**2017南京理工大学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。

我们授权全国大学生数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

我们参赛选择的题号是（从A/B中选择一项填写）： B

我们的参赛报名号为（报名编号）： 120171163

所属学院（请填写完整的全名）： 电子工程与光电技术学院

参赛队员 (打印并签名) ：1. 田浩

2. 朱力维

3. 林翰轩

日期：2017年5月17日

评阅编号（由组委会评阅前进行编号）：

基于节约矩阵法的南京旅游路线设计模型

摘要

随着21世纪初我国经济飞速增长，人民生活水平的不断提高以及收入不断增加，旅游逐渐成为人们在闲暇之时所选择的一种放松方式。随之而来的问题便是在确定目的地后该怎样规划合理的旅游路线。近年来，南京城市面貌焕然一新，多条地铁线开通，公交线路也进行了优化，公共交通换乘优惠实施，公共自行车租赁成为一道亮丽风景。本文中针对南京的一些景点在达到上述交通条件的情况下，对给出的四十个景点针对不同的问题进行了分析。

针对问题一，本文通过运用**节约矩阵法**、**贪心算法**和**旋转扫描法**进行景点路线优化。优化目标为确定在景点开放时间内的路径规划设计，得出到达四十个景点所需里程最小的最优解。模型的原始数据方面，我们合理地将目标地点进行简化，将景点的位置进行**点化处理**，建立了基于经纬的平面坐标。将景点间的路线进行**网状结构处理**，把各个密集景点区域**集成化处理**（把在密集区的相邻景点群集成为一个点，然后再通过最优算法处理内部的最短路程），把起点到各景点的路线按照地图的线路假设成线，从而把问题转化为寻找点与线组成的网络图中各点与各线的最佳路径问题。  
 针对问题二，在原始的数据收集上，利用百度地图，选取了在当前路径情况下的最优的交通工具，由此得到了每两个目标地点间路途所用的**最短时间**。然后在模型的建立和准备方面，本文将路途中所需要的时间作为原始数据，景点的开放时间作为模型计算的约束条件，最后运用**TSP问题的解法**在约束条件的限制下得到目标函数，将原始数据通过**节约运算法**这一最优算法，并运用**Matlab**软件得出目标函数的较优解。最后在综合考虑其他影响因子对最优解的影响，最终得到合理的最优解，及最优路径规划方案。  
 针对问题三，在加入景点逗留时间后，求游览的最短里程。即加入另一约束条件（各景点的**逗留时间因子**），并考虑约束条件之间的制约关系（到达每个目标地点的时间和逗留时间需在景点开放时间这个**硬时间窗**的限制之内）。建立的优化模型中，我们合理详细的加入了这些约束条件以及各个影响因子之间的制约关系，在问题一的基础上很好的解决了到达全部目标地点**里程数最短**这一问题。  
在加入逗留时间后，求最短的游览时间。即在加入逗留时间因子的约束条件下，重新考虑各影响因子的制约关系以及对最优解的总体影响。本文建立了一个以**时间最短**为目标，各影响因子共同制约条件下的最优路径规划模型，很好的解决了在考虑逗留时间情况下的最短游览时间的路径规划设计。  
 此外，本文对于模型进行了**合理性分析**，从原始数据搜集，节约算法的优化与改良，各影响因子变动，新的影响因子的出现对最终结果的合理性与可靠性的影响进行了全方位的分析。同时阐述了该模型在现实领域中的应用，如解决物流配送问题，物资运输网络系统的建立问题，仓储中心的选址问题，使该模型具备相当完善的推广价值。

关键词：节约矩阵法 贪心算法 旋转扫描法 TSP问题解法

**目录**

摘要...........................................................1

一、问题重述...................................................3

1.1 问题背景..................................................3

1.2 需要解决的问题............................................3

二、问题分析...................................................3

2.1问题的重要性分析............................................3

2.2有关方面在这个问题上做过的研究..............................4

2.3问题的思路分析..............................................4

三、模型假设...................................................4

四、符号说明...................................................5

4.1符号说明....................................................5

五、模型的建立及求解...........................................5

5.1 模型一....................................................6

5.1.1 模型一的准备............................................6

5.1.2 模型一的建立以及求解....................................11

5.2 模型二.....................................................12

5.2.1 模型二的准备.............................................12

5.2.2 模型二的建立与求解.......................................13

5.3 模型三.....................................................17

5.3.1 模型三的准备.............................................17

5.3.2 模型三的建立与求解.......................................17

5.4 模型四.....................................................22

5.4.1 模型四的准备.............................................22

5.4.2 模型四的建立与求解.......................................22

六、模型的分析.................................................26

6.1假设的合理性分析............................................26

6.2灵敏度（稳定性、可靠性）分析（给出模型的适用范围）..........26

6.3模型的合理性分析............................................26

七、模型的评价与优化...........................................26

7.1模型的优缺点分析............................................26

7.1.1模型的优点................................................26

7.1.2模型的缺点................................................26

7.2模型的优化..................................................27

八、模型应用领域及推广.........................................27

九、参考文献...................................................27

附录...........................................................28

**一、问题重述**

## 1.1 问题背景

近年来，南京城市面貌焕然一新，多条地铁线开通，公交线路也进行了优化，公共交通换乘优惠实施，公共自行车租赁成为一道亮丽风景。为饱览南京公园景点的美丽景色，一旅游爱好者打算使用公共交通畅游南京公园景点。最近，他已办理好南京市公共自行车卡，并花费230元购买了2017年南京游园年卡，使用该卡他可不限次游览40家景点（见附表）。假定景点的开放时间统一约定为8:00至18:00**。**建立相关数学模型并为该旅游爱好者设计详细的行程表（涵盖起始地点、具体时间点、使用交通线路名称、站牌等信息），回答下面三个问题。

## 1.2 需要解决的问题

1. 如果只考虑景点开放时间。游客每天早晨从南京理工大学出发，晚上回南京理工大学休息，通过乘坐地铁、公交车等公共交通，使用公共自行车、共享单车或步行到达全部景点（景点逗留时间、步行路程不计），至少需要经过多少公里（含骑行加步行、不含早晚地铁）。请建立相关数学模型，设计游览行程表并给出最少里程。

2. 如果游客每天7:00从南京理工大学出发，回到学校时间自行确定，使用地铁、公交车等公共交通、公共自行车、共享单车或步行到达全部景点（景点逗留时间不计），至少需要经过多少天？

3. 假设每个景点游览逗留时间如附表，重新考虑问题1与问题2.

**注：** 相关地点、路线、拥堵时间等数据统一使用百度地图，但公交和地铁线路等以当前所知南京已开通或已优化的为准。

**二、问题分析**

## 2.1问题的重要性分析

随着21世纪初我国经济飞速增长，人民生活水平的不断提高以及收入不断增加，旅游逐渐成为人们在闲暇之时所选择的一种放松方式。随之而来的问题便是在确定目的地后该怎样规划合理的旅游路线。

如果在开始旅行前没有认真地考虑、合理的规划路线以及乘坐哪种交通工具，那么就会产生一些不必要的损失：走过的路程可能会更多；花费的时间可能会更多；消耗的资金可能会更多；影响观赏景点时的心情等等。

对于这些可能发生的问题，合理的规划是很重要的，但是怎样规划使路程最短，或者时间最少，这些问题都需要合理的解决。

## 

## 2.2有关方面在这个问题上做过的研究

## 关于旅游景点线路的优化，早在20世纪60年代就有美国的科学家研究过，当时美国密执安大学的一位教授在对自然和人工自适应系统的研究过程中，在对旅游线路优化问题上，模拟生物在自然环境中的遗传和进化过程形成了一种自适应全局优化概率搜索算法即遗传算法，在当时以至于延续到今日都受到了很大程度的认可。后续的关于这方面的问题还有模拟退火法、蚁群算法以及montocarlo算法等等。

## 2.3问题的思路分析

关于南京公园景点旅游线路设计的问题，本文进行了以下分析：

1.针对问题一，在不考虑路途中的时间以及逗留时间的情况下设计到达40个目标景点的路径使经过的路程最短。在景点开放时间以及出发时间的限制条件下，确定每天在景点开放时间内，去哪几个景点，走什么路线，使得游完四十个景点后所需里程最小。  
 2.针对问题二，同问题一在不考虑逗留时间的情况下设计到达40个目标景点的路径同时在景点开放时间以及出发时间的限制条件下，确定每天在景点开放时间内，去哪几个景点，走什么路线，使得游完四十个景点后所需路途时间最短。  
 3.问题三在考虑路途中所需要的时间，逗留时间，景点开放时间的限制条件下重新考虑问题一二，即给问题一二加入了限制条件（硬时间窗），即在问题一二的条件下引入了多组必须满足的限制条件，在满足所有限制条件的路径时间规划的前提下设计游览方案。  
 4.针对两个目标地点间的距离问题，我们利用百度地图中提供的权威数据结合经纬度将每个景点化为以孝陵卫地铁站（即出发点南京理工大学）为原点，南北方向为x轴，东西方向为y轴的坐标。将目标抽象化，同时查询了在实际情况下两个目标地点的距离位置关系，综合二者考虑路径距离问题。  
 5.针对两个目标地点间路途中所需要的时间问题，我们考虑到了不同交通工具的平均行驶速度，各种公共交通的等待时间，换乘时间等影响因素。同时在不同的路径选用最适合的公共交通工具以满足当前题意下的最优数据，提高具体数据的可信度。

