

2011 高教社杯全国大学生数学建模竞赛

承 诺 书

我们仔细阅读了中国大学生数学建模竞赛的竞赛规则.

我们完全明白, 在竞赛开始后参赛队员不能以任何方式(包括电话、电子邮件、网上咨询等)与队外的任何人(包括指导教师)研究、讨论与赛题有关的问题。

我们知道, 抄袭别人的成果是违反竞赛规则的, 如果引用别人的成果或其他公开的资料(包括网上查到的资料), 必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺, 严格遵守竞赛规则, 以保证竞赛的公正、公平性。如有违反竞赛规则的行为, 我们将受到严肃处理。

我们参赛选择的题号是(从 A/B/C/D 中选择一项填写): B

我们的参赛报名号为(如果赛区设置报名号的话):

所属学校(请填写完整的全名):

参赛队员(打印并签名): 1.

2.

3.

指导教师或指导教师组负责人(打印并签名):

日期: 2011 年 9 月 11 日

赛区评阅编号(由赛区组委会评阅前进行编号):

2011 高教社杯全国大学生数学建模竞赛

编 号 专 用 页

赛区评阅编号（由赛区组委会评阅前进行编号）：

赛区评阅记录（可供赛区评阅时使用）：

评 阅 人										
评 分										
备 注										

全国统一编号（由赛区组委会送交全国前编号）：

全国评阅编号（由全国组委会评阅前进行编号）：

交巡警服务平台的设置与调度

摘要

本题讨论了如何设置交巡警服务平台、各平台的管辖范围以及警务资源调度问题。实质上是关于多目标的优化问题。根据题中所给的条件和问题提取出相关的约束条件和目标函数，建立模型。

对于问题一

1.a 是关于各平台的分配管辖范围问题，首先编程实现 92 个路口节点的标号和连线，用欧式算法求出相邻两路口节点之间的距离，建立 92×92 的邻接矩阵，然后在 matlab 环境下采用 floyd 算法求出任意两个点之间的最短距离，从中提取出 92×20 的矩阵，再引入 0-1 整型规划模型，最后建立以总路程最小为目标函数，以各个平台发案率均衡为约束条件，建立优化模型，使用 Lingo 编程实现区域的自动划分；

1.b 是关于如何封锁 13 个交通要道口，以“一个平台的警力最多封锁一个路口”为约束条件，以“最后到达的警力所花时间的最小值（时间转化为路程）”为目标函数，建立相关模型，求出最优解；

1.c 是要在原有平台数的基础上增加 2—5 个平台，以发案均衡量和出警时间为约束条件，建立模型求出结果，再对结果进行分析适当的增减平台数使目标最优。

对于问题二

2.a 针对全市的具体情况,分析该市现有交巡警服务平台设置方案的合理性。分区内和区外两方面考虑。首先区内分析，类似 A 区的做法，对 B C D E F 各区进行划分平台的管辖范围，再筛选出不合理的平台；其次区外分析，结合各个城区面积和人口的影响，把面积和人口作为权重（采用变异系数赋权法）进而计算各个区所需平台数，与原有平台数相比较筛选出不合理的平台，建立模型得出解决方案。

2.b 在该市地点 P 处发生重大案件，服务平台接到报警后，犯罪嫌疑人已驾车逃跑了 3 分钟。就可以找出逃犯在 3 分钟内逃跑的范围，我们以此范围可以部署 3 道警力防线：

第 1 道防线：以 P 中心点到周边 3 分钟的路程的路口部署警力封锁各个路口，形成第一道封锁圈；

第 2 道防线：由于出警也需要时间，以 P 中心点到周边 $(3+t)$ 分钟的路程的路口部署警力封锁各个路口，形成第二道封锁环；

第 3 道防线：封锁该市的出市区的 17 个交通要道口，防止逃出市区，形成第三道封锁。

三道防线同时封锁，层层围堵，最终抓捕逃犯

关键词: matlab floyd 算法 0-1 整型规划 lingo 编程 变异系数赋权法

一、问题的重述

为了更有效地贯彻实施“有困难找警察”职能，需要在市区的一些交通要道和重要部位设置交巡警服务平台。每个交巡警服务平台的职能和警力配备基本相同。由于警务资源是有限的，根据城市的实际情况与需求就合理调度警务资源、管辖范围设置、交巡警服务平台分配提出了以下问题。

问题一：

1、根据该市中心城区 A 的交通网络和现有的 20 个交巡警服务平台的设置情况及相关的数据信息。请为各交巡警服务平台分配管辖范围，使其在所管辖的范围内出现突发事件时，尽量能在 3 分钟内有交巡警（警车的时速为 60km/h）到达事发地。

2、对于重大突发事件，如何调度全区 20 个交巡警服务平台的警力资源，对进出该区的 13 条交通要道实现快速全封锁。（一个平台的警力最多封锁一个路口）

3、由于现有交巡警服务平台的工作量不均衡和有些地方出警时间过长的实际情况，拟在该区内再增加 2 至 5 个平台，请确定需要增加平台的具体个数和位置。

问题二：

1、针对全市（主城六区 A, B, C, D, E, F）的具体情况，按照设置交巡警服务平台的原则和任务，分析研究该市现有交巡警服务平台设置方案（详见附件）的合理性。如果有明显不合理，请给出解决方案。

2、如果该市地点 P（第 32 个节点）处发生了重大刑事案件，在案发 3 分钟后接到报警，犯罪嫌疑人已驾车逃跑。为了快速搜捕嫌疑犯，请给最佳围堵方案。

二、模型假设

- 1、假设每个路段道路畅通，可以双向行驶，没有堵车现象；
- 2、假设每辆巡警车和犯罪嫌疑人的车行驶中速度保持匀速且车速均为 60km/h；
- 3、假设每辆巡警车到事故现场的路径均为最短路径；

三、符号说明

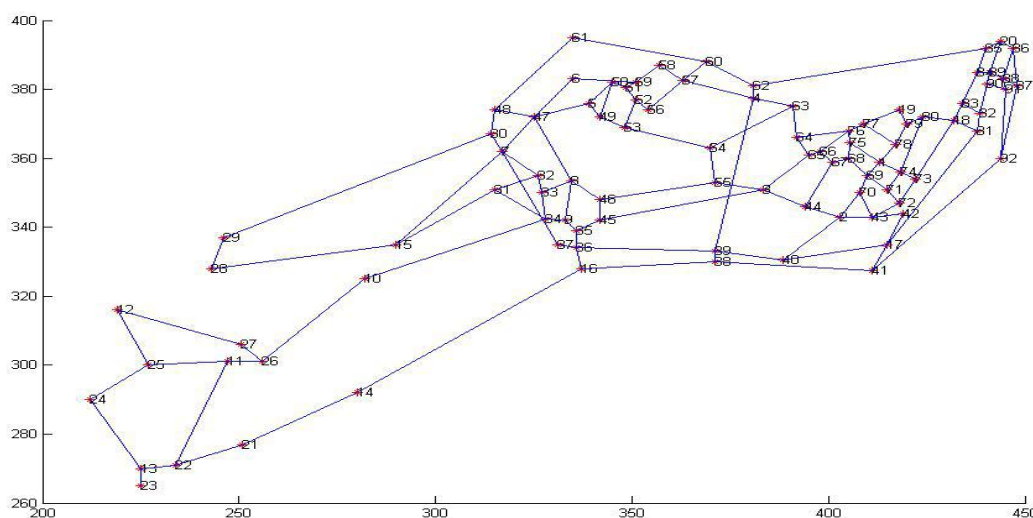
- i 全市第 i 个路口节点
- j 第 j 个交巡警服务平台
- k 第 k 个出入市区的路口节点
- c_i 表示第 i 个路口的发案率
- d_{ij} 第 i 个路口节点到第 j 个交巡警服务平台的最短距离
- a_1 案发率的偏差限
- $W_{\text{总}}$ 92 个交通路口节点的案发率的总和
- a_2 距离的偏差限
- v_m 警车的时速
- v 犯罪嫌疑人的车速
- s_p p 点到全市各出口的距离
- t_j 第 j 个城区所需的平台个数（ $j=1, 2, 3, 4, 5, 6$ ）
- W_1 人均发案率权重
- W_2 人口密度权重
- Z_{ij} 第 i 个影响因素分别对六个城区的影响程度（ $i=1, 2$ ； $j=1, 2, 3, 4, 5, 6$ ）

