评委一评分，签名及备注	队号： 10476	评委三评分，签名及备注
评委二评分，签名及备注	选题： C	评委四评分，签名及备注

题目：

摘要

本文研究的是最佳旅游路线的选择问题，此问题属于旅行商问题，我们建立了路径最短，花费最少，省钱、省时、方便三个模型。根据不同需求，我们用蚁群算法、改良圈算法和多目标规划解决了该问题，之后我们结合实际情况对三个模型进行科学地误差分析，并分析了该算法的复杂性。

针对问题一，根据 Global mapper 查找出 10 个景点的经纬度，设计一条最短的旅行路线，即从驻地出发，经过每个景点恰好一次，再回到驻点。我们根据实际的地理位置经纬度，利用几何知识计算出每两个城市之间的距离。其形式化描述为：用节点来表示 10 个城市,连接两节点之间的边表示旅行路线,并将路线的长度等属性表示为该边的权值,那么就可以把城市旅行网络抽象为一个带权有向图。给定一个带权有向图 G 为二元组 G=(V,{E}),其中 V 是包含 n 个节点的集合,E 是包含 h 条边(弧段)的集合,（i, j）是 E 中从节点 i 至 j 的边，w_{ij} 是边（i,j）的非负权值。设 S,T 分别为 V 中的起始节点和目标节点,则最优路径问题就是指在带权有向图G中,则最优路径问题就是指在带权有向图G中,寻找从指定起始节点到目标节点的一条具有最小权值总和的路径。其最短的旅游线路长度为 92.2 公里。

针对问题二，该问题的目的是设计最经济的旅行方案，即最少路程花费。我们运用改良圈算法求解旅行商问题，以任意两点之间路程的最少花费矩阵为权重，利用 $w_1(i,j)_{10\times10}$ 邻接矩阵构造无向图 UG_1，据题意不知起始地点，因此利用 Matlab 软件重复进行 10 次改良圈算法即以每一个城市为出发点，从 10 个 Hamilton 圈得到了最优圈 $circle_1$，即路费最少的旅行路线，其最少花费为 185 元。

针对问题三，这里以享受为主，在模型一、模型二结果的基础上，我们设定原则：优先考虑方便，当两地乘坐出租车所用的费用比乘坐豪华大巴所用费用高不出某个范围时，则乘坐出租车。此处通过动态规划来实现此方案，在最经济、最短的路线的基础之上，通过改换乘坐方式，使最终的花费偏离出最小花费的值在我们的允许范围内，从而达到了省钱、省时又方便的目的。最终得到满足游客自身需要的旅行方案,总花费 1643 元（不包含住宿费、伙食费的情况下）。

之后我们结合实际情况对三个模型进行科学误差分析，并分析了所用算法的复杂性，同时对我们解决旅行商所采用的算法进行了评价，这使我们对旅行商问题有了更深一步的理解。

关键字：旅行商问题　蚁群算法　改良圈算法　动态规划　误差分析

最佳旅游路线的选择模型

1 问题重述

暑假即将来临，很多家长会选择这个时间带孩子去某城市旅游，但不同的家庭有不同的需求(人数，费用限制，时间限制等)，请您任选一个旅游城市(比如你所在的城市)，综合考虑旅行路线，费用、时间以及其它你认为比较重要的因素，为有不同需求的家庭设计一份最佳旅游套餐。

根据不同的需要，我们可以把问题分为从三个不同的方面来考虑：

（1）按地理位置（经纬度）设计最短路旅行方案。

（2）假设任意两个城市之间都有豪华大巴和出租车，乘坐出租车的价格是两点间距离 2 倍（单位：元），豪华大巴的价格是分段的，在 30 公里之内是距离的 2.5 倍，超过 30 公里且在 70 公里之内的是距离的 1.7 倍，超过 70 公里的是距离的 1.4 倍，如果某家庭可选择出租车、豪华大巴，设计最经济的旅行方案。