**三、模型假设**

1、假设题目所给的数据真实可靠；

2、假设在交通允许的情况下在骑行时不计寻找共享单车的时间；

3、假设在路途中游客身体状况良好，步行或骑行时能保持平均速度；

4、假设在旅行过程中没有遇到其他特殊情况如自然灾害，交通事故等。

5、假设以到达景点售票处为到达该景点的标志。

6、假设到达城墙的起点为到达该地点。

**四、符号说明**

**4.1 符号说明**

|  |  |
| --- | --- |
| **符号** | **解释说明** |
| （a，b） | a，b两点间的直线距离 |
| （a，b） | 表示由于将两个回路合并成一个回路而节约的距离 |
| N | 天数 |
| Ai | 景点，i=1时代表孝陵卫地铁站，i=2代表明孝陵景区，i=1...41（为了第二三问计算方便所做出的编号上的改变） |
|  | 景点i到景点j所需要的时间，单位为分钟。 |
|  | 第n天景点i到景点j的回路上， =1代表在， =0代表没有 |
|  | 每天早上出门到达第一个景点所需要的时间 |
|  | 景区逗留时间，，其他的为数据（i-1）对应附录1中的标号。 |
|  | 表示景点i到景点j的距离 |

**五、模型的建立及求解**

**问题一：**

针对问题一，需要在景点开放的时间内，以最少的行程游完四十个景点，我们通过百度地图找出四十个地方的经纬度（见附录2）和每个景点到原点的直线距离（见附录3），并将所有景点转化为以孝陵卫地铁站为原点的（x，y）坐标带入模型中计算，得出最佳路线后，在确保当天去到最后一个景点在18:00之前的前提下，利用百度地图，将景点与景点之间的路线根据实际路况确定下来，并得出到达每个景点的时间，两点间的距离，以及在距离最短的情况下最合适的到达方式。

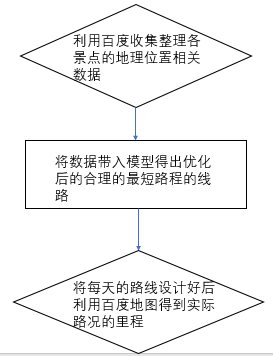


图1 问题一解决方法流程图

**5.1 旅游景点路线优化模型之最少里程（模型一）**

**5.1.1 模型一的准备**

（1）在计算时，我们通过查阅权威资料得出：南京地铁平均时速：60km/h；

南京公交车平均时速：30km/h；南京自行车平均时速：15km/h。

（2）利用百度标出的所有景点布局图，及局部密集景点的放大图。



图2 旅游景点总示意图

注：标号对照附录附表1。



图3 景点密集区示意图

注：图中两条黑线为两段城墙，即12、14。

（3）对于不在密集区的几个相邻较近的点，我们先将其圈起来作为一个点，然后计算圈内的最短距离加到总行程里。以下几幅图为根据总图的景点分布所得到的可以将几个相距特别近的几个点作为一个点，在代入模型计算时按照相距原点最近的由经纬度转化后的景点坐标计算。以下为示例，其余参见附录五（相邻较近的点的图片）



图4 景点33,34,35局部图

**5.1.2 模型一的建立以及求解**

当我们对一些不在密集区且相对来说相离特别近的点进行圈点后，我们可以建立以下**模型一**：



（a，b），a，b两点间的直线距离；（a，b），表示由于将两个回路合并成一个回路而节约的距离；

设南京理工大学附近的孝陵卫地铁站为起点，即为O点，位置坐标为（），各个旅游景点（经过合并后）为1、2、……、n，的位置坐标分别为（）、（）、……、（）。节约矩阵分析法的主要求解步骤包括：

1. **确定距离方阵**。确认距离方阵是要求出起点与各景点之间的距离，在坐标系中两点之间的距离公式为：

Dist（a，b）= （1）

式中a，b是0—n之间的任意数。

1. **确定节约方阵**。节约方阵是指将两个景点放在一条旅游路线上时节约的累积，本文按照距离建立节约方阵。用S（A,B）表示由于将o-a-o、o-b-o两个回路合并成o-a-b-o一个回路而节约的距离，由表1可知节约方阵如下：

S（a，b）=Dist（o，a）+Dist（o，b）-Dist（a，b） （2）

表1 节约方阵

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 景点1 | 景点2 | … | 景点m | … | 景点n |
| 景点1 | S(1，1) |  |  |  |  |  |
| 景点2 | S(2，1) | S(1，1) |  |  |  |  |
| … | … | … | … |  |  |  |
| 景点m | S(m，1) | S(m，2) | … | S(m，m) |  |  |
| … | … | … | … | … | … |  |
| 景点n | S(n，1) | S(n，2) | … | S(n，m) | … | S(n，n) |

表1 节约方阵

1. **确定****每条旅游路线经过的景点**。确定每条旅游路线经过的景点时，目标是在满足每天到达总行程的最后一个景点时还没有到18:00（景点关闭时间）的前提下使总的节约距离最大。方法是首先为各个景点确定单独的旅游回路，任意选择一个景点为起点，按照节约距离越大优先权越高的原则优先合并各景点间的旅游路线直至累积时间接近18:00。接着第二天开始新的路线直至所有景点都经过。
2. **确定每条旅游路线**。利用**贪心算法**可以简单有效的求解问题的最优解或近似最优解。贪心算法步骤如下：
3. 从点1出发计算点1与余下的n-1个点的距离，最短距离为d1，相对应的点记为2’；
4. 计算点2’与余下的n-2个点的距离，最短距离为d2，相对应的点记作3’，……；
5. 计算点j’与余下n-j个点间的距离，最小距离为dj，相对应的点记作（j+1）’，……；
6. 最后计算点（n-1）’与余下的点n’间的距离的d（n-1），总路程：

 （3）

式中为点n’到点1的距离

此模型求得的整体解不一定最优，但可以后期通过画图对结果进行直观的优化；

1. **优化初始旅游路线。**综合节约矩阵法和贪心算法得到旅游路线后，采用**旋转扫描法**，对于明显不合理的景点进一步优化，从而得到使运输行程变短的旅游路线。

将整合的数据（见附录二三）带入模型后得出的**路线**：

表2 第一天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路程/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—2 | 3.0 | 8:00 | 公交+步行 |
| 2—3 | 1.4 | 8:06 | 骑行 |
| 3—27 | 1.8 | 8:14 | 骑行 |
| 27—1 | 1.5 | 8:20 | 骑行 |
| 1—5 | 1.7 | 8:27、 | 骑行 |
| 5—6 | 3.3 | 8:41 | 骑行 |
| 6—4 | 1.8 | 8:49 | 骑行 |
| 4—14 | 4.0 | 9:26 | 公交+步行 |
| 14—8 | 3.3 | 10:07 | 公交+步行 |
| 8—9 | 6.7 | 10:46 | 公交+步行 |
| 9—21 | 3.9 | 11:02 | 骑行 |
| 21—40 | 4.5 | 11:21 | 骑行 |
| 40—24 | 54.2 | 13:32 | 地铁+公交+步行 |
| 24—17 | 52.3 | 15:44 | 地铁+公交+步行 |
| 17—18 | 1.6 | 16:06 | 步行 |
| 18—37 | 4.7 | 16:25 | 骑行 |
| 37—11 | 19.4 | **17:43** | 骑行 |
| 11—孝陵卫地铁站 | 14.5 | 19:04 | 公交+步行 |
| 总公里数 | 183.6 |  |  |

上表为第一天的路线，已完成17个景点的游览，下表为第二天的行程

表3 第二天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路线/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—26 | 5.6 | 8:00 | 地铁+步行 |
| 26—36 | 0.51 | 8:03 | 骑行 |
| 36—10 | 0.29 | 8:05 | 骑行 |
| 10—28 | 0.66 | 8:08 | 骑行 |
| 28—12 | 3.3 | 8:22 | 骑行 |
| 12—13 | 2.1 | 8:50 | 步行 |
| 13—16 | 0.92 | 9:03 | 步行 |
| 16—39 | 1.7 | 9:10 | 骑行 |
| 39—15 | 2.2 | 9:19 | 骑行 |
| 15—7 | 1.8 | 9:27 | 骑行 |
| 7—35 | 2.1 | 9:50 | 公交+步行 |
| 35—33 | 0.16 | 9:53 | 步行 |
| 33—34 | 0.43 | 9:59 | 步行 |
| 34—32 | 1.5 | 10:06 | 骑行 |
| 32—38 | 32.1 | 12:14 | 公交+骑行 |
| 38—23 | 22.1 | 13:45 | 骑行 |
| 23—30 | 89.1 | **17:57** | 公交+地铁+步行 |
| 30—孝陵卫地铁站 | 50.9 | 20:19 | 公交+地铁+步行 |
| 总计 | 217.47 |  |  |