- e 设计合理方案时的指标系数
 - L_k 第 k 个城区分区后所有距离的平均值 ($k=1,2,3,4,5,6$)
 - m_k 设计合理方案时第 k 个城区距离的限制条件
 - n_k 设计合理方案时每个区可设置的最少平台数
 - r 每个区的路口总数
- 地图距离和实际距离的比例是 1:100000, 即 1 毫米对应 100 米

四、问题分析

1. a 因为每个交巡警服务平台的职能和警力配备基本相同, 所以要考虑每个平台工作量的均衡下能在最短时间内到达突发事件现场, 主要考虑的方向是各个平台管辖范围内的总的时间最短(最短时间可转化为出警的最短路程)与均衡每个平台的发案率这两个因素, 显然, 这是个双目标问题, 为了方便求解, 把双目标函数单一化, 将各个平台发案率的均衡转化为约束条件建立模型, 进而划分出区域。其中, 我们引入了 0-1 规划模型, 采用了 floyd 算法求出图中任意两个站点之间的最短距离, 再根据所建立的模型划分出具体区域。具体做法如下:

1)、首先, 根据附录 2 中 92 个路口节点的横纵坐标, 使用 matlab 编程 (程序见附录 1), 进而将每个节点标号、连线。图形如下:



2)、再用 $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ 公式算出两两之间的距离 (如果有路), 得出 92×92 的邻接矩阵, 其中矩阵中的元素表示两两之间的距离, 若不存在路, 则用一个较大的数代替, 在 matlab 环境下利用 floyd 算法求出两两之间的最短路程和最短路径, 然后从中抽出 92 个节点分别到 20 个服务平台的最短距离。(程序见附录 2)

3)、引入 0-1 整型规划变量, 然后以 92 个节点分别到 20 个服务平台的总的路程最小为目标函数, 以各个平台发案率的均衡为约束条件建立优化模型;

4)、使用 lingo 软件编程, 实现区域的自动划分。(程序见附录 3)

1. b 为了对进出 A 区的交通要道实现快速全封锁, 即以到达路口时最长的

为标准（时间可以转内化为路程），建立目标函数为该标准最小，即最大距离最小化问题，以一个平台的警力最多封锁一个路口为约束条件的模型。利用 lingo 编程从而得出该去交巡警服务平台警力合理的调度方案。（程序见附录 4），

1. c 由于现有交巡警服务平台的工作量不均衡和有些地方出警时间过长，为了使工作量，时间量均衡，题中要求增加 2 至 5 个平台，所以我们建立了以距离，发案率为权值的目标函数，再根据题意建立最优模型，最终得出需要增加的合适的平台个数和位置。

2. a 该题要求按照设置交巡警服务平台的原则和任务，分析研究该市六个主城区现有交巡警服务平台设置方案的合理性。所以要从区内，区外两大方面考虑，整体考虑时人口密度、人均发案率为主要影响因素，我们采用了变异系数赋权法将 2 个影响因素的权重算出，进而列出每个城区所需的平台个数，然后与现有的进行比较，将明显不合理的城区挑出；内部考虑时出警时间、工作量的均衡性为主要影响因素，因此我们先根据 1.a 的模型将另外 5 个城区进行划分（考虑工作量的均衡性），然后在划分结果的基础上筛选指标系数小于 10% 的城区，即为不合理的城区。建立模型将不合理的城区内的服务平台进行适当的增减，重新划分各平台的管辖范围以使得效果最优。

2. b 在该市地点 P 处发生重大案件，服务平台接到报警后，犯罪嫌疑人已驾车逃跑了 3 分钟。就可以找出逃犯在 3 分钟内逃跑的范围，我们以此范围可以部署 3 道警力防线：

第 1 道防线：以 P 中心点到周边 3 分钟的路程的路口部署警力封锁各个路口，形成第一道封锁圈；

第 2 道防线：由于出警也需要时间，同时逃犯还在继续逃跑，就要以 P 中心点到周边(3+t)分钟的路程的路口部署警力封锁各个路口，形成第二道封锁环；

第 3 道防线：封锁该市的出市区的 17 个交通要道口，防止逃出市区，形成第三道封锁。

三道防线同时封锁，层层围堵，最终抓捕逃犯

五、模型的建立与求解

模型的建立：

一、1. a

该题要求为各交巡警服务平台分配管辖范围，由于路程为所花的主要间，要达到各平台以最快速度到达突发事件的地点，所以我们主要考虑路线问题，选择最优路线，因此，我们建立了以最短路程为目标，以服务平台的发案率均衡为限制条件的模型来划分区域。（程序见附录 2）

目标函数：

$$\min = \sum_{i=1}^{92} \sum_{j=1}^{20} d_{ij} * x_{ij}$$

约束条件：

$$x_{ij} = \begin{cases} 1 & \text{第 } i \text{ 个路口节点到第 } j \text{ 个服务平台} \\ 0 & \text{第 } i \text{ 个路口节点不到第 } j \text{ 个服务平台} \end{cases} \quad (i=1,2,\dots,92 \quad j=1,2,\dots,20)$$

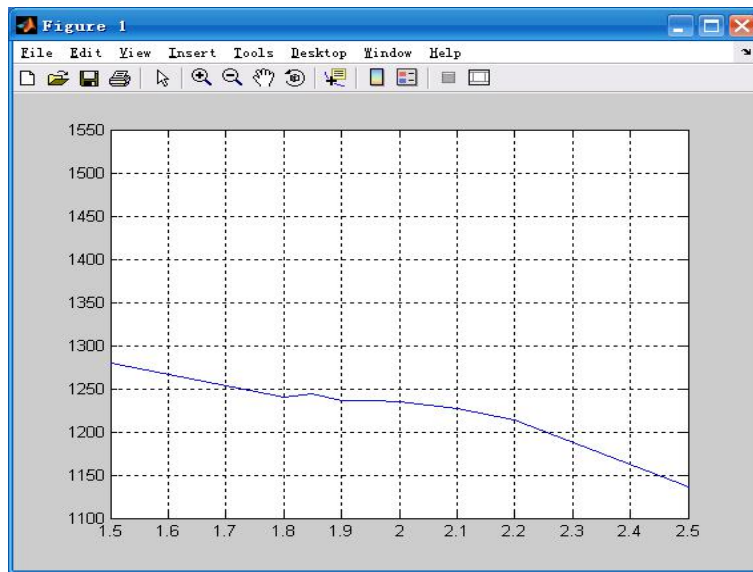
$$x_{ii} = 1 \quad (i=1, 2, \dots, 20)$$

$$\sum_{j=1}^{20} x_{ij} = 1 \quad (i=1, 2, \dots, 92)$$

$$\left| \frac{w_{\text{总}}}{20} - \sum_{i=1}^{92} c_i * x_{ij} \right| < a_1 \quad (j=1, 2, \dots, 20)$$

偏差限的确定:

我们画出了 1.5 到 2.5 之间的所有不同的偏差值与目标最优解的坐标图如下:



由图可看出在 1.9 附近，目标函数值变动最小，为此我们选择 1.9 为偏差限，此时最优目标函数值为：1236.497

求解结果:

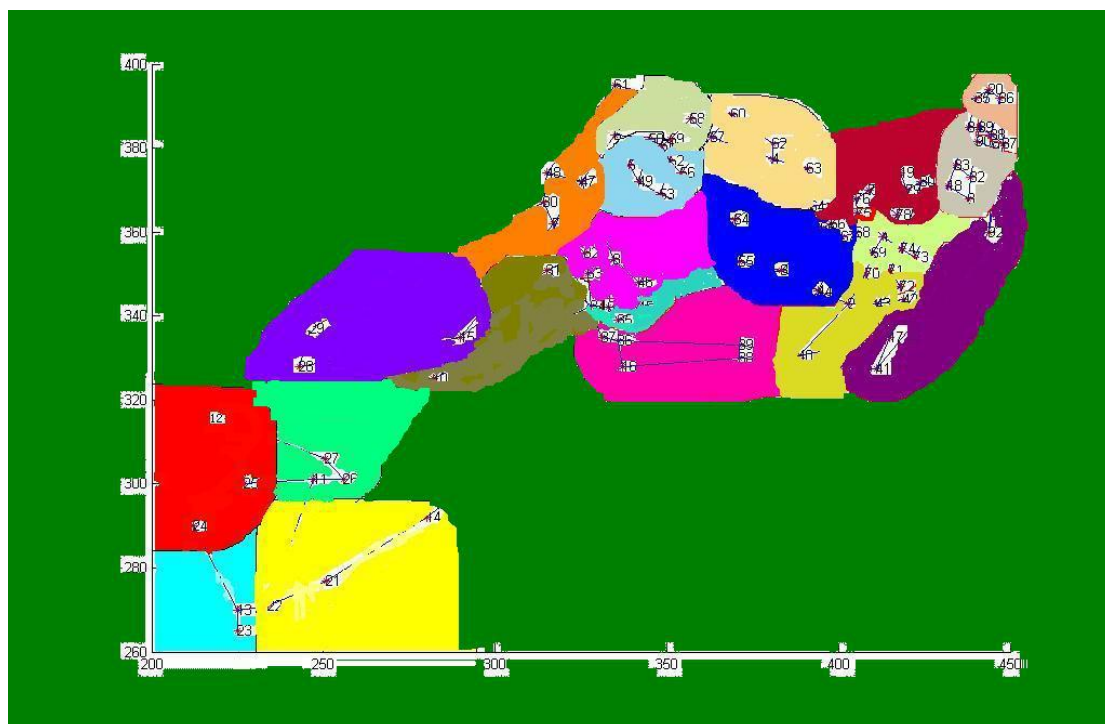
当 $a_1=1.9$ 时，划分结果最优为:

平台 1:	68	69	71	73	74	75
平台 2:	40	43	70	72		
平台 3:	44	54	55	65	66	67
平台 4:	57	60	62	63	64	
平台 5:	49	50	51	52	53	56
平台 6:	58	59				
平台 7:	30	47	48	61		
平台 8:	32	33	46			
平台 9:	35	45				
平台 10:	31	34				
平台 11:	26	27				
平台 12:	24	25				
平台 13:	23					
平台 14:	21	22				

平台 15: 28 29
 平台 16: 36 37 38 39
 平台 17: 41 42 92
 平台 18: 81 82 83 84 90 91
 平台 19: 76 77 78 79 80
 平台 20: 85 86 87 88 89

此时目标函数值为: 1236.497

划分图为:



1. b

该题要求调度 20 个交巡警服务平台的警力资源, 对进出的 13 条交通要道实现快速全封锁, 且一个平台的警力最多封锁一个路口, 所以要求最后一个到达的应该最小, 因此, 建立模型如下 (程序见附录 4):

目标函数:

$$\min = \max(d_{jk} * x_{jk})$$

约束条件:

$$x_{jk} = \begin{cases} 1 & \text{第 } j \text{ 个服务平台到第 } k \text{ 个出入市区的路口节点} \\ 0 & \text{第 } j \text{ 个服务平台不到第 } k \text{ 个出入市区的路口节点} \end{cases}$$

$$(j=1,2,\dots,20; \quad k=1,2,\dots,13)$$

$$\sum_{k=1}^{13} x_{jk} \leq 1 \quad (j=1, 2, \dots, 20)$$

$$\sum_{j=1}^{20} x_{jk} \geq 1 \quad (k=1, 2, \dots, 13)$$

求解结果：

调度封锁方案：

路口	12	14	16	21	22	23	24	28	29	30	38	48	62
平台	11	16	2, 3, 8, 9	14	10	12	13	15	7	5	4	6	1

1. c

该题是要求在原有平台的基础上增加 2 至 5 个，使得改变现有的平台工作量不均衡，时间过长的实际情况，因此我们既要考虑时间（路程），又要考虑发案率，从而建立模型如下：（程序见附录 5）

目标函数：

$$\min = \sum_{i=1}^{92} \sum_{j=1}^{92} c_i d_{ij} x_{ij}$$

约束条件：

$$x_{ij} = \begin{cases} 1 & \text{选择第 } i \text{ 个路口节点到第 } j \text{ 个服务平台} \\ 0 & \text{不选择第 } i \text{ 个路口节点到第 } j \text{ 个服务平台} \end{cases}$$

$$(i=1, 2, \dots, 92 \quad j=1, 2, \dots, 92)$$

$$x_{ii} = 1 \quad (i=1, 2, \dots, 20)$$

$$22 \leq \sum_{i=1}^{92} x_{ii} \leq 25 \quad (i=1, 2, \dots, 20)$$

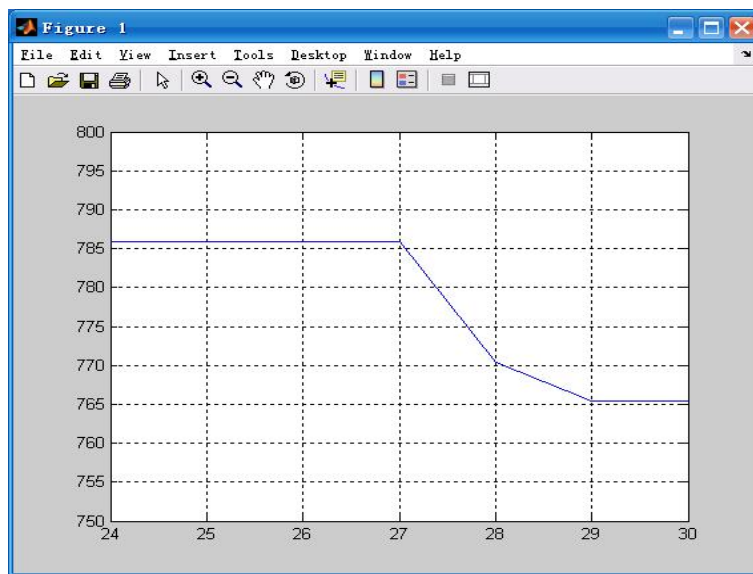
$$\sum_{j=1}^{92} x_{ij} = 1 \quad (i=1, 2, \dots, 92)$$

$$\max(d_{ij} * x_{ij}) \leq a_2 \quad (i=1, 2, \dots, 92 \quad j=1, 2, \dots, 92)$$

$$\text{当 } x_{jj} = 0 \text{ 时, } \sum_{i=1}^{92} x_{ij} = 0, \text{ 否则为 } \sum_{i=1}^{92} x_{ij}$$

求解结果：

从 24 至 30 范围内取出若干个偏差限与所对应的目标函数值，得坐标图如下：



由图可得，当 $a_2=29$ 时，此时最优目标函数值:765. 496

增加 5 个平台，标号与坐标分别为：

- 29 (246, 337)
- 39 (371, 333)
- 48 (315, 374)
- 51 (348.5, 380.5)
- 88 (444.5, 383)

由于该模型发案率的均衡性不是很好，增加的 5 个平台不一定是划分 A 区的最优解。因此要进行进一步的分析处理。将 25 个平台代入 1. a 的模型中，与原 A 区的划分结果相比较（以 $d_{ij}>30$ 的节点个数的多少为标准），逐个去掉平台个数，再重复以上操作，最终得出结果。

使用和求 A 区偏差限相同的方法分别确定增加 5 个平台、4 个平台、3 个平台时的偏差限，即最好均衡性分别为 1. 9、1. 85、1. 85，并设计表格进行对比，表格如下：