（3）在综合实际情况下考虑省钱、省时又方便，设定出相应的评价准则和指标，建立相应改进后的数学模型，并在此前提下修订上述两种方案。假设豪华大巴和出租车都可以随到随走，出租车的速度是 80 公里/小时，豪华大巴的速度是 50 公里/小时。

2 条件假设与符号约定

2.1 条件假设

（1）计算景点之间的距离时忽略地形如丘陵盆地等自然因素对计算结果的影响；

（2）假设在旅途中的车速一定，且不考虑突发事件干扰出租车或豪华大巴的行程；

（3）假设任意两个景点之间都有豪华大巴和出租车，乘坐出租车的价格是两点间距离 2 倍（单位：元），豪华大巴的价格是分段的，在 30 公里之内是距离的 2.5 倍，超过 30 公里且在 70 公里之内的是距离的 1.7 倍，超过 70 公里的是距离的 1.4 倍；

（4）假设在每个景点停留时间为一天。

2.2 符号约定

n：表示城市的个数；

d_{ij}：两个城市$i与j$之间的距离，$1<i+1<j<10$；

$$x_{ij}=\begin{cases}1, & 表示走过城市i到城市j的路\\ 0, & 表示没有选择走这条路\end{cases}$$

C：初始圈；

C_{ij}：C的改良圈，$1<i+1<j<10$；

$\sum_{j=1}^{n} x_{ij}=1$：每个点只有一条边出去，$i=1,2,\cdots,10$；

$\sum_{i=1}^{n} x_{ij}=1$：每个点只有一条边出去，$j=1,2,\cdots,10$；

$F(i,j)_{10\times 10}$：任意两点之间所花费的最小费用构成的距阵。

3 问题分析

3.1 问题一

该问题是一个旅行路径的组合优化问题，其形式化描述为：用节点来表示 10 个景点,连接两节点之间的边表示旅行路线,并将路线的长度等属性表示为该边的权值,那么就可以把景点旅行网络抽象为一个带权有向图。给定一个带权有向图 G 为二元组 G=(V,{E}),其中 V 是包含 n 个节点的集合,E 是包含 h 条边(弧段)的集合,（i,j）是 E 中从节点 i 至 j 的边, w_{ij} 是边（i,j）的非负权值。设 S,T 分别为 V 中的起始节点和目标节点,则最优路径问题就是指在带权有向图G中,则最优路径问题就是指在带权有向图G中,寻找从指定起始节点到目标节点的一条具有最小权值总和的路径。

在不同需求条件的影响下，景点旅行路线网不仅有道路路线等物理属性,同时也具有路线长度、旅行时间、车票价格等各种其它逻辑属性，因而改变着旅行的最优路线。

问题一中，我们根据实际的地理位置经纬度，利用几何知识计算出每两个城市之间的距离，并以距离为权，进而建立模型，求解。设定出游遍 10 个景点的最优旅行路线。

3.2 问题二

根据 10 个景点的经纬度，设计一条最经济的旅行路线，即从驻地出发，经过每个景点恰好一次且花钱最少。由此可知，此问题是属于旅行商问题，我们可以考虑运用改良圈算法求解此问题。按景点顺序给各景点编号 $1,2,\cdots,10$ ，根据问题一中求出的各个景点间的距离，我们可以考虑以任意两点之间的车程最少花费为权重，利用 $w(i,j)_{10\times10}$ 构造无向图 UG_1，考虑到没有给出起点，如果以某一景点为出发点，利用改良圈算法得到的最优圈未必是最优解，所以我们将利用 Matlab 软件编程重复进行 10 次改良圈算法，将会得到最优圈 $circle_1$，从而保证最优解，即最经济的旅行路线。用终点返回起点构成的闭合回路为最经济的路线。这样就会设计出一条最经济的旅游线路。