上表为第二天的路线，已完成34个景点的游览，下表为第三天的行程

表4 第三天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路线/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—19 | 14.6 | 8:00 | 公交+步行 |
| 19—31 | 6.0 | 8:35 | 公交+步行 |
| 31—20 | 1.2 | 8:55 | 公交+步行 |
| 20—29 | 10.4 | 10:20 | 公交+步行 |
| 29—22 | 50.7 | 13:43 | 骑行 |
| 22—25 | 16.2 | **15:55** | 公交+步行 |
| 25—孝陵卫地铁站 | 71.1 | 20:08 | 地铁+公交+步行 |
| 总计 | 170.2 |  |  |

上表为第三天的路线，已完成40个景点的游览。

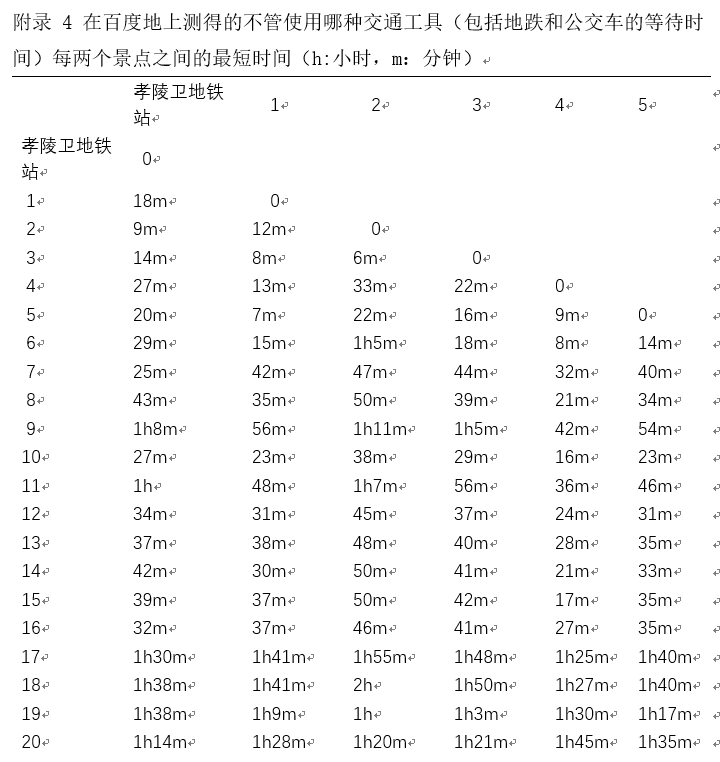
三天里程总计：183.6+217.47+170.2=571.27km

**问题二：**

**5.2 旅游景点路线优化模型之最少时间（模型二）**

**5.2.1 模型二的准备**

针对问题二，我们通过百度地图查询到了四十一个点（包括四十个景点和孝陵卫地铁站）间每两个点之间的行程（使用题目中所给的交通工具，如果有的时间段无公共交通工具，我们才会考虑乘坐出租车）所需的最短时间（见附录4），其中包括了**地铁公交等待时间**，而且在骑自行车和使用其他交通工具时，如果都用时很长但时间相差不大，优先考虑使用除自行车外的其他交通工具，以达到达到更合理的更人性化的要求。



附录4部分数据图（全部数据见附录四）

**5.2.2模型二的建立以及求解**

目标函数为

Min n

约束条件：



N为天数，Ai为景点，i=1时代表南京理工大学，i=2代表明孝陵景区，i=1...41；为景点i到景点j所需要的时间，单位为分钟；为第n天景点i到景点j的回路上，=1代表在，=0代表没有；为第n天从景点i出发后到达目的地等待或花费的时间。

将附录4（景点i到景点j最快的交通方式所用时间）的数据通过以上算法经过Matlab软件处理得出min n=3，路线为下表（此路线的标号为题目所给的标号一致，即为算法里的标号减去1）

表5 早上7:00出发第一天的路线表

|  |  |
| --- | --- |
| 路线 | 到达时间 |
| 孝陵卫地铁站—30 | 9:13 |
| 30—25 | 11:08 |
| 25—22 | 12:11 |
| 22—29 | 15:34 |
| 29—20 | 16:10 |
| 20—31 | 16:14 |
| 31—19 | 16:45 |
| 19—2 | **17:45** |
| 2—孝陵卫地铁站 | 17:54 |
| 总计 | 10小时45分钟 |

上表为第一天的路线，已完成8个景点的游览，下表为第二天的行程

表6 早上7:00出发第二天的路线表

|  |  |
| --- | --- |
| 路线 | 到达时间 |
| 孝陵卫地铁站—23 | 10:05 |
| 23—38 | 11:34 |
| 38—37 | 12:50 |
| 37—18 | 13:09 |
| 18—17 | 13:19 |
| 17—11 | 14:18 |
| 11—32 | 14:39 |
| 32—34 | 14:44 |
| 34—33 | 14:46 |
| 33—35 | 14:47 |
| 35—7 | 14:59 |
| 7—15 | 15:06 |
| 15—28 | 15:17 |
| 28—10 | 15:20 |
| 10—36 | 15:21 |
| 36—26 | 15:23 |
| 26—16 | 15:51 |
| 16—13 | 16:08 |
| 13—39 | 16:14 |
| 39—12 | **16:18** |
| 12—孝陵卫地铁站 | 16:52 |
| 总计 | 9小时18分钟 |

上表为第二天的路线，已完成28个景点的游览，下表为第三天的行程

表七 早上7:00出发第三天的路线表

|  |  |
| --- | --- |
| 路线 | 到达时间 |
| 孝陵卫地铁站—3 | 7:14（等到8:00） |
| 3—27 | 8:08 |
| 27—1 | 8:15 |
| 1—5 | 8:23 |
| 5—6 | 8:37 |
| 6—4 | 8:45 |
| 4—14 | 9:06 |
| 14—8 | 9:16 |
| 8—21 | 9:50 |
| 21—9 | 9:55 |
| 9—40 | 10:08 |
| 40—24 | 12:19 |
| 24—孝陵卫地铁站 | 14h35m |
| 总计 | 5小时19分钟 |

上表为第三天的路线，已完成40个景点的游览。

经过计算得出遍历所有景点后最少时间为（不包含从当天最后一个景点返回孝陵卫地铁站所花费的时间）：10小时45分+9小时18分钟+5小时19分钟=**25小时22分**，最少要分**三天**将所有景点游玩一遍。

**问题三之重述问题一：**

**5.3旅游景点路线优化模型之最少路程（模型三）**

**5.3.1 模型三的准备**

根据问题三第一小问要求，我们引入了**景区逗留时间**，，其他的为附录1当中的数据（i-1）对应附录1中的标号。由于是求最少的公里数，那么我们可以先排列出每个景点i到景点j的哪种交通方式公里数最少，将其用表示。

**5.3.2模型三的建立以及求解**

目标函数为



约束条件：

，

Ai为景点，i=1时代表南京理工大学，i=2代表明孝陵景区，i=1...41；为景点i到景点j所需要的时间，单位为分钟；为第n天景点i到景点j的回路上，=1代表在，=0代表没有；为第n天从景点i出发后到达目的地等待或花费的时间；代表景区逗留时间，，其他的为附录当中的数据；表示景点i到景点j的距离。

我们通过将排列出每个景点i到景点j的哪种交通方式公里数最少的数据带入上面的算法经过使用Matlab软件得出下方的最佳路线。

表8 第一天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路程/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—2 | 3.0 | 8:00 | 公交+步行 |
| 2—3 | 1.4 | 9:06 | 骑行 |
| 3—27 | 1.8 | 9:44 | 骑行 |
| 27—1 | 1.5 | 10:50 | 骑行 |
| 1—5 | 1.7 | 11:57 | 骑行 |
| 5—6 | 3.3 | 13:11 | 骑行 |
| 6—4 | 1.8 | 13:49 | 骑行 |
| 4—14 | 4 | 14:57 | 公交+骑行 |
| 14—8 | 3.3 | 15:58 | 公交+骑行 |
| 8—孝陵卫地铁站 | 11.6 | 18:55 | 骑行 |
| 总计 | 69.4 |  |  |

上表为第一天的路线，已完成9个景点的游览，下表为第二天的行程

表9 第二天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路程/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—26 | 5.9 | 8:00 | 骑行 |
| 26—36 | 0.51 | 9:33 | 骑行 |
| 36—10 | 0.29 | 10:35 | 骑行 |
| 10—28 | 0.66 | 12:08 | 骑行 |
| 28—15 | 2.6 | 13:19 | 骑行 |
| 15—7 | 2.6 | 16:00 | 骑行 |
| 7—孝陵卫地铁站 | 10 | 18;11 | 骑行 |
| 总计 | 22.56 |  |  |

上表为第二天的路线，已完成15个景点的游览，下表为第三天的行程

表10 第三天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路程/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—12 | 7.4 | 8:00 | 骑行 |
| 12—13 | 2.1 | 8:39 | 骑行 |
| 13—16 | 1.1 | 9:44 | 骑行 |
| 16—39 | 1.7 | 11:21 | 骑行 |
| 39—35 | 4.4 | 12:39 | 骑行 |
| 35—33 | 0.16 | 13:10 | 骑行 |
| 33—34 | 0.43 | 13:46 | 步行 |
| 34—32 | 1.6 | 14:23 | 骑行 |
| 32—11 | 5.1 | 15:44 | 骑行 |
| 11—21 | 7.9 | 16:46 | 骑行 |
| 21—孝陵卫地铁站 | 17.7 | 18:57 | 骑行 |
| 总计 | 49.59 |  |  |

