):@sum(ren(k):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+T(i)+(1-a(k,22)*9))))=z;
min=@for(banci(x):@sum(ren(k):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+(1-a(k,22)))))*9=q(x);
min=@sum(banci(x):q(x));
min=@sum(v(x)-v(x-1));
@sum(banci(x):t(x)<=1440;
@sum(banci(x):t(x)>=1410;
@for(jiedian(i):@for(ren(k):@sum(banci(x):a(k,i)=480/T(i));
@for(ren(k):@sum(jiedian(i):a(i,j))=qz(i)+1);
@for(ren(k):@for(banci(x):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)-@for(banci(x):@sum(jiedian(i):@sum(jiedian(j):a((k+9),i)*(tt(i,j)+ti(i))<=10;
@for(ren(k):@sum(banci(x):@for(jiedian(i):@for(jiedian(j):a(k,i)*(tt(i,j)+ti(i))=T(j));
@bin(a(i));@gin(qz(i));@gin(qz1(i));

```

附件 5：（问题三-----程序 2）

```

LINGO 11.0 - [LINGO Model - santi 1]
File Edit LINGO Window Help

model:
SETS:
jiedian/1..26/:a,qz,qz1,ti,Te,T;
ren/1..26/:rs;
banci/1,2,3/:x;
link(jiedian,jiedian)//:tt;
ENDSETS
DATA:
tt=0 2 3 5 4 4 6 6 11 9 11 15 13 5 17 15 7 18 7 9 6 8 9 10 8 11
2 0 1 3 2 2 4 4 9 7 9 13 11 3 15 13 5 15 5 7 4 6 7 8 6 9
3 1 0 4 1 1 3 3 8 6 8 12 10 2 14 12 4 16 6 8 5 7 8 9 5 8
5 3 4 0 5 5 7 7 7 10 12 16 14 6 18 16 8 19 8 5 1 3 4 5 9 12
4 2 1 5 0 2 2 4 9 7 9 13 11 3 15 13 5 17 7 9 6 8 9 10 6 9
4 2 1 5 2 0 4 2 7 5 7 11 9 1 13 11 3 14 7 9 6 8 9 9 4 7
6 4 3 6 2 4 0 6 11 8 10 15 13 6 17 15 7 18 9 11 8 10 11 12 8 11
6 4 3 7 4 2 6 0 5 7 9 13 11 3 11 13 1 13 9 11 8 9 8 7 2 5
11 9 8 7 9 7 11 5 0 12 14 14 16 8 12 17 4 14 8 6 6 4 3 2 3 6
9 7 6 10 7 5 9 7 12 0 2 6 4 6 8 6 8 9 12 14 11 13 14 14 9 12
11 9 8 12 9 7 11 9 14 2 0 8 2 8 7 4 10 7 14 16 13 15 16 16 11 13
15 13 12 16 13 11 15 13 14 6 8 0 10 12 2 12 12 4 18 20 17 18 17 16 11 8
13 11 10 14 11 9 13 11 16 4 2 9 0 10 7 2 12 5 16 18 15 17 18 19 13 13
5 3 2 6 3 1 5 3 8 6 8 12 10 0 14 12 4 16 8 10 7 9 10 10 5 8
17 15 14 18 15 13 17 11 9 8 7 2 7 14 0 5 10 2 20 18 18 16 15 14 9 6
15 13 12 16 13 12 16 13 18 6 4 7 2 12 5 0 14 3 18 20 17 19 20 19 14 11
7 5 4 8 5 3 7 1 4 8 10 14 12 4 10 15 0 12 10 10 9 8 7 6 1 4
18 15 16 19 17 14 18 13 14 9 9 4 5 16 2 3 12 0 21 19 20 18 17 16 11 8
7 5 6 8 7 7 9 9 14 12 14 18 14 8 20 18 10 21 0 2 6 4 5 6 11 14
9 7 8 5 9 9 11 11 6 14 16 20 18 10 18 20 10 19 2 0 4 2 3 4 9 12
6 4 5 1 6 6 8 8 6 11 13 17 15 7 18 17 9 20 6 4 0 2 3 4 9 12
8 6 7 3 8 8 10 9 4 13 15 18 17 9 16 19 8 18 4 2 2 0 1 2 7 10
9 7 8 4 9 9 11 8 3 14 16 17 18 10 15 20 7 17 5 3 3 1 0 1 6 9
10 8 9 5 10 9 12 7 2 14 16 16 19 10 14 19 6 16 6 4 4 2 1 0 5 8
8 6 5 9 6 4 7 2 3 9 11 11 13 5 9 14 1 11 11 9 9 7 6 5 0 3
11 9 8 12 9 7 11 5 6 12 13 18 13 8 6 11 4 8 14 12 12 10 9 8 3 0;
ti=3 2 3 2 3 2 3 4 2 3 2 5 3 2 3 2 2 3 2 3 2 2 2 2;
T=35 50 35 35 720 35 80 35 35 120 35 35 80 35 35 35 480 35 35 35 80 35 35 120 35;
ENDDATA
min=@for(banci(x)):@sum(jiedian(i):@sum(jiedian(j):@sum(ren(k):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+T(i)+(1-a(k,22))*9-@sum(ren(k):@sum(jiedian(i)):@sum(jiedian(p):a(a(k,i)*a(k,p)*10))))))
min=@for(banci(x):@sum(ren(k):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)+ti(i)+(1-a(k,22)))))*9=q(x);
min=@sum(banci(x):q(x));
min=@sum(v(x)-v(x-1));
@sum(banci(x):t(x)<=1440;
@sum(banci(x):t(x)>=1410;
@for(jiedian(i):@for(ren(k):@sum(banci(x):a(k,i)=480/T(i));
@for(ren(k):@sum(jiedian(i):a(i,j))=qz(i)+1;
@for(ren(k):@for(banci(x):@sum(jiedian(i):@sum(jiedian(j):a(k,i)*a(k,j)*(tt(i,j)-@for(banci(x):@sum(jiedian(i):@sum(jiedian(j):a((k+s),i)*a((k+s),i)*(tt(i,j)+ti(i))<=10;
@for(ren(k):@sum(banci(x):@for(jiedian(i):@for(jiedian(j):a(k,i)*(tt(i,j)+ti(i))=T(j)));
@bin(a(i));@gin(qz(i));@gin(qz1(i));p#ne#j

```

附件 6: (划分区域-----lingo 程序)