3.3 问题三

针对问题三，在模型一、模型二结果的基础上，我们可以考虑设定原则：优先考虑方便，当两地乘坐出租车所用的费用比乘坐豪华大巴所用费高不出某个范围时，则乘坐出租车。考虑通过动态规划来实现此方案，在最经济、最短的路线的基础之上，通过改换乘坐方式，若最终的花费偏离出最小花费在我们的允许范围内，则接受此方案，达到了省钱、省时又方便的目的。最终得到满足自身需要的旅行方案。

4 模型建立及求解

4.1 问题一的模型建立与求解

根据问题分析，我们首先创建景点旅行网络图，给定加权图 $G=(V_G,E_G)$ ，其中 V_G 表示顶点（景点）的集合表示为： $V_G=\{1,2,\cdots,10\}$ 。E_G 为赋权边(表示两个景点间的距离)的集合，即两地间距离（km）。表示为 $E_G=\{(i,j,w_{ij}),i,j\in V_G,w_{ij}\in R^+\}$ 。

Hamilton 路径图可以表示为：$S=<v_1v_2\cdots v_nv_1>$；其中 $v_i\in V_G$，$1\le\forall i\ne j\le 34, v_i\ne v_j$，且对$1\le i\le 10$，有$(v_i,v_{(i\bmod n)+1},w_{v_iv_{(i\bmod n)+1}})\in E_G$。记$H(G)$为$G$中所有 Hamilton 回路的集合，定义 $w(S)=\sum_{1\le i\le n-1}w_{v_iv_{i+1}}+w_{v_nv_1}$；目的是要寻找一条最短的 Hamilton 回路$S^*$， 使得$w(S^*)=\min_{S\in H(G)}w(S)$。