增加的点数	均衡性	最优解	时间超过 3 分钟的路口数	增加的平台
5	1. 9	886. 2554	5	29 39 48 51 88
4	1. 85	954. 6909	4	29 39 48 88
3	1. 85	1170. 171	4	29 39 48

从表格中可以看出增加 5 个平台与增加 4 个平台的均衡性都接近时，最优解（即最短距离的路程之和）相差不大，实际中增加一个平台所需要的花费比较大；因此两个中选择增加 4 个平台；增加 3 个平台时在与增加 4 个平台时均衡性一致的前提下最优解太大，由上述可知：

选择增加 4 个平台，标号与坐标为：

- 29 (246, 337)
- 39 (371, 333)
- 48 (315, 374)
- 88 (444.5, 383)

二、2. a

根据设置交巡警服务平台的原则和任务，需要从以下两大方面、四个因素来考虑。

(1) 首先从全市范围内考虑，以人口密度、人均发案率两个影响因素作为权重（各个影响因素在总体因素中的重要程度），为此我们采用了变异系数赋权法求得权重 w_i 。算法如下：

$$\overline{Z}_i = \frac{1}{n} \sum_{j=1}^n Z_{ij}$$

$$S_i = \left[\frac{1}{n-1} \sum_{j=1}^n (Z_{ij} - \overline{Z}_i)^2 \right]^{\frac{1}{2}}$$

$$V_i = S_i / |\overline{Z}_i|$$

$$W_i = V_i / \sum_{i=1}^{10} V_i$$

求解结果：

$$W_1=0.218352$$

$$W_2=0.781648$$

第 j 个区域所需的平台数为 t_j ：

$$t_j = \frac{\sum_{i=1}^2 w_i * z_{ij}}{\sum_{j=1}^6 \sum_{i=1}^2 w_i * z_{ij}} * 80 \quad (i=1, 2; \quad j=1, 2, 3, 4, 5, 6.)$$

根据上面公式，分别计算出每个区所需设置的平台数，并与现有平台数比较判断其合理性，

结果如下图：

	A	B	C	D	E	F
现有平台数	20	8	17	9	15	11
所需平台数	35	11	14	5	7	8

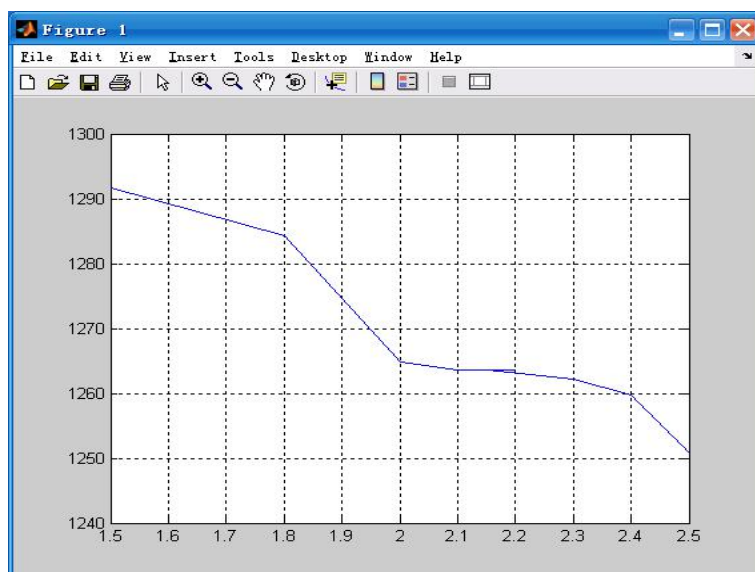
由上图可得 A 区明显不合理

(2) 其次按六个区内分别考虑：

以工作量的均衡性与最短的出警时间两个因素作为其合理性的评判标准。评判标准为 $e=0.1$ 即每个区 90% 的平台出警时间都小于最短出警时间 m_k 就认为其合理。

首先考虑工作量的均衡性，按照 1. a 的模型对 A、B、C、D、E、F 进行划分。划分结果分别为：

B 区：

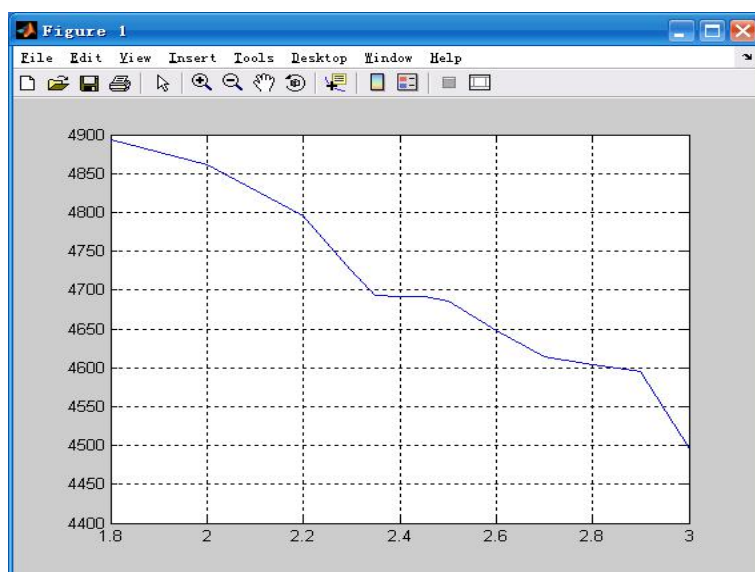


取 2.1 时有最优解：1263.616

B 区划分结果如下：

平台 93: 101 102 103 104 121 156
 平台 94: 105 106 107 108 109 110 111 112 117 118 119 120
 平台 95: 113 114 115 116 123 126 128 129 154 155
 平台 96: 127 128 134 138 139 140 141 145 146 147 150 151
 平台 97: 131 135 137 142 143
 平台 98: 157 158 159 160 161 162 163 164 165
 平台 99: 136 144 148 149 152 153
 平台 100: 122 124 125 132 133

C 区：



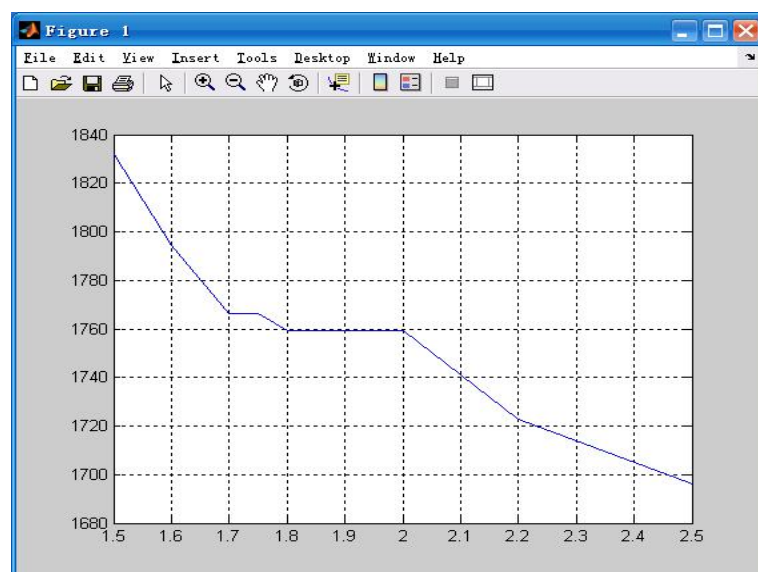
取 2.4 时有最优解：4691.035

C 区划分结果为：

平台 166: 261 262 263 264 265 266

平台 167: 248 249 250 251 252 255 258 259 260
 平台 168: 189 190 191 192 195 232 234
 平台 169: 239 240 253 254 273
 平台 170: 223 224 225 274 275 276 277 278 280 282 283
 平台 171: 216 230 231 241 242 243 244 246
 平台 172: 217 218 226 227 228 229
 平台 173: 233 235 236 237 238 245 247
 平台 174: 211 212 213 214 219 220 221 222
 平台 175: 193 194 196 197 198 215
 平台 176: 183 184 185 186 187 188
 平台 177: 199 200 201 202 206 207 208 210
 平台 178: 203 204 205 209 284 285 286 287 288 301
 平台 179: 279 281 289 290 291 295 296 297 298 299
 平台 180: 269 300 302 303 304 305 306 310 311 312 314 315
 平台 181: 267 268 307 308 309 313 316 317 318 319
 平台 182: 256 257 270 271 272 292 293 294