上表为第三天的路线，已完成25个景点的游览，下表为第四天的行程

表11 第四天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路程/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—9 | 16.9 | 8:00 | 骑行 |
| 9—40 | 4.1 | 9:17 | 骑行 |
| 40—17 | 14.5 | 11:04 | 公交+地铁+步行 |
| 17—18 | 1.6 | 13:26 | 步行 |
| 18—37 | 8.9 | 15:02 | 骑行 |
| 37—孝陵卫地铁站 | 31.4 | 19:28 | 公交+步行 |
| 总计 | 77.7 |  |  |

上表为第四天的路线，已完成30个景点的游览，下表为第五天的行程

表12 第五天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路程/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—38 | 38.0 | 8:30 | 公交+步行 |
| 38—23 | 26.2 | 11:58 | 公交+步行 |
| 23—30 | 89.1 | 16:59 | 公交+地铁+骑行 |
| 30——地铁站 | 50.9 | 20.11 | 公交+地铁+步行 |
| 总计 | 204.2 |  |  |

上表为第五天的路线，已完成33个景点的游览，下表为第六天的行程

表13 第六天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路程/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—29 | 26.3 | 8:00 | 骑行 |
| 29—22 | 50.7 | 13:50 | 骑行+步行 |
| 22—25 | 15.6 | 15:53 | 骑行 |
| 25—卫地铁站 | 71.1 | 21:36 | 公交+骑行+步行 |
| 总计 | 163.7 |  |  |

上表为第六天的路线，已完成36个景点的游览，下表为第七天的行程

表14 第七天路线表

|  |  |  |  |
| --- | --- | --- | --- |
| 路线 | 路程/km | 时间/h | 到达方式 |
| 孝陵卫地铁站—19 | 14.6 | 8:00 | 公交+步行 |
| 19—31 | 7.6 | 10:01 | 骑行 |
| 31—20 | 0.67 | 11:04 | 骑行 |
| 20—24 | 84.6 | 15:17 | 公交+地铁+步行 |
| 24—孝陵卫地铁站 | 68.1 | 19:03 | 地铁+步行 |
| 总计 | 175.57 |  |  |

上表为第七天的路线，已完成40个景点的游览。

最短里程为:69.4+22.56+49.59+77.7+204.2+163.7+175.57=762.72km

**问题三之重述问题二：**

**5.4旅游景点路线优化模型之最少时间（模型四）**

**5.4.1模型四的准备**

在模型二的基础上，我们引入了**景区逗留时间**，，其他的为附录1当中的数据（i-1）对应附录1中的标号。

**5.4.2模型四的建立以及求解**

目标函数为

Min n

约束条件：



Ai为景点，i=1时代表南京理工大学，i=2代表明孝陵景区，i=1...41；为景点i到景点j所需要的时间，单位为分钟；为第n天景点i到景点j的回路上，=1代表在，=0代表没有；为第n天从景点i出发后到达目的地等待或花费的时间；代表景区逗留时间，，其他的为附录当中的数据

将附录4的数据（景点i到景点j最快的交通方式所用时间）带入上述算法，通过使用Matlab软件得到n=7。并且给出每天的路线表。

表15 第一天路线表

|  |  |  |
| --- | --- | --- |
| 第一天路线 | 到达时间 | 离开时间 |
| 孝陵卫地铁站-39 | 7:43 | 8:43 |
| 39-12 | 9:00 | 9:30 |
| 12-13 | 9:45 | 10:45 |
| 13-16 | 11:00 | 13:15 |
| 16-28 | 13:30 | 14:30 |
| 28-10 | 14:33 | 16:00 |
| 10-36 | 16:05 | 17:05 |
| 36-38 | 17:12 | 17:43 |

上表为第一天的路线，已完成个8景点的游览，下表为第二天的行程。

表16第二天路线表

|  |  |  |
| --- | --- | --- |
| 第二天路线 | 到达时间 | 离开时间 |
| 孝陵卫地铁站-9 | 8:00 | 9:00 |
| 9-21 | 9:05 | 10:05 |
| 21-8 | 10:40 | 12:40 |
| 8-14 | 12:50 | 13:20 |

上表为第二天的路线，已完成12个景点的游览，下表为第三天的行程

表17第三天路线表

|  |  |  |
| --- | --- | --- |
| 第三天路线 | 到达时间 | 离开时间 |
| 孝陵卫地铁站-31 | 8:15 | 9:15 |
| 31-20 | 9:20 | 9:50 |
| 20-19 | 10:25 | 12:00 |
| 19-2 | 13:00 | 14:00 |
| 2-3 | 14:05 | 14:35 |
| 3-27 | 14:50 | 15:50 |
| 27-1 | 15:55 | 16:55 |
| 1-29 | 17:00 | 18:00 |

上表为第三天的路线，已完成个20景点的游览，下表为第四天的行程

表18 第四天路线表

|  |  |  |
| --- | --- | --- |
| 第四天路线 | 到达时间 | 离开时间 |
| 孝陵卫地铁站-40 | 8:12 | 8:42 |
| 40-11 | 9:00 | 9:30 |
| 11-7 | 10:05 | 11:35 |
| 7-15 | 11:42 | 13:30 |
| 15-32 | 13:44 | 14:44 |
| 32-36 | 15:05 | 16:35 |
| 26-40 | 17:25 | 17:55 |

上表为第四天的路线，已完成27个景点的游览，下表为第五天的行程

表19 第五天路线表

|  |  |  |
| --- | --- | --- |
| 第五天路线 | 到达时间 | 离开时间 |
| 孝陵卫地铁站-17 | 8:30 | 10:30 |
| 17-18 | 10:40 | 11:40 |
| 18-37 | 12:40 | 14:40 |
| 37-23 | 16:50 | 17:50 |

上表为第五天的路线，已完成31个景点的游览，下表为第六天的行程

表20 第六天路线表

|  |  |  |
| --- | --- | --- |
| 第六天路线 | 到达时间 | 离开时间 |
| 孝陵卫地铁站-30 | 9:15 | 10:15 |
| 30-25 | 12:30 | 14:00 |
| 25-32 | 16:30 | 17:30 |

上表为第六天的路线，已完成34个景点的游览，下表为第七天的行程

表21 第七天路线表

|  |  |  |
| --- | --- | --- |
| 第七天路线 | 到达时间 | 离开时间 |
| 孝陵卫地铁站-4 | 8:00 | 8:30 |
| 4-5 | 8:40 | 9:40 |
| 5-6 | 9:55 | 10:25 |
| 6-33 | 10:5 | 11:25 |
| 33-34 | 11:35 | 12：05 |
| 34-35 | 12：07 | 12：37 |

上表为第七天的路线，已完成40个景点的游览。

所以游客至少需要7天。

**六、模型的分析**

## 6.1假设的合理性分析

1.假设一使每两个地点间的距离与旅行时间真实可靠。

2.假设二排除了骑行等待时间对总时间的影响。

3.假设三排除了路途中速度不一致的情况，确保了路途时间的可靠性。

4.假设四排除了一些特殊情况对路径规划的影响。

5.假设五排除了范围较大的景点无法确定到达位置的情况。

6.假设六排除了城墙长度对路径规划的影响。

## 6.2灵敏度（稳定性、可靠性）分析（给出模型的适用范围）

本文以节约矩阵法并且利用TSP问题相似的解法构建了针对由一个起点出发以最短的时间/最短的路程到达各个目标地点的最优算法的模型。后续改进的模型在原模型的基础上夹加上了硬时间窗限制，构建了一个受限制条件制约，考虑到各种其他因素影响下的最优算法的模型。此模型经过合理改进之后可以适用于各种旅游路径规划，现代物流配送方案设计，最优方案设计，最优时间方案设计的各种问题，具有相当广泛的适用范围。

## 6.3模型的合理性分析

## 本文构建的模型充分考虑到了各种原始条件的相互制约关系，原始数据的真实性与合理性，即原始条件以及数据的合理性。同时利用节约矩阵法、贪心算法和旋转扫描法这些成熟的最优算法，并且利用TSP问题相似的解法进行在硬时间窗限制下的最优路径规划设计，即模型设计中算法过程以及求解过程的合理，可信。最终的结果与其他影响因子进行综合考量改进，即结果的合理性。

**七、模型的评价与优化**

## 7.1模型的优缺点分析

**7.1.1模型的优点**

1.通过一些合理适当的假设，去除了一些特殊外部因素的干扰，提升了计算结果的可靠性。

2.所有的数据均来自于百度地图提供的数据，真实可靠。

3.本文模型考虑了例如地铁换乘，公交等待时间，符合实际情况。

4.模型中两个景点间的路途中选取了最优的交通工具，保证了时间/距离的最优化解。

**7.1.2模型的缺点**

1. 本模型将距离相近的景点集成为同一个景点后规划路线，两段城墙将起始点作为坐标确定点，模型计算模拟路线可能不准确。
2. 在数据收集时，两个景点间的距离以及路途中所需要的时间考虑到不同交通工具间的等待时间，存在一定的不确定性。