利用蚁群算法对所得数据进行搜索计算。

4.1.1 城市间距离的估算

通过 Global mapper 查找 10 个景点经纬度具体数值（如表 1）所示：

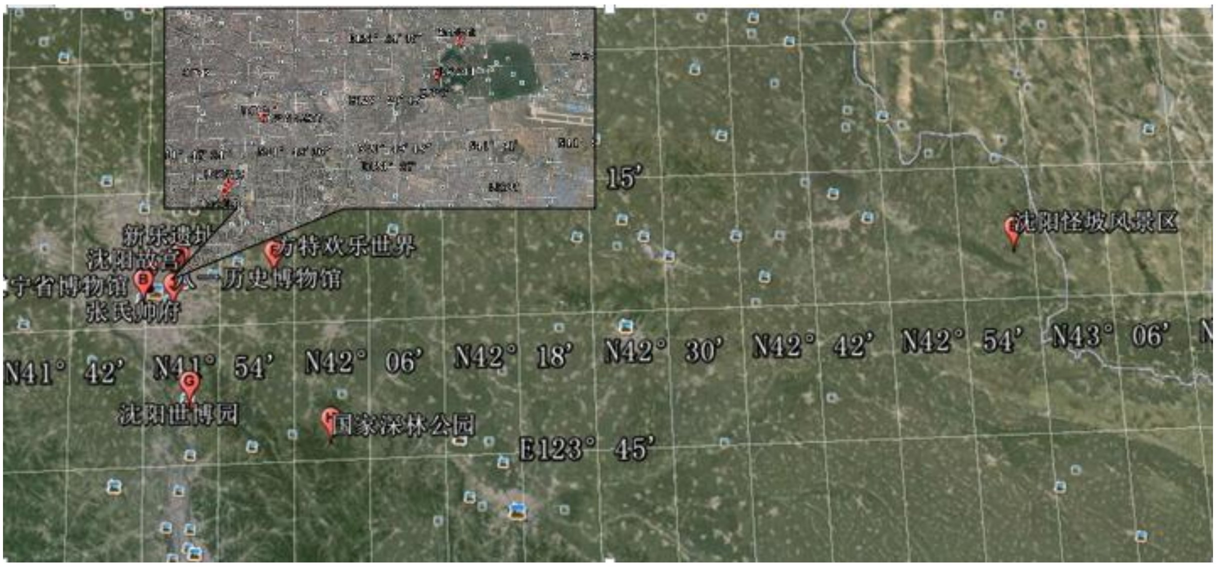


图 1　10 个景点地图位置显示

编号	景点	纬度(北纬)	经度(东经)
1	沈阳故宫	41° 47′ 49.73″	123° 27′ 20.62″
2	张氏帅府	41° 47′ 41.49″	123° 27′ 29.62″
3	北陵公园	41° 50′ 20.76″	123° 25′ 29.83″
4	新乐遗址	41° 50′ 40.87″	123° 24′ 50.03″
5	沈阳怪坡	42° 04′ 12.58″	123° 37′ 47.31″
6	方特欢乐世界	41° 58′ 20.04″	123° 24′ 49.95″
7	沈阳世博园	41° 51′ 38.70″	123° 38′ 50.40″
8	沈阳国家森林公园	42° 02′ 53.87″	123° 43′ 19.31″
9	辽宁省博物馆	41° 48′ 12.11″	123° 26′ 11.34″
10	九一八历史博物馆	41° 50′ 12.42″	123° 28′ 04.04″