D 区:

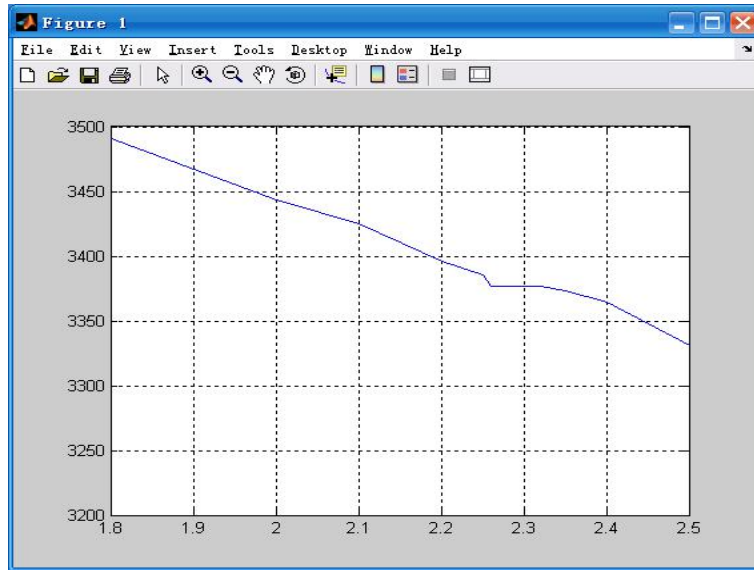


取 1.8 时有最优解: 1759.241

D 区划分结果为:

平台 320: 348 349 350 369 371
 平台 321: 351 353 354 355 356 357 358 370
 平台 322: 359 367 368
 平台 323: 344 345 360 361 362
 平台 324: 364 365 366
 平台 325: 347 363
 平台 326: 343 346 352
 平台 327: 337 338 339 340 341 342
 平台 328: 329 330 331 332 333 334 335 336

E 区:

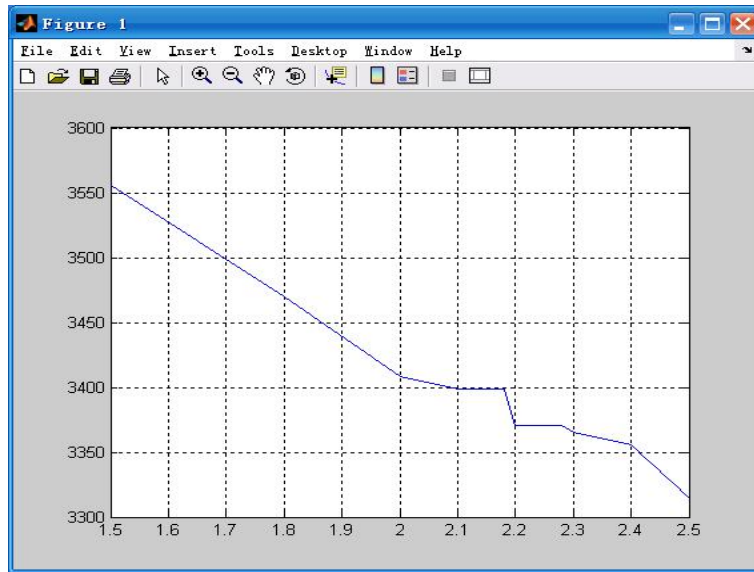


取 2.26 时有最优解: 3376.953

E 区划分结果为:

平台 372: 455 456 462
 平台 373: 437 438 445 446 450 453
 平台 374: 427 428 432 433 434 435 436 437
 平台 375: 424 425 426 429 430 431
 平台 376: 415 423
 平台 377: 411 412 416
 平台 378: 418 458 459
 平台 379: 417 419 420 421 422
 平台 380: 387 388 389 390 391 392 393 394 395 396
 平台 381: 397 398 399 400 405 406 407
 平台 382: 401 402 403 404 407 408 409 413 414
 平台 383: 452 454 460 461 463 464 469 470
 平台 384: 465 466 467 468 471 472
 平台 385: 448 449 451 473 474
 平台 386: 439 440 441 442 443 444 447

F 区:



取 2.2 时有最优解：3371.010

F 区划分结果为：

平台 475: 550 551 554 555 556 557 558 564
 平台 476: 532 533 534 535 544 545 546 547 552 553
 平台 477: 493 494 495 496 497 498 499 500 501 502 503 504 505
 506 507 508 516 519 520
 平台 478: 514 515 522 523 524 527 528 536 538 542 543
 平台 479: 573 575 576 577 578 579 580 581 582
 平台 480: 561 562 563 566 567 574
 平台 481: 490 491 492 517 518 521 529 530 531 548 549
 平台 482: 486 487 488 489 559 560
 平台 483: 509 510 511 512 513 525
 平台 484: 526 537 539 540 541
 平台 485: 565 568 569 570 571 572

求出每个区的除平台以外的节点与平台的距离的平均值，根据 $L_1/30=L_k/m_k$ 公式算出每个区尽可能的最短出警时间 $m_k/10$ ，筛选出每个区最短距离大于 m_k 的路口个数并求出这些个数之和，再用

$$\frac{(d_{ij} * x_{ij} - m_k) \geq 0 \text{ 的节点数之和}}{\text{总的节点个数}}$$
 公式得出 6 个区的结果，并由公式

$$\frac{(d_{ij} * x_{ij} - m_k) \geq 0 \text{ 的节点数之和}}{\text{总的节点个数}} \geq e$$
 筛选出不合理的城区，得出 A、B、D 区

不合理。

(3) 最终建立模型解决方案

建立模型如下：（程序见附录 6）

目标函数：

$$\min = \sum \sum c_i * d_{ij} * x_{ij}$$

约束条件：

$$\sum_{j=1}^{92} x_{ij} = 1 \quad (i=1, 2)$$

$$x_{ij} = \begin{cases} 1 & \text{第} i \text{个路口节点到第} j \text{个服务平台} \\ 0 & \text{第} i \text{个路口节点不到第} j \text{个服务平台} \end{cases}$$

$$\frac{(d_{ij} * x_{ij} - m_k) \geq 0 \text{的节点数之和}}{\text{总的节点个数}} \leq e \quad ;$$

$$\sum_{i=1}^r x_{ii} \leq n_k ;$$

$$\text{若 } x_{ij} = 0 ; \text{ 则 } \sum_{i=1}^{92} x_{ij} = 0 ;$$

求解结果：

A 区增加的平台：21、25、29、32、39、51、66、88

B 区增加的平台：102、113、123、128、142、150、158

D 区增加的平台：333、338、347、357、365、370

2. b

根据题意，为了快速搜捕嫌疑犯，也就是说，各个平台到封锁路口的时间要最短，即最大搜索距离最短，首先求出需要封锁的路口，具体做法为：先计算出嫌疑犯 3 分钟走的路程为 30，再以 P32 点为圆心，以 30 为半径形成一个包围圈，在这个包围圈的 ε 邻域内选出若干个路口，再以这些路口为圆心，10t 为半径形成若干个包围圈，

从而建立模型如下：（程序见附录 7）：

目标函数：

$$\min = \max(d_{ij} * x_{ij})$$

约束条件：

$$x_{ij} = \begin{cases} 1 & \text{选择第} i \text{个路口节点到第} j \text{个服务平台} \\ 0 & \text{不选择第} i \text{个路口节点到第} j \text{个服务平台} \end{cases}$$

$$\max(d_{ij} / v_m * x_{ij}) < \min(s_p / v - 3/60)$$

$$\text{即 } \max(d_{ij} * x_{ij}) < \min(s_p - 30)$$

$$\sum_{j=1}^{563} x_{ij} \leq 1$$

$$\sum_{i=1}^{80} x_{ij} \geq 1$$

求解结果:

路口	平台		
15	173		
151	93	96	382
153	95		
177	177		
202	175		
203	180		
235	16		
236	15		
264	182		
317	178	181	
325	324		
328	327		
332	380		
362	323		
387	100		
418	375		
483	478		
541	476		
572	484		
578	485	479	

此时最短时间为: 12.68027

六、模型的检验

在上述所建立的模型中,所有含有的偏差限的模型,其中的偏差限均为人为给定,则肯定会给模型的求解带来影响,为了减少对模型的影响,我们对偏差限做了较为严格的分析。

以 1. a 为例分析,给偏差限 a_1 若个不同的值,以 a_1 为横坐标,相应的目标函数为纵坐标,画出图形,观察图形中目标函数变动最小的位置,则该点为最优解。同理对其他模型分析。

七、模型的评价与推广

本题的模型有效的解决了合理分配交巡警平台的管辖范围问题,出警时间的合理安排,警力资源的分配以及对各路口的有效封锁问题。整个模型的建立思路清晰,遵循可操作性原则,可比性原则及科学性原则,该模型建立了在较为理想状态下交巡警平台的最优设置,缩短了出警时间,提高了效率。但该模型也有一

定的局限性，如模型建立在理想化的环境中，如道路的畅通性，出警车辆和人员配备的可行性等忽略了生活中存在的不定因素。

在对不合理的交巡警服务平台处理时，可根据实际不同的环境进行不同的修改，如在人口密度较大的地区和案发率较高的地区可安排较多的服务平台，依路口的密集程度来安排警力的多少等修改方法。

本题模型较好的解决了交巡警的出警问题，追捕逃犯的封堵路口的分配问题，在发生事件时能在第一时间出现在现场，有效地提高了交巡警的任职的效率，在科技和经济快速发展的今天，农村城市化的变迁，人口的迅速增长等，治安能力成为城市性能好坏的重要因素，本模型除此之外，还可用于消防救援的最优安排问题，安全事故的应急救援问题，出租车省油的最佳路径问题等现实生活中。总的来说，在实际生活中有很大的利用价值，一定程度上可作为参考。

八、参考文献

- [1]严蔚敏、吴伟民，《数据结构》，清华大学出版社，2006年12月。
- [2]薛毅主编，《数学建模基础》，北京工业大学出版社，2005年1月。
- [3]朱旭、李换琴、籍万新，《MATLAB 软件与基础数学实验》，西安交通大学出版社，2010年7月。
- [4]初春玲，曹步维，周俊彦.室内空气品质模糊综合评价，通风除尘.1999，9-11;
- [5]Chihui Zhu, Nianping Li, Wei Wen. Grey Assessment of Indoor Air Quality. The 4th International Conference on IAQVEC. Changsha, China.2001,81-85;
- [6]张桂芳，汤广发，李念平等.室内空气品质的灰色综合评判，湖南大学学报.2001，107-111;
- [7]沈晋明，毛继传，孙光前.上海办公大楼室内空气品质客观评价,通风除尘.1995，14-17;
- [8] 《设置交巡警服务平台的原则和任务》
<http://zhidao.baidu.com/question/317709495.html> ,2011年9月10日。

九、附录

附录 1(在 matlab 中显示 A 区的图坐标的标号):

```
hold on
plot(x,y,'r*')
text(x(1),y(1),'1');
text(x(2),y(2),'2');
text(x(3),y(3),'3');
text(x(4),y(4),'4');
text(x(5),y(5),'5');
text(x(6),y(6),'6');
text(x(7),y(7),'7');
text(x(8),y(8),'8');
text(x(9),y(9),'9');
text(x(10),y(10),'10');
text(x(11),y(11),'11');
text(x(12),y(12),'12');
text(x(13),y(13),'13');
```

```
text(x(14),y(14),' 14');
text(x(15),y(15),' 15');
text(x(16),y(16),' 16');
text(x(17),y(17),' 17');
text(x(18),y(18),' 18');
text(x(19),y(19),' 19');
text(x(20),y(20),' 20');
text(x(21),y(21),' 21');
text(x(22),y(22),' 22');
text(x(23),y(23),' 23');
text(x(24),y(24),' 24');
text(x(25),y(25),' 25');
text(x(26),y(26),' 26');
text(x(27),y(27),' 27');
text(x(28),y(28),' 28');
text(x(29),y(29),' 29');
text(x(30),y(30),' 30');
text(x(31),y(31),' 31');
text(x(32),y(32),' 32');
text(x(33),y(33),' 33');
text(x(34),y(34),' 34');
text(x(35),y(35),' 35');
text(x(36),y(36),' 36');
text(x(37),y(37),' 37');
text(x(38),y(38),' 38');
text(x(39),y(39),' 39');
text(x(40),y(40),' 40');
text(x(41),y(41),' 41');
text(x(42),y(42),' 42');
text(x(43),y(43),' 43');
text(x(44),y(44),' 44');
text(x(45),y(45),' 45');
text(x(46),y(46),' 46');
text(x(47),y(47),' 47');
text(x(48),y(48),' 48');
text(x(49),y(49),' 49');
text(x(50),y(50),' 50');
text(x(51),y(51),' 51');
text(x(52),y(52),' 52');
text(x(53),y(53),' 53');
text(x(54),y(54),' 54');
text(x(55),y(55),' 55');
text(x(56),y(56),' 56');
text(x(57),y(57),' 57');
```

```

text(x(58),y(58),' 58');
text(x(59),y(59),' 59');
text(x(60),y(60),' 60');
text(x(61),y(61),' 61');
text(x(62),y(62),' 62');
text(x(63),y(63),' 63');
text(x(64),y(64),' 64');
text(x(65),y(65),' 65');
text(x(66),y(66),' 66');
text(x(67),y(67),' 67');
text(x(68),y(68),' 68');
text(x(69),y(69),' 69');
text(x(70),y(70),' 70');
text(x(71),y(71),' 71');
text(x(72),y(72),' 72');
text(x(73),y(73),' 73');
text(x(74),y(74),' 74');
text(x(75),y(75),' 75');
text(x(76),y(76),' 76');
text(x(77),y(77),' 77');
text(x(78),y(78),' 78');
text(x(79),y(79),' 79');
text(x(80),y(80),' 80');
text(x(81),y(81),' 81');
text(x(82),y(82),' 82');
text(x(83),y(83),' 83');
text(x(84),y(84),' 84');
text(x(85),y(85),' 85');
text(x(86),y(86),' 86');
text(x(87),y(87),' 87');
text(x(88),y(88),' 88');
text(x(89),y(89),' 89');
text(x(90),y(90),' 90');
text(x(91),y(91),' 91');
text(x(92),y(92),' 92');
hold on
plot(x,y,'r*')
text(x(1),y(1),' 1');
text(x(2),y(2),' 2');
text(x(3),y(3),' 3');
text(x(4),y(4),' 4');
text(x(5),y(5),' 5');
text(x(6),y(6),' 6');
text(x(7),y(7),' 7');