```

LINGO 11.0 - [LINGO Model - wwj]
File Edit LINGO Window Help

model:
sets:
jiedian/1..26/:ti;
link(jiedian,jiedian):t,a;
endsets
data:
t=0 2 3 5 4 4 6 6 11 9 11 15 13 5 17 15 7 18 7 9 6 8 9 10 8 11
2 0 1 3 2 2 4 4 9 7 9 13 11 3 15 13 5 15 5 7 4 6 7 8 6 9
3 1 0 4 1 1 3 3 8 6 8 12 10 2 14 12 4 16 6 8 5 7 8 9 5 8
5 3 4 0 5 5 7 7 7 10 12 16 14 6 18 16 8 19 8 5 1 3 4 5 9 12
4 2 1 5 0 2 2 4 9 7 9 13 11 3 15 13 5 17 7 9 6 8 9 10 6 9
4 2 1 5 2 0 4 2 7 5 7 11 9 1 13 11 3 14 7 9 6 8 9 9 4 7
6 4 3 6 2 4 0 6 11 8 10 15 13 6 17 15 7 18 9 11 8 10 11 12 8 11
6 4 3 7 4 2 6 0 5 7 9 13 11 3 11 13 1 13 9 11 8 9 8 7 2 5
11 9 8 7 9 7 11 5 0 12 14 14 16 8 12 17 4 14 8 6 6 4 3 2 3 6
9 7 6 10 7 5 9 7 12 0 2 6 4 6 8 6 8 9 12 14 11 13 14 14 9 12
11 9 8 12 9 7 11 9 14 2 0 8 2 8 7 4 10 7 14 16 13 15 16 16 11 13
15 13 12 16 13 11 15 13 14 6 8 0 10 12 2 12 12 4 18 20 17 18 17 16 11 8
13 11 10 14 11 9 13 11 16 4 2 9 0 10 7 2 12 5 16 18 15 17 18 19 13 13
5 3 2 6 3 1 5 3 8 6 8 12 10 0 14 12 4 16 8 10 7 9 10 10 5 8
17 15 14 18 15 13 17 11 9 8 7 2 7 14 0 5 10 2 20 18 18 16 15 14 9 6
15 13 12 16 13 12 16 13 18 6 4 7 2 12 5 0 14 3 18 20 17 19 20 19 14 11
7 5 4 8 5 3 7 1 4 8 10 14 12 4 10 15 0 12 10 10 9 8 7 6 1 4
18 15 16 19 17 14 18 13 14 9 9 4 5 16 2 3 12 0 21 19 20 18 17 16 11 8
7 5 6 8 7 7 9 9 14 12 14 18 14 8 20 18 10 21 0 2 6 4 5 6 11 14
9 7 8 5 9 9 11 11 6 14 16 20 18 10 18 20 10 19 2 0 4 2 3 4 9 12
6 4 5 1 6 6 8 8 6 11 13 17 15 7 18 17 9 20 6 4 0 2 3 4 9 12
8 6 7 3 8 8 10 9 4 13 15 18 17 9 16 19 8 18 4 2 2 0 1 2 7 10
9 7 8 4 9 9 11 8 3 14 16 17 18 10 15 20 7 17 5 3 3 1 0 1 6 9
10 8 9 5 10 9 12 7 2 14 16 16 19 10 14 19 6 16 6 4 4 2 1 0 5 8
8 6 5 9 6 4 7 2 3 9 11 11 13 5 9 14 1 11 11 9 9 7 6 5 0 3
11 9 8 12 9 7 11 5 6 12 13 18 13 8 6 11 4 8 14 12 12 10 9 8 3 0;
ti=3 2 3 2 2 3 2 3 4 2 3 2 5 3 2 3 2 2 2 3 3 2 3 2 2 2;
enddata
min=@sum(jiedian(i):@sum(jiedian(j):a(i,j)*(t(i,j)+ti(i))));
@sum(jiedian(i):@sum(jiedian(j):a(i,j)*(t(i,j)+ti(i))))>=470;
@sum(jiedian(i):@sum(jiedian(j):a(i,j)*(t(i,j)+ti(i))))<=470;
@for(link(i,j):@bin(a(i,j)));

```

运行的部分程序如下:

```

A( 7, 17)      0.000000      9.000000
A( 7, 18)      0.000000      20.00000
A( 7, 19)      0.000000      11.00000
A( 7, 20)      0.000000      13.00000
A( 7, 21)      1.000000      10.00000
A( 7, 22)      0.000000      12.00000
A( 7, 23)      0.000000      13.00000
A( 7, 24)      0.000000      14.00000
A( 7, 25)      1.000000      10.00000
A( 7, 26)      0.000000      13.00000
A( 8, 1)       0.000000      9.000000
A( 8, 2)       0.000000      7.000000
A( 8, 3)       0.000000      6.000000
A( 8, 4)       1.000000      10.00000
A( 8, 5)       0.000000      7.000000
A( 8, 6)       0.000000      5.000000

```