## 7.2模型的优化

1.可以考虑到人的精力对步行以及骑行速度的影响，引入影响因子ki。

2.可以考虑到交通高峰期对陆上交通速度的影响，引入影响因子li。

**八、模型应用领域及推广**

1. 本文提出的基于**节约矩阵法、贪心算法**和**旋转扫描法**进行景点路线优化。可以应用于旅行路径的设计，提高路径设计的可行性与合理性。
2. 本文构建的模型可以应用与现代物流行业的配送设计以及配送时间的规划之中，是物流系统优化中的关键一环，是电子商务活动中不可缺少的一部分。问题三中构建的模型可以应用于在硬时间窗的限制下，物流配送的路径以及时间规划问题之中，以提高物流配送的效率，迎合物流配送的顾客需求，减少配送过程中的运输成本，进而在整体上对配送车辆进行优化调度，提高物流的经济效益，实现物流科学化，构建合理稳定的物联网。

**九、参考文献**

[1]南京交通运输局网址http://www.njjt.gov.cn/

[2] 谭国真，高文．时间依赖的网络中最小时间路径算法［Ｊ］．计算机学报，２００２，２５（２）：１６５-１７１

[3] 黄贵玲，高西全，靳松杰，等．基于蚁群算法的最短路径问题的研究和应用［Ｊ］．计算机工程与应用，２００７，４３（１３）：２３３-２３５

[4] 温惠英，徐建闽，邹亮．基于遗传算法的离散时间动态网络最短路径求解［Ｊ］．华南理工大学学报：自然科学版，２００８，３６（２）：１３-２８．

[5] 杨庆芳，梅朵，郑黎黎，等．基于云计算的城市路网最短路径遗传算法求解［Ｊ］．华南理工大学学报：自然科学版，２０１４，４２（３）：１６６-１６９

**附录**

**附表1：2017年南京游园年卡覆盖景点以及各景点游览逗留时间**

|  |  |  |
| --- | --- | --- |
| 序 | 景点名 | 游览逗留时间(单位：小时) |
| 1 | 明孝陵景区 | 1 |
| 2 | 灵谷寺景区 | 1 |
| 3 | 音乐台景区 | 0.5 |
| 4 | 白马石刻公园 | 0.5 |
| 5 | 中山植物园 | 1 |
| 6 | 紫金山天文台 | 0.5 |
| 7 | 莫愁湖公园 | 1.5 |
| 8 | 红山森林动物园 | 2 |
| 9 | 燕子矶公园 | 1 |
| 10 | 总统府 | 1.5 |
| 11 | 阅江楼景区 | 0.5 |
| 12 | 东水关至集庆门段城墙（含中华门瓮城） | 0.5 |
| 13 | 瞻园（太平天国历史博物馆，含夜瞻园） | 1 |
| 14 | 神策门瓮城至太平门段城墙（含明城垣史博物馆） | 0.5 |
| 15 | 市博物馆（朝天宫） | 1.5 |
| 16 | 市民俗博物馆（甘熙宅第） | 1.5 |
| 17 | 珍珠泉旅游度假区（含狮虎园） | 2 |
| 18 | 珍珠泉名石艺术馆 | 1 |
| 19 | 明文化村（阳山碑材） | 1.5 |
| 20 | 汤山古猿人洞 | 0.5 |
| 21 | 长江观音景区 | 1 |
| 22 | 大金山风景区 | 1 |
| 23 | 南京万成生态园 | 1 |
| 24 | 金牛湖风景区 | 1.5 |
| 25 | 天生桥风景区 | 1.5 |
| 26 | 毗卢寺 | 1.5 |
| 27 | 南京美龄宫 | 1 |
| 28 | 江宁织造博物馆 | 1 |
| 29 | 汤山翠谷 | 1 |
| 30 | 南京蔬菜花卉科技园 | 1 |
| 31 | 南京直立人化石遗址博物馆 | 1 |
| 32 | 国防教育馆 | 1 |
| 33 | 李剑晨艺术馆 | 0.5 |
| 34 | 魏紫熙艺术馆 | 0.5 |
| 35 | 龚贤纪念馆 | 0.5 |
| 36 | 六朝博物馆 | 1 |
| 37 | 南京老山国家森林公园 | 2 |
| 38 | 南京雨发生态园 | 1.5 |
| 39 | 愚园 | 1 |
| 40 | 达摩古洞景区 | 0.5 |

附录二：四十个景点的经纬度（°）

|  |  |  |
| --- | --- | --- |
| 1 | 明孝陵景区 | 2.0644154346,118.8461735960 |
| 2 | 灵谷寺景区 | 32.0606242502,118.8745043883 |
| 3 | 音乐台景区 | 32.0607641816,118.8619798573 |
| 4 | 白马石刻公园 | 32.0665917291,118.8224128082 |
| 5 | 中山植物园 | 32.0587884821,118.8407072953 |
| 6 | 紫金山天文台 | 32.0709808640,118.8359849371 |
| 7 | 莫愁湖公园 | 32.0429987347,118.7670319280 |
| 8 | 红山森林动物园 | 32.0982419015,118.8097694058 |
| 9 | 燕子矶公园 | 32.1527854011,118.8238468021 |
| 10 | 总统府 | 32.0495003216,118.8034188196 |
| 11 | 阅江楼景区 | 32.0991719109,118.7541743056 |
| 12 | 东水关至集庆门段城墙（含中华门瓮城） | 32.0296294844,118.8048959736 |
| 13 | 瞻园（太平天国历史博物馆，含夜瞻园） | 32.0267685088,118.7914622371 |
| 14 | 神策门瓮城至太平门段城墙（含明城垣史博物馆） | 32.0945347748,118.7941080578 |
| 15 | 市博物馆（朝天宫） | 32.0422043146,118.7790440148 |
| 16 | 市民俗博物馆（甘熙宅第） | 32.0324336694,118.7885524361 |
| 17 | 珍珠泉旅游度假区（含狮虎园） | 32.1278344298,118.6655313857 |
| 18 | 珍珠泉名石艺术馆 | 32.1260644298,118.6622913857 |
| 19 | 明文化村（阳山碑材） | 32.0679017312,119.0081373354 |
| 20 | 汤山古猿人洞 | 32.0614592375,119.0519829886 |
| 21 | 长江观音景区 | 32.1341098404,118.8022144175 |
| 22 | 大金山风景区 | 31.6986922670,119.1249582960 |
| 23 | 南京万成生态园 | 31.9499585805,118.4323906896 |
| 24 | 金牛湖风景区 | 32.4757684389,118.9749837015 |
| 25 | 天生桥风景区 | 31.6511276829,118.9957960656 |
| 26 | 毗卢寺 | 32.0485503216,118.8096288196 |
| 27 | 南京美龄宫 | 32.0512934012,118.8531278718 |
| 28 | 江宁织造博物馆 | 32.0494154083,118.8008096854 |
| 29 | 汤山翠谷 | 32.0062214767,119.0805706833 |
| 30 | 南京蔬菜花卉科技园 | 31.7256289035,118.7839578173 |
| 31 | 南京直立人化石遗址博物馆 | 32.0617214877,119.0468305114 |
| 32 | 国防教育馆 | 31.6984922670,119.1278182960 |
| 33 | 李剑晨艺术馆 | 32.0554145917,118.7662357522 |
| 34 | 魏紫熙艺术馆 | 32.0569645917,118.7687257522 |
| 35 | 龚贤纪念馆 | 32.0542845917,118.7673157522 |
| 36 | 六朝博物馆 | 32.0487303216,118.8056788196 |
| 37 | 南京老山国家森林公园 | 32.1192175253,118.6276649804 |
| 38 | 南京雨发生态园 | 32.0406971337,118.5428959147 |
| 39 | 愚园 | 32.0225849242,118.7804926861 |
| 40 | 达摩古洞景区 | 32.1364381638,118.7987972995 |

附录三；根据百度地图的测距工具得到的真实数据

|  |  |
| --- | --- |
| 景点 | 距起点的距离/km |
| 孝陵卫地铁站（起点） | 0 |
| 33、34、35 | 9.3 |
| 19、20、31 | 13.9 |
| 17、18、37 | 21.1 |
| 1、2、3、4、5、6、27 | 1.7 |
| 10、26、28、36 | 5.3 |
| 9、21、40 | 12.0 |
| 12、13、16、39 | 5.8 |
| 8、14 | 8.5 |
| 7 | 9.1 |
| 11 | 12.4 |
| 15 | 7.9 |
| 22 | 45.1 |
| 23 | 42.2 |
| 24 | 50.7 |
| 25 | 52.5 |
| 29 | 20.6 |
| 30 | 35.9 |
| 32 | 9.9 |
| 38 | 30.5 |