```

```
text(x(8),y(8),'8');
text(x(9),y(9),'9');
text(x(10),y(10),'10');
text(x(11),y(11),'11');
text(x(12),y(12),'12');
text(x(13),y(13),'13');
text(x(14),y(14),'14');
text(x(15),y(15),'15');
text(x(16),y(16),'16');
text(x(17),y(17),'17');
text(x(18),y(18),'18');
text(x(19),y(19),'19');
text(x(20),y(20),'20');
text(x(21),y(21),'21');
text(x(22),y(22),'22');
text(x(23),y(23),'23');
text(x(24),y(24),'24');
text(x(25),y(25),'25');
text(x(26),y(26),'26');
text(x(27),y(27),'27');
text(x(28),y(28),'28');
text(x(29),y(29),'29');
text(x(30),y(30),'30');
text(x(31),y(31),'31');
text(x(32),y(32),'32');
text(x(33),y(33),'33');
text(x(34),y(34),'34');
text(x(35),y(35),'35');
text(x(36),y(36),'36');
text(x(37),y(37),'37');
text(x(38),y(38),'38');
text(x(39),y(39),'39');
text(x(40),y(40),'40');
text(x(41),y(41),'41');
text(x(42),y(42),'42');
text(x(43),y(43),'43');
text(x(44),y(44),'44');
text(x(45),y(45),'45');
text(x(46),y(46),'46');
text(x(47),y(47),'47');
text(x(48),y(48),'48');
text(x(49),y(49),'49');
text(x(50),y(50),'50');
text(x(51),y(51),'51');
```

```

text(x(52),y(52),'52');
text(x(53),y(53),'53');
text(x(54),y(54),'54');
text(x(55),y(55),'55');
text(x(56),y(56),'56');
text(x(57),y(57),'57');
text(x(58),y(58),'58');
text(x(59),y(59),'59');
text(x(60),y(60),'60');
text(x(61),y(61),'61');
text(x(62),y(62),'62');
text(x(63),y(63),'63');
text(x(64),y(64),'64');
text(x(65),y(65),'65');
text(x(66),y(66),'66');
text(x(67),y(67),'67');
text(x(68),y(68),'68');
text(x(69),y(69),'69');
text(x(70),y(70),'70');
text(x(71),y(71),'71');
text(x(72),y(72),'72');
text(x(73),y(73),'73');
text(x(74),y(74),'74');
text(x(75),y(75),'75');
text(x(76),y(76),'76');
text(x(77),y(77),'77');
text(x(78),y(78),'78');
text(x(79),y(79),'79');
text(x(80),y(80),'80');
text(x(81),y(81),'81');
text(x(82),y(82),'82');
text(x(83),y(83),'83');
text(x(84),y(84),'84');
text(x(85),y(85),'85');
text(x(86),y(86),'86');
text(x(87),y(87),'87');
text(x(88),y(88),'88');
text(x(89),y(89),'89');
text(x(90),y(90),'90');
text(x(91),y(91),'91');
text(x(92),y(92),'92');
plot([x(1);x(75)], [y(1);y(75)]);
plot([x(1);x(78)], [y(1);y(78)]);
plot([x(2);x(44)], [y(2);y(44)]);

```

```
plot([x(3);x(45)], [y(3);y(45)]);
plot([x(3);x(65)], [y(3);y(65)]);
plot([x(4);x(39)], [y(4);y(39)]);
plot([x(4);x(63)], [y(4);y(63)]);
plot([x(5);x(49)], [y(5);y(49)]);
plot([x(5);x(50)], [y(5);y(50)]);
plot([x(6);x(59)], [y(6);y(59)]);
plot([x(7);x(32)], [y(7);y(32)]);
plot([x(7);x(47)], [y(7);y(47)]);
plot([x(2);x(44)], [y(2);y(44)]);
plot([x(8);x(47)], [y(8);y(47)]);
plot([x(8);x(9)], [y(8);y(9)]);
plot([x(9);x(35)], [y(9);y(35)]);
plot([x(10);x(34)], [y(10);y(34)]);
plot([x(11);x(22)], [y(11);y(22)]);
plot([x(11);x(26)], [y(11);y(26)]);
plot([x(12);x(25)], [y(12);y(25)]);
plot([x(14);x(21)], [y(14);y(21)]);
plot([x(15);x(7)], [y(15);y(7)]);
plot([x(15);x(31)], [y(15);y(31)]);
plot([x(16);x(14)], [y(16);y(14)]);
plot([x(16);x(38)], [y(16);y(38)]);
plot([x(17);x(40)], [y(17);y(40)]);
plot([x(17);x(42)], [y(17);y(42)]);
plot([x(17);x(81)], [y(17);y(81)]);
plot([x(18);x(81)], [y(18);y(81)]);
plot([x(18);x(83)], [y(18);y(83)]);
plot([x(19);x(79)], [y(19);y(79)]);
plot([x(20);x(86)], [y(20);y(86)]);
plot([x(21);x(22)], [y(21);y(22)]);
plot([x(22);x(13)], [y(22);y(13)]);
plot([x(23);x(13)], [y(23);y(13)]);
plot([x(24);x(13)], [y(24);y(13)]);
plot([x(24);x(25)], [y(24);y(25)]);
plot([x(25);x(11)], [y(25);y(11)]);
plot([x(26);x(27)], [y(26);y(27)]);
plot([x(26);x(10)], [y(26);y(10)]);
plot([x(27);x(12)], [y(27);y(12)]);
plot([x(28);x(29)], [y(28);y(29)]);
plot([x(28);x(15)], [y(28);y(15)]);
plot([x(29);x(30)], [y(29);y(30)]);
plot([x(30);x(7)], [y(30);y(7)]);
plot([x(30);x(48)], [y(30);y(48)]);
plot([x(31);x(32)], [y(31);y(32)]);
```

```

plot([x(31);x(34)], [y(31);y(34)]);
plot([x(32);x(33)], [y(32);y(33)]);
plot([x(33);x(34)], [y(33);y(34)]);
plot([x(33);x(8)], [y(33);y(8)]);
plot([x(34);x(9)], [y(34);y(9)]);
plot([x(35);x(45)], [y(35);y(45)]);
plot([x(36);x(35)], [y(36);y(35)]);
plot([x(36);x(37)], [y(36);y(37)]);
plot([x(36);x(16)], [y(36);y(16)]);
plot([x(36);x(39)], [y(36);y(39)]);
plot([x(37);x(7)], [y(37);y(7)]);
plot([x(38);x(39)], [y(38);y(39)]);
plot([x(38);x(41)], [y(38);y(41)]);
plot([x(39);x(40)], [y(39);y(40)]);
plot([x(40);x(2)], [y(40);y(2)]);
plot([x(41);x(17)], [y(41);y(17)]);
plot([x(41);x(92)], [y(41);y(92)]);
plot([x(42);x(43)], [y(42);y(43)]);
plot([x(43);x(2)], [y(43);y(2)]);
plot([x(43);x(72)], [y(43);y(72)]);
plot([x(3);x(65)], [y(3);y(65)]);
plot([x(44);x(3)], [y(44);y(3)]);
plot([x(45);x(46)], [y(45);y(46)]);
plot([x(28);x(29)], [y(28);y(29)]);
plot([x(46);x(8)], [y(46);y(8)]);
plot([x(46);x(55)], [y(46);y(55)]);
plot([x(47);x(48)], [y(47);y(48)]);
plot([x(47);x(5)], [y(47);y(5)]);
plot([x(47);x(6)], [y(47);y(6)]);
plot([x(31);x(34)], [y(31);y(34)]);
plot([x(48);x(61)], [y(48);y(61)]);
plot([x(49);x(50)], [y(49);y(50)]);
plot([x(49);x(53)], [y(49);y(53)]);
plot([x(50);x(51)], [y(50);y(51)]);
plot([x(51);x(52)], [y(51);y(52)]);
plot([x(51);x(59)], [y(51);y(59)]);
plot([x(52);x(56)], [y(52);y(56)]);
plot([x(53);x(52)], [y(53);y(52)]);
plot([x(53);x(54)], [y(53);y(54)]);
plot([x(54);x(55)], [y(54);y(55)]);
plot([x(54);x(63)], [y(54);y(63)]);
plot([x(55);x(3)], [y(55);y(3)]);
plot([x(56);x(57)], [y(56);y(57)]);
plot([x(57);x(58)], [y(57);y(58)]);