表 1　各个城市经纬度

根据 10 个景点的经纬度，利用 matlab 编程（详见附表 1）绘出 10 个景点的地理位置关系。

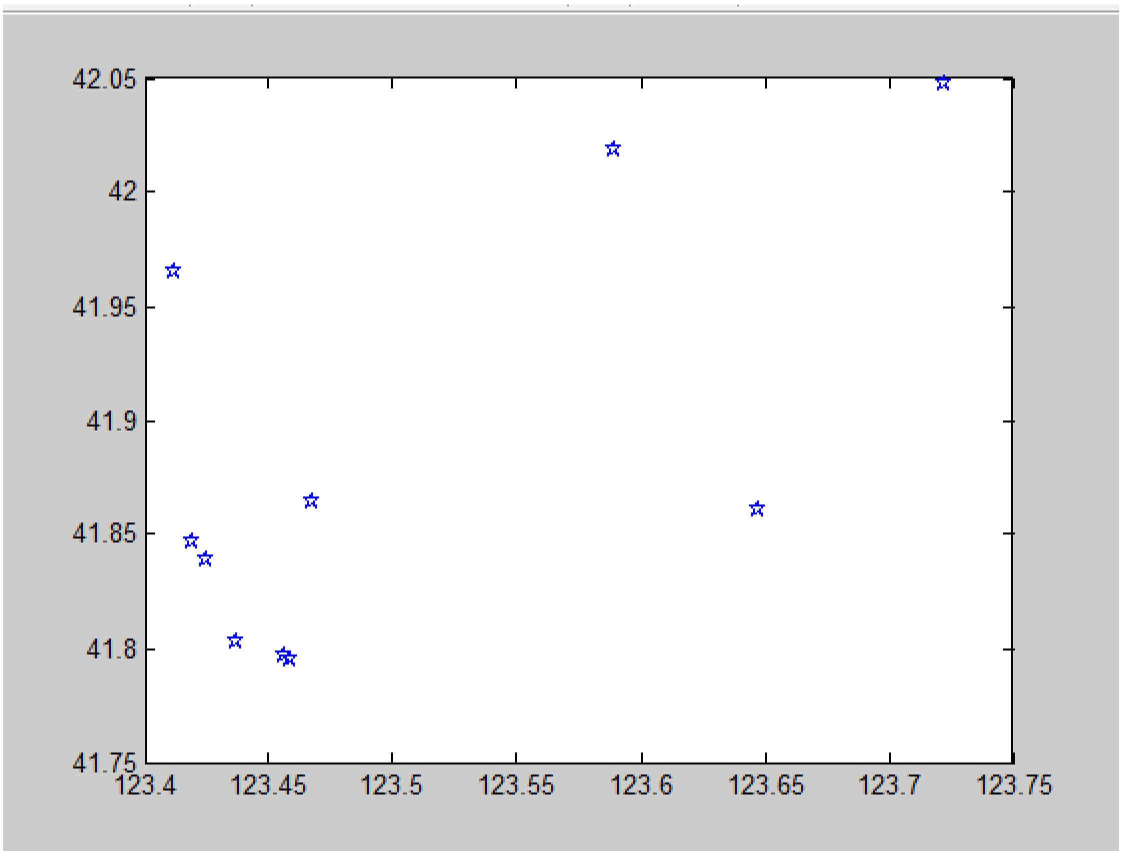


图 2　10 个景点地理位置关系图

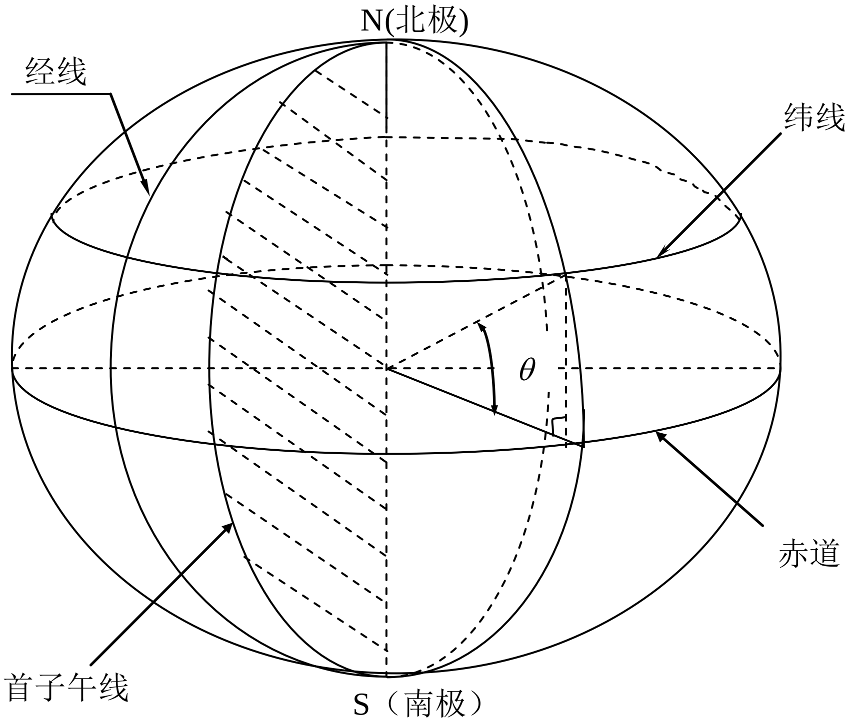


图 3　地球几何解析图

首先把地球看成是标准球体，球心为点 O，把任意两个景点看成是 A 、B 两点，则

地球上的两个点 A、 B 在赤道面上的投影点分别为 A' 、B' 。假设地球平均半径为 R，A 点北纬 θ_i，东经 φ_i，B 点北纬 θ_j，东经 φ_j，基于球体几何知识因此可以得到：