附录4 在百度地上测得的不管使用哪种交通工具（包括地跌和公交车的等待时间）每两个景点之间的最短时间（h:小时，m：分钟）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 孝陵卫地铁站 | 1 | 2 | 3 | 4 | 5 |
| 孝陵卫地铁站 | 0 |  |  |  |  |  |
| 1 | 18m | 0 |  |  |  |  |
| 2 | 9m | 12m | 0 |  |  |  |
| 3 | 14m | 8m | 6m | 0 |  |  |
| 4 | 27m | 13m | 33m | 22m | 0 |  |
| 5 | 20m | 7m | 22m | 16m | 9m | 0 |
| 6 | 29m | 15m | 1h5m | 18m | 8m | 14m |
| 7 | 25m | 42m | 47m | 44m | 32m | 40m |
| 8 | 43m | 35m | 50m | 39m | 21m | 34m |
| 9 | 1h8m | 56m | 1h11m | 1h5m | 42m | 54m |
| 10 | 27m | 23m | 38m | 29m | 16m | 23m |
| 11 | 1h | 48m | 1h7m | 56m | 36m | 46m |
| 12 | 34m | 31m | 45m | 37m | 24m | 31m |
| 13 | 37m | 38m | 48m | 40m | 28m | 35m |
| 14 | 42m | 30m | 50m | 41m | 21m | 33m |
| 15 | 39m | 37m | 50m | 42m | 17m | 35m |
| 16 | 32m | 37m | 46m | 41m | 27m | 35m |
| 17 | 1h30m | 1h41m | 1h55m | 1h48m | 1h25m | 1h40m |
| 18 | 1h38m | 1h41m | 2h | 1h50m | 1h27m | 1h40m |
| 19 | 1h38m | 1h9m | 1h | 1h3m | 1h30m | 1h17m |
| 20 | 1h14m | 1h28m | 1h20m | 1h21m | 1h45m | 1h35m |
| 21 | 1h13m | 59m | 1h16m | 1h9m | 46m | 59m |
| 22 | 3h42m | 3h56m | 4h | 3h52m | 4h18m | 3h56m |
| 23 | 3h5m | 3h22m | 3h30m | 3h36m | 3h18m | 3h43m |
| 24 | 2h16m | 2h23m | 2h40m | 2h45m | 2h6m | 2h31m |
| 25 | 3h9m | 3h25m | 3h30m | 3h37m | 3h12m | 3h42m |
| 26 | 19m | 21h | 35m | 26m | 25m | 20m |
| 27 | 26m | 7m | 14m | 8m | 17m | 10m |
| 28 | 17m | 25m | 38m | 30m | 16m | 24m |
| 29 | 1h42m | 1h53m | 1h45m | 1h46m | 2h6m | 2h |
| 30 | 2h13m | 2h29m | 2h36m | 2h41m | 2h14m | 2h54m |
| 31 | 1h15m | 1h26m | 1h18m | 1h20m | 1h45m | 36m |
| 32 | 44m | 41m | 52m | 47m | 30m | 40m |
| 33 | 40m | 38m | 52m | 43m | 28m | 36m |
| 34 | 41m | 37m | 59m | 44m | 27m | 36m |
| 35 | 40m | 37m | 52m | 43m | 28m | 36m |
| 36 | 25m | 23m | 36m | 28m | 14m | 22m |
| 37 | 1h41m | 1h58m | 2h7m | 2h12m | 1h55m | 2h13m |
| 38 | 2h18m | 2h20m | 2h41m | 2h29m | 2h24m | 2h36m |
| 39 | 44m | 43m | 55m | 47m | 34m | 41m |
| 40 | 1h12m | 1h | 1h7m | 1h9m | 46m | 59m |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | 6 | 7 | 8 | 9 | 10 |  |
| 6 | 0 |  |  |  |  |  |
| 7 | 38m | 0 |  |  |  |  |
| 8 | 26m | 44m | 0 |  |  |  |
| 9 | 47m | 1h5m | 31m | 0 |  |  |
| 10 | 22m | 20m | 34m | 55m | 0 |  |
| 11 | 42m | 34m | 28m | 42m | 15m |  |
| 12 | 30m | 16m | 42m | 1h3m | 24m |  |
| 13 | 33m | 12m | 45m | 1h6m | 14m |  |
| 14 | 24m | 34m | 10m | 34m | 14m |  |
| 15 | 33m | 7m | 35m | 1h5m | 1h24m |  |
| 16 | 34m | 10m | 43m | 1h6m | 1h27m |  |
| 17 | 1h33m | 1h18m | 1h15m | 1h24m | 1h27m |  |
| 18 | 1h33m | 1h26m | 1h16m | 1h25m | 1h42m |  |
| 19 | 1h21m | 1h32m | 1h45m | 1h37m | 1h24m |  |
| 20 | 1h39m | 1h35m | 1h54m | 1h55m | 1h42m |  |
| 21 | 52m | 1h4m | 34m | 5m | 1h |  |
| 22 | 4h37m | 4h9m | 4h19m | 3h36m | 4h10m |  |
| 23 | 3h41m | 2h47m | 3h17m | 3h43m | 3h30m |  |
| 24 | 2h33m | 2h13m | 2h15m | 2h26m | 2h26m |  |
| 25 | 3h32m | 3h2m | 3h17m | 3h46m | 3h |  |
| 26 | 19m | 22m | 31m | 52m | 4m |  |
| 27 | 20m | 40m | 37m | 59m | 22m |  |
| 28 | 22m | 19m | 34m | 55m | 3m |  |
| 29 | 2h4m | 2h24m | 2h28m | 2h20m | 2h7m |  |
| 30 | 2h36m | 2h8m | 2h18m | 2h47m | 2h5m |  |
| 31 | 1h38m | 1h36m | 2h18m | 1h54m | 1h40m |  |
| 32 | 36m | 14m | 37m | 1h1m | 20m |  |
| 33 | 33m | 11m | 38m | 1h1m | 16m |  |
| 34 | 33m | 11m | 36m | 1h1m | 16m |  |
| 35 | 33m | 12m | 38m | 1h2m | 16m |  |
| 36 | 20m | 21m | 32m | 54m | 1m |  |
| 37 | 2h18m | 2h | 1h49m | 1h59m | 1h38m |  |
| 38 | 2h36m | 1h43m | 2h12m | 2h22m | 2h15m |  |
| 39 | 40m | 12m | 49m | 1h13m | 20m |  |
| 40 | 52m | 58m | 30m | 13m | 51m |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | 11 | 12 | 13 | 14 | 15 |  |
| 11 | 0 |  |  |  |  |  |
| 12 | 1h3m | 0 |  |  |  |  |
| 13 | 1h6m | 7m | 0 |  |  |  |
| 14 | 34m | 35m | 31m | 0 |  |  |
| 15 | 32m | 19m | 10m | 32m | 0 |  |
| 16 | 37m | 12m | 17m | 30m | 6m |  |
| 17 | 59m | 1h30m | 1h28m | 1h10m | 1h25m |  |
| 18 | 59m | 1h35m | 1h33m | 1h13m | 1h25m |  |
| 19 | 1h56m | 1h52m | 1h32m | 1h54m | 1h34m |  |
| 20 | 2h11m | 1h46m | 1h42m | 1h53m | 1h44m |  |
| 21 | 32m | 1h6m | 1h5m | 1h6m | 1h |  |
| 22 | 4h34m | 4h | 3h59m | 4h18m | 4h10m |  |
| 23 | 3h10m | 3h16m | 3h | 3h17m | 3h4m |  |
| 24 | 2h10m | 2h14m | 2h11m | 2h4m | 2h22m |  |
| 25 | 3h37m | 3h16m | 2h54m | 3h13m | 3h4m |  |
| 26 | 36m | 24m | 16m | 27m | 15m |  |
| 27 | 55m | 27m | 34m | 34m | 32m |  |
| 28 | 32m | 14m | 13m | 34m | 11m |  |
| 29 | 2h39m | 2h | 1h44m | 2h30m | 2h17m |  |
| 30 | 2h38m | 2h2m | 1h58m | 2h17m | 2h9m |  |
| 31 | 2h13m | 1h47m | 1h44m | 1h54m | 1h45m |  |
| 32 | 21m | 27m | 21m | 40m | 14m |  |
| 33 | 24m | 23m | 17m | 32m | 10m |  |
| 34 | 23m | 24m | 19m | 24m | 11m |  |
| 35 | 24m | 23m | 18m | 25m | 11m |  |
| 36 | 35m | 12m | 15m | 26m | 13m |  |
| 37 | 34m | 2h | 1h45m | 1h42m | 1h41m |  |
| 38 | 1h56m | 2h26m | 1h53m | 2h5m | 2h |  |
| 39 | 41m | 4m | 6m | 36m | 9m |  |
| 40 | 13m | 1h2m | 58m | 28m | 53m |  |
|  |  |  |  |  |  |  |
|  | 16 | 17 | 18 | 19 | 20 |  |
| 16 | 0 |  |  |  |  |  |
| 17 | 1h28m | 0 |  |  |  |  |
| 18 | 1h38m | 10m | 0 |  |  |  |
| 19 | 1h51m | 2h40m | 2h46m | 0 |  |  |
| 20 | 1h42m | 2h42m | 2h47m | 32m | 0 |  |
| 21 | 1h18m | 1h28m | 1h44m | 1h45m | 2h35m |  |
| 22 | 4h | 5h4m | 5h9m | 4h4m | 5h25m |  |
| 23 | 3h | 2h39m | 3h | 4h7m | 4h16m |  |
| 24 | 2h17m | 2h12m | 2h26m | 3h19m | 3h28m |  |
| 25 | 2h54m | 4h2m | 4h7m | 4h | 4h24m |  |
| 26 | 28m | 1h28m | 1h23m | 1h12m | 1h30m |  |
| 27 | 33m | 1h46m | 1h49m | 1h3m | 1h26m |  |
| 28 | 12m | 1h21m | 1h6m | 1h10m | 1h32m |  |
| 29 | 2h | 3h33m | 4h7m | 57m | 36m |  |
| 30 | 2h | 3h2m | 3h11m | 3h16m | 3h24m |  |
| 31 | 1h43m | 2h49m | 2h48m | 31m | 4m |  |
| 32 | 18m | 1h9m | 1h20m | 1h42m | 1h55m |  |
| 33 | 15m | 1h10m | 1h18m | 1h38m | 1h53m |  |
| 34 | 15m | 1h11m | 1h18m | 1h39h | 1h53m |  |
| 35 | 15m | 1h11m | 1h18m | 1h39m | 1h53m |  |
| 36 | 14m | 1h26m | 1h28m | 1h23m | 1h36m |  |
| 37 | 1h36m | 45m | 19 | 2h44m | 2h54m |  |
| 38 | 2h10m | 1h19m | 1h12m | 3h22m | 3h29m |  |
| 39 | 7m | 1h34m | 1h35m | 1h41m | 1h52m |  |
| 40 | 55m | 1h20m | 1h21m | 1h49m | 2h8m |  |
|  |  |  |  |  |  |  |
|  | 21 | 22 | 23 | 24 | 25 |  |
| 21 | 0 |  |  |  |  |  |
| 22 | 4h56m | 0 |  |  |  |  |
| 23 | 3h26m | 6h30m | 0 |  |  |  |
| 24 | 2h22m | 6h | 5h33m | 0 |  |  |
| 25 | 3h53m | 1h3m | 5h20m | 4h7m | 0 |  |
| 26 | 52m | 4h14m | 3h51m | 2h10m | 4h2m |  |
| 27 | 59m | 5h | 4h2m | 2h29m | 3h54m |  |
| 28 | 50m | 4h | 3h44m | 2h | 3h54m |  |
| 29 | 2h28m | 3h23m | 3h27m | 4h30m | 3h22m |  |
| 30 | 2h57m | 3h51m | 1h50m | 3h48m | 1h55m |  |
| 31 | 2h2m | 6h | 3h46m | 3h25m | 3h41m |  |
| 32 | 48m | 4h35m | 4h3m | 2h33m | 3h58m |  |
| 33 | 50m | 4h30m | 3h59m | 2h32m | 3h54m |  |
| 34 | 52m | 4h30m | 4h | 2h32m | 3h55m |  |
| 35 | 51m | 4h30m | 4h01m | 2h32m | 3h54m |  |
| 36 | 53m | 4h9m | 3h52m | 2h11m | 3h34m |  |
| 37 | 1h49m | 6h5m | 2h7m | 3h | 5h37m |  |
| 38 | 2h12m | 6h28m | 1h29m | 3h34m | 5h5m |  |
| 39 | 1h2m | 4h2m | 3h46m | 2h32m | 3h41m |  |
| 40 | 19m | 4h51m | 3h32m | 2h11m | 4h32m |  |
|  |  |  |  |  |  |  |
|  | 26 | 27 | 28 | 29 | 30 |  |
| 26 | 0 |  |  |  |  |  |
| 27 | 20m | 0 |  |  |  |  |
| 28 | 4m | 23m | 0 |  |  |  |
| 29 | 2h15m | 1h51m | 2h45h | 0 |  |  |
| 30 | 2h10m | 2h29m | 2h | 3h16m | 0 |  |
| 31 | 1h31m | 1h24m | 1h30m | 39m | 3h23 |  |
| 32 | 19m | 40m | 16m | 2h27m | 2h26m |  |
| 33 | 18m | 37m | 15m | 2h25m | 2h22m |  |
| 34 | 18m | 38m | 15m | 2h24m | 2h22m |  |
| 35 | 18m | 37m | 15m | 2h24m | 2h22m |  |
| 36 | 2m | 21m | 3m | 2h8m | 2h5m |  |
| 37 | 1h40m | 1h58m | 1h31m | 4h5m | 3h |  |
| 38 | 2h15m | 2h18m | 2h8m | 4h32m | 3h36m |  |
| 39 | 22m | 41m | 18m | 2h27m | 2h26m |  |
| 40 | 53m | 1h3m | 49m | 2h35m | 2h45m |  |
|  |  |  |  |  |  |  |
|  | 31 | 32 | 33 | 34 | 35 |  |
| 31 | 0 |  |  |  |  |  |
| 32 | 1h55m | 0 |  |  |  |  |
| 33 | 1h53m | 5m | 0 |  |  |  |
| 34 | 1h37m | 5m | 2m | 0 |  |  |
| 35 | 1h51m | 5m | 1m | 1m | 0 |  |
| 36 | 1h37m | 21m | 17m | 17m | 16m |  |
| 37 | 2h55m | 1h42m | 1h42m | 1h46m | 1h40m |  |
| 38 | 3h12m | 2h10m | 2h14m | 2h20m | 1h58m |  |
| 39 | 1h52m | 22m | 19m | 20m | 18m |  |
| 40 | 2h6m | 45m | 49m | 50m | 48m |  |
|  |  |  |  |  |  |  |
|  | 36 | 37 | 38 | 39 | 40 |  |
| 36 | 0 |  |  |  |  |  |
| 37 | 1h39m | 0 |  |  |  |  |
| 38 | 2h17m | 1h16m | 0 |  |  |  |
| 39 | 20m | 1h38m | 2h19m | 0 |  |  |
| 40 | 51m | 1h40m | 2h17m | 1h | 0 |  |