```



```
plot([x(57);x(60)], [y(57);y(60)]);
plot([x(57);x(4)], [y(57);y(4)]);
plot([x(58);x(59)], [y(58);y(59)]);
plot([x(60);x(62)], [y(60);y(62)]);
plot([x(61);x(60)], [y(61);y(60)]);
plot([x(62);x(4)], [y(62);y(4)]);
plot([x(62);x(85)], [y(62);y(85)]);
plot([x(63);x(64)], [y(63);y(64)]);
plot([x(64);x(65)], [y(64);y(65)]);
plot([x(64);x(76)], [y(64);y(76)]);
plot([x(65);x(66)], [y(65);y(66)]);
plot([x(66);x(67)], [y(66);y(67)]);
plot([x(66);x(76)], [y(66);y(76)]);
plot([x(67);x(44)], [y(67);y(44)]);
plot([x(67);x(68)], [y(67);y(68)]);
plot([x(68);x(69)], [y(68);y(69)]);
plot([x(68);x(75)], [y(68);y(75)]);
plot([x(69);x(70)], [y(69);y(70)]);
plot([x(69);x(71)], [y(69);y(71)]);
plot([x(69);x(1)], [y(69);y(1)]);
plot([x(70);x(2)], [y(70);y(2)]);
plot([x(70);x(43)], [y(70);y(43)]);
plot([x(70);x(72)], [y(70);y(72)]);
plot([x(71);x(72)], [y(71);y(72)]);
plot([x(71);x(74)], [y(71);y(74)]);
plot([x(72);x(73)], [y(72);y(73)]);
plot([x(73);x(74)], [y(73);y(74)]);
plot([x(73);x(18)], [y(73);y(18)]);
plot([x(74);x(1)], [y(74);y(1)]);
plot([x(74);x(80)], [y(74);y(80)]);
plot([x(75);x(76)], [y(75);y(76)]);
plot([x(76);x(77)], [y(76);y(77)]);
plot([x(77);x(78)], [y(77);y(78)]);
plot([x(77);x(19)], [y(77);y(19)]);
plot([x(78);x(79)], [y(78);y(79)]);
plot([x(79);x(80)], [y(79);y(80)]);
plot([x(80);x(18)], [y(80);y(18)]);
plot([x(81);x(82)], [y(81);y(82)]);
plot([x(82);x(83)], [y(82);y(83)]);
plot([x(82);x(90)], [y(82);y(90)]);
plot([x(83);x(84)], [y(83);y(84)]);
plot([x(84);x(85)], [y(84);y(85)]);
plot([x(85);x(20)], [y(85);y(20)]);
plot([x(86);x(87)], [y(86);y(87)]);
```

```

plot([x(86);x(88)], [y(86);y(88)]);
plot([x(87);x(88)], [y(87);y(88)]);
plot([x(87);x(92)], [y(87);y(92)]);
plot([x(88);x(89)], [y(88);y(89)]);
plot([x(88);x(91)], [y(88);y(91)]);
plot([x(89);x(20)], [y(89);y(20)]);
plot([x(89);x(84)], [y(89);y(84)]);
plot([x(89);x(90)], [y(89);y(90)]);
plot([x(90);x(91)], [y(90);y(91)]);
plot([x(91);x(92)], [y(91);y(92)]);
hold off

```

附录 2(floyd 算法求最短路径及距离)

建立带权邻接矩阵 A:

```

p=xlsread('1.xls');
x=p(:,1);
y=p(:,2);
i=p(:,3);
j=p(:,4);
A=zeros(92,92);
for k=1:140
    a=i(k);
    b=j(k);
    A(a,b)=1;
    A(b,a)=1;
end
for m=1:92
    for n=1:92
        if(A(m,n)==1)
            A(m,n)=sqrt((x(m)-x(n))^2+(y(m)-y(n))^2);
        else
            A(m,n)=5000;
        end
    end
end
end
[D,path]=floyd(A);
floyd 算法:
function [D,R]=floyd(a)
n=size(a,1);
D=a;
for i=1:n
    for j=1:n
        R{i,j}=[num2str(i), num2str(j)];
    end
end
end

```

```

for k=1:n
    for i=1:n
        for j=1:n
            if D(i,k)+D(k,j)<D(i,j)
                D(i,j)=D(i,k)+D(k,j);
                l=length(R{i,k})-length(num2str(k));
                R{i,j}=[R{i,k}(1:l),R{k,j}];
            end
        end
    end
end
end

```

附录 3（用 lingo 划分 A 区域）:

```

model:
sets:
department/1..92/;
type/1..20/;
a/1..92/:c;
benefit(department,type):d,x;
endsets
min=@sum(benefit(i,j):d(i,j)*x(i,j));
@for(benefit:@bin(x));
@for(department(i):
    @sum(type(j):x(i,j))=1);
@for(type(j):(6.225-@sum(a(i):x(i,j)*c(i)))<1.9);
@for(type(j):(6.225-@sum(a(i):x(i,j)*c(i)))>-1.9);
@for(type(i):x(i,i)=1);
data:
d=@ole('G:/2011B/shortlength.xls',data1);
c=@ole('G:/2011B/shortlength.xls',data2);
enddata
end

```

data1 为 92 个路口分别到 20 个平台的最短距离，
data2 为 92 个路口事件的发生概率。

附录 4（用 lingo 设计 A 区最佳封锁方案）:

```

model:
sets:
department/1..20/;
type/1..13/;
a/1..20/:c;
benefit(department,type):d,x;
endsets
min=@max(benefit(i,j):d(i,j)*x(i,j));
@for(benefit:@bin(x));
@for(department(i):

```

```

        @sum(type(j):x(i,j))<1);
@for(type(j):
    @sum(department(i):x(i,j))>1);
data:
d=@ole('G:/2011B/2.xls',data1);
enddata
end
data1 为 20 个平台到 13 个出入口的最短距离。

```

附录 5（用 lingo 设计增加的平台个数）:

```

model:
sets:
type/1..20/;
a/1..92/:c;
benefit(a,a):d,x;
endsets
min=@sum(benefit(i,j):d(i,j)*x(i,j)*c(i));
@for(benefit:@bin(x));
@sum(a(i):x(i,i))>=22;
@sum(a(i):x(i,i))<=25;
@for(a(i):
    @sum(a(j):x(i,j))=1);
@for(a(i):
    @for(a(j):(x(i,j)*d(i,j))<29));
@for(a(j):@sum(a(i):x(i,j))=@if(x(j,j)#eq#0,0,@sum(a(i):x(i,j))));
@for(type(i):x(i,i)=1);
data:
d=@ole('G:/short.xls',data1);
c=@ole('G:/short.xls',data2);
enddata
end
data1为92*92的两两节点的最短距离,
data2为92个节点的发案率。

```

附录 6（用lingo进行合理性划分区域）:

```

model:
sets:
type/1..20/;
a/1..92/:c;
benefit(a,a):d,x;
endsets
min=@sum(benefit(i,j):d(i,j)*x(i,j)*c(i));
@for(benefit:@bin(x));
@for(a(i):

```

```

        @sum(a(j):x(i,j))=1);
@for(a(j):@sum(a(i):(c(i)*x(i,j)))<6);
@for(a(j):@sum(a(i):x(i,j))=@if(x(j,j)#eq#0,0,@sum(a(i):x(i,j))));
@for(type(i):x(i,i)=1);
@sum(a(i):
    @sum(a(j):
        @if(x(i,j)*d(i,j)-30#gt#0,1,0)))/72<0.1;
@sum(a(i):x(i,i))<28;
data:
d=@ole('G:/2011B/2a/shortlength1.xls',data1);
c=@ole('G:/2011B/2a/shortlength1.xls',data2);
enddata
end

```

data1为每个区路口的两两节点的最短距离，
data2为每个区路口节点的发案率。

附录7（用lingo设计全市围捕计划）：

```

model:
sets:
department/1..80/;
type/1..19/;
benefit(department,type):d,x;
endsets
min=@max(benefit(i,j):(d(i,j)*x(i,j))/10);
@for(benefit:@bin(x));
@for(department(i):
    @sum(type(j):x(i,j))<1);
@for(type(j):
    @sum(department(i):x(i,j))>1);
@for(department(i):
    @max(type(j):d(i,j)*x(i,j))<20);
data:
d=@ole('G:/z2.xls',data1);
enddata
end

```