附录5

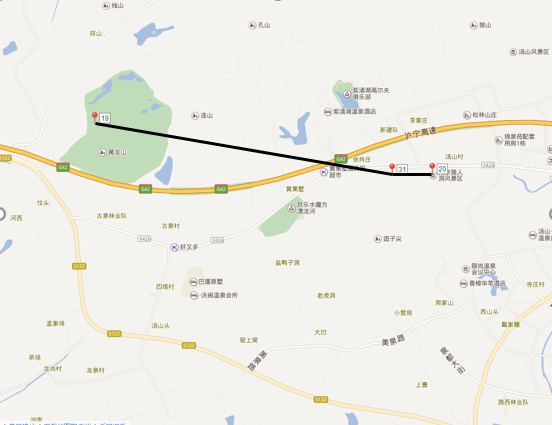


图5 景点19,20,31局部示意图



图6 景点17,18,37局部示意图



图7 景点1,2,3,4,5,6局部示意图



图8 景点10,26,28,36局部示意图



图9 景点9,21,40局部示意图



图10 景点12,13,16,39局部示意图



图11 景点8,14局部示意图

附录6 MATLAB代码

问题一：

%主程序

function ga

[f,p]=objf(s)

gn=1;

while gn<gnmax+1

for j=1:2:inn

seln=sell(s,ps);

scro=cross(s,seln,pc);

scnew(j,:)=scross(1,:);

scnew(j+1,:)=scross(2,:);

smnew(j,:)=chang(scnew(j,:),pm);

smnew(j+1,:)=chang(scnew(j+1,:),pm);

end

s=smnew;

[f,p]=objf(s,dislist);

[fmax,nmax]=max(f);

ymean(gn)=1/mean(f);

ymax(gn)=1/fmax;

x=s(nmax,:);

gn=gn+1;

%pause;

end

gn=gn-1;

figure(2);

plot(ymax,'r'); hold on;

plot(ymean,'b');grid;

title('搜索过程');

legend('最优解','平均解');

end

function pcc=pro(pc);

test(1:100)=0;

l=round(100\*pc);

test(1:l)=1;

n=round(rand\*99)+1;

pcc=test(n);

end

%------------

function [f,p]=objf(s);

y=zeros(citynum+1,citynum+1);

for i=1:inn-1

a=s(i,:);

for j=1:KM-1

m=a(j);

n=a(j+1);

m=m+1;

n=n+1;

end

y(m,n)=1;

y=y';

for i=1:citynum

for j=1:citynum

mubiaob=c(i,j)\*y(i,:);

end

end

xuq1=0;

for i=1:citynum

for j=1:citynum

xuq1=xuq1+s(i)\*y(i,:)-q(i);

end

xuqiu=max((xuq1),0)\*M;

end

end

shij1=0;

shij2=0;

for i=1:citynum

for j=1:citynum

for l=1:citynum

shij1=shij1+t(i)-a(i);

shij2=shij12+b(i)-t(i);

end

shij3=max((shij1),0);

shij4=max((shij2),0);

shijian=M\*shij3+M\*shij4;

end

end

f=mubiao+xuqiu+shijian;

f=1/f;

end

end

%------

fsum=0;

for i=1:inn

fsum=fsum+f(i);

end

for i=1:inn

ps(i)=f(i)/fsum;

end

%计算累积概率

p(1)=ps(1);

for i=2:inn

p(i)=p(i-1)+ps(i);

end

p=p';

p

end

%---------------------

function seln=sell(s,p)

inn=size(p,1);

for i=1:2

r=rand; %产生一个随机数

prand=p-r;

j=1;

while prand(j)<0

j=j+1;

end

seln(i)=j; %选中个体的序号

end

sel1=seln(1);

sel2=seln(2);

end

%--------------------------------------------------------------------------

%“交叉”操作

function snew=cross(A,B,pc)

A=s(sel1,:);

B=s(sel2,:);

c=find(A==0);

d=find(B==0);

a=sym(A);

b=sym(B);

k=size(a,2);

for i=1:size(a,2)

for j=1:k

e(i,c(k))=a(i+k-1);

end

end

for i=1:size(a,2)

for j=1:k

e(i,d(k))=b(i+k-1);

end

end

c0=round(rand\*(k-1))+1;

c1=round(rand\*(k-1))+1;

a=[f(:,c0),a];

b=[e(:,c1),b];

for i=1:size(a,2);

j=1:size(e,2)

if a(i)==e(j)

a(i)==[];

end

end

for i=1:size(b,2);

j=1:size(f,2)

if b(i)==f(j)

b(i)==[];

end

end

a=double(a);

b=double(b);

g=zeros(size(A));

for i=1:size(a)

for j=1:size(c)

g(i+j)=a(i);

end

end

h=zeros(size(A));

for i=1:size(b)

for j=1:size(d)

h(i+j)=b(i);

end

end

g=g';

h=h';

snew=[g h];

%--------------------------------------------------------------------------

%变异

function snew=chang(snew,pm)

bn=size(snew,2);

snnew=snew;

c2=round(rand\*(bn-2))+1;

c3=round(rand\*(bn-2))+1;

chb1=min(c2,c2);

chb2=max(c3,c2);

x=snew(chb1+1:chb2);

snnew(chb1+1:chb2)=fliplr(x);

pmm=pro(pm);

if pmm==1

c2=round(rand\*(bn-2))+1;

c3=round(rand\*(bn-2))+1;

chb1=min(c2,c3);

chb2=max(c2,c3);

x=snew(chb1+1:chb2);

snnew(chb1+1:chb2)=fliplr(x);

end

end

end

end

问题二：

plot(m,n,'.r') %»­Â·Í¼ for i=1:272 Y1=Road(i,3); X1=Road(i,2); Y2=Road(i,5); X2=Road(i,4);

line([X1,X2],[Y1,Y2]); hold on end

for x=1:19

str= num2str(Road(x,1)); plot(X1(x),Y1(x),'ro')

m=jpoint(:,1); n=jpoint(:,2); plot(m,n,'.r');

x=[2,3,4,0,2,3,0,1,4]; y=[2,2,2,3,3,3,4,4,4];

z=[80,82,84,79,61,65,84,84,86];

subplot(2,1,1);stem3(x,y,z);title('RAW DATA'); xi=0:0.1:4;yi=2:0.2:4; [XI,YI]=meshgrid(xi,yi);

ZI=griddata(x,y,z,XI,YI,'v4');

subplot(2,1,2);mesh(XI,YI,ZI);title('GRIDDATA');

function [d,r]=floyd(a) n=size(a,1); d=a;

for i=1:n for j=1:n r(i,j)=j; 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); r(i,j)=r(i,k)31

end  end  end end

 clc,clear

a(1,2)=56;a(1,3)=35;a(1,4)=21;a(1,5)=51;a(1,6)=60; a(2,3)=21;a(2,4)=57;a(2,5)=78;a(2,6)=70; a(3,4)=36;a(3,5)=68;a(3,6)=68; a(4,5)=51;a(4,6)=61; a(5,6)=13; a(6,:)=0; a=a+a';

c1=[5 1:4 6]; L=length(c1); flag=1;

while flag>0 flag=0; for m=1:L-3 for n=m+2:L-1

if a(c1(m),c1(n))+a(c1(m+1),c1(n+1))<a(c1(m),c1(m+1))+a(c1(n),c1(n+1)) flag=1;

c1(m+1:n)=c1(n:-1:m+1); end end end End sum1=0; for i=1:L-1

sum1=sum1+a(c1(i),c1(i+1)); end

circle=c1; sum=sum1;

c1=[5 6 1:4];%改变初始圈，该算法的最后一个顶点不动 flag=1;

while flag>0 flag=0; for m=1:L-3 for n=m+2:L-1

if a(c1(m),c1(n))+a(c1(m+1),c1(n+1))< a(c1(m),c1(m+1))+a(c1(n),c1(n+1)) flag=1;

c1(m+1:n)=c1(n:-1:m+1); end32

end end end sum1=0; for i=1:L-1

sum1=sum1+a(c1(i),c1(i+1)); end

if sum1<sum sum=sum1; circle=c1; end

circle,sum

运行结果： n=3；

问题三：

clc;close all;clear all; X=107.972554615162; Y=26.6060305362822; file='D:\test\1.xls';

[data text]=xlsread(file); y=data(:,3); x=data(:,2);

plot(x,y,'+k',X,Y,'pr'); for n=1:16764

plot(data(n,2),data(n,3),'ro')

str=['(' num2str(x(n)) ',' num2str(y(n)) ')']; text(x(n)+0.2,y(n)+1,str); end

for i=1:45442 Y1=DATA(:,2); X1=DATA(:,1); Y2=DATA(:,4); X2=DATA(:,3);

line([X1(i),X2(i)],[Y1(i),Y2(i)]); str= num2str(i);

text(X1(i)+0.2,Y1(i)+1,str) end

%»­pieÍ¼

X=107.972554615162; Y=26.6060305362822; y=Customer(:,2); x=Customer(:,1);

plot(x,y,'+k',X,Y,'pr'); hold on

m=jpoint(:,1); n=jpoint(:,2); plot(m,n,'.r')

%»­pieÍ¼

X=107.972554615162; Y=26.6060305362822; y=Customer(:,2); x=Customer(:,1);

plot(x,y,'+k',X,Y,'pr'); hold on

m=jpoint(:,1); n=jpoint(:,2);

plot(m,n,'.r') %»­Â·Í¼ for i=1:272 Y1=Road(i,3); X1=Road(i,2); Y2=Road(i,5); X2=Road(i,4);

line([X1,X2],[Y1,Y2]); hold on end

for x=1:19

str= num2str(Road(x,1)); plot(X1(x),Y1(x),'ro')

1 m=jpoint(:,1); n=jpoint(:,2); plot(m,n,'.r');

x=[2,3,4,0,2,3,0,1,4]; y=[2,2,2,3,3,3,4,4,4];

z=[80,82,84,79,61,65,84,84,86];

subplot(2,1,1);stem3(x,y,z);title('RAW DATA'); xi=0:0.1:4;yi=2:0.2:4; [XI,YI]=meshgrid(xi,yi);

ZI=griddata(x,y,z,XI,YI,'v4');

subplot(2,1,2);mesh(XI,YI,ZI);title('GRIDDATA');

10.2.6 floyd算法matlab代码 function [d,r]=floyd(a) n=size(a,1); d=a;

for i=1:n for j=1:n r(i,j)=j; 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); r(i,j)=r(i,k)

end end end

1

a(1,2)=56;a(1,3)=35;a(1,4)=21;a(1,5)=51;a(1,6)=60; a(2,3)=21;a(2,4)=57;a(2,5)=78;a(2,6)=70; a(3,4)=36;a(3,5)=68;a(3,6)=68; a(4,5)=51;a(4,6)=61; a(5,6)=13; a(6,:)=0; a=a+a';

c1=[5 1:4 6]; L=length(c1); flag=1;

while flag>0 flag=0; for m=1:L-3 for n=m+2:L-1

if a(c1(m),c1(n))+a(c1(m+1),c1(n+1))<a(c1(m),c1(m+1))+a(c1(n),c1(n+1)) flag=1;

c1(m+1:n)=c1(n:-1:m+1); end end end End sum1=0; for i=1:L-1

sum1=sum1+a(c1(i),c1(i+1)); end

circle=c1; sum=sum1;

c1=[5 6 1:4];%改变初始圈，该算法的最后一个顶点不动 flag=1;

while flag>0 flag=0; for m=1:L-3 for n=m+2:L-1

if a(c1(m),c1(n))+a(c1(m+1),c1(n+1))< a(c1(m),c1(m+1))+a(c1(n),c1(n+1)) flag=1;

c1(m+1:n)=c1(n:-1:m+1); end

end end sum1=0; for i=1:L-1

sum1=sum1+a(c1(i),c1(i+1)); end

if sum1<sum sum=sum1; circle=c1; end

circle,sum 运行结果：

circle = 1 2 4 6 7 5 3 8 sum = 84.4332

10.2.8 0-1规划lingo代码 MODEL: SETS:

Logistics/L1..L9/:x; Customer/C1..C100/:w;

Link(Logistics,Customer):z,d; ENDSETS DATA:

w=@OLE('D:\weight.XLS'); d=@

OLE('D:\distance.XLS'); ENDDATA

Min=@sum(customer(j):w(j)\*link(i,j):d(i,j)\*z(i,j)); x(9)=1;

@for(customer(j):

@sum(link(i.j):z(i,j))=1); @for(logistics(i):

@sum(Logistics(i):x(i))=6) @for(Link(i,j):@bin(z)); @for(L(i):@bin(x)); END