$$OA' = R\cos\theta_i,\quad OB' = R\cos\theta_j \tag{1}$$

投影下来的 $\angle A'OB' = \varphi_i - \varphi_j$ （2）

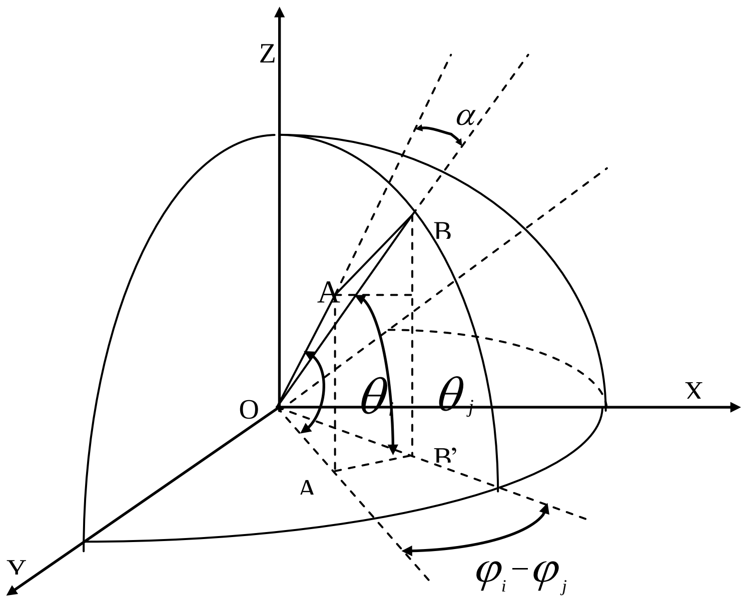


图 4　地球经纬度

利用余弦定理可得到：

$$A'B'^2 = R^2\cos^2\theta_i + R^2\cos^2\theta_j - 2\times R\cos\theta_i \times R\cos\theta_j \times \cos(\varphi_i - \varphi_j) \tag{3}$$

$$AB'' = R\times(\sin\theta_i - \sin\theta_j) \tag{4}$$

利用勾股定理得到：

$$AB^2 = A'B'^2 + AB''^2 \tag{5}$$

将（3）、（4）代入（5）式中：

$$AB^2 = 2R^2 - 2R^2\cos\theta_i\cos\theta_j\cos(\varphi_i - \varphi_j) - 2R^2\sin\theta_i\sin\theta_j \tag{6}$$

令∠AOB 为 α，同样利用余弦定理可得到：

$$AB^2 = R^2 + R^2 - 2\times R\times R\times\cos\alpha \tag{7}$$

化简得：

$$\cos\alpha = \cos\theta_i\cos\theta_j\cos(\varphi_i - \varphi_j) + \sin\theta_i\sin\theta_j \tag{8}$$

则地球上两点 AB 之间距离即 AB 弧长为

$$\widehat{AB}=\alpha\times R=R\times ar\cos(\cos\theta_i\cos\theta_j\cos(\varphi_i-\varphi_j)+\sin\theta_i\sin\theta_j) \quad (9)$$

即 $\widehat{AB}$ 为两地之间的实际路径。

因此我们根据表一中所示的各景点经纬度利用 matlab 软件编程（详见附录 2）计算得到各景点间的距离（详见附录 3）。

4.1.2 蚁群算法建立模型

蚁群觅食的过程与此组合优化问题的求解相似，找到一条只经过每个景点一次且回到起点的、最短路径的回路。设景点 i 和 j 之间的距离为 d_{ij}。

求解中，假设蚁群算法中的每只蚂蚁是具有下列特征的简单智能体。

（1）每次周游，每只蚂蚁在其经过的支路（i，j）上都留下信息素。

（2）蚂蚁选择景点的概率与景点之间的距离和当前连接支路上所包含的信息素余量有关。

（3）为了强制蚂蚁进行合法的周游，直到一次周游完成后，才允许蚂蚁游走已访问过的景点（这可由禁忌表来控制）。

蚁群算法中的基本变量和常数有：m，蚁群中蚂蚁的总数；n，TSP 问题中景点的个数；d_{ij} 为景点 i 和 j 之间的距离，其中 $i,j\in(1,34)$；$\tau_{ij}(t)$，表示 t 时刻在路径（i，j）连线上残留的信息量。在初始时刻各条路径上信息量相等，并设 $\tau_{ij}(0)$=const（const 为常数）。

蚂蚁 k（k=1，2，…，m ）在运动过程中，根据各条路径上的信息量决定其转移方向。$p_{ij}^k(t)$ 表示在 t 时刻蚂蚁 k 由景点 i 转移到景点 j 的状态转移概率，根据各条路径上残留的信息量 $\tau_{ij}(t)$ 及路径的启发信息 η_{ij} 来计算的，如（10）所示。表示蚂蚁在选择路径时会尽量选择离自己距离较近且信息素浓度较大的方向。

$$p_{ij}^k(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\alpha}\cdot[\eta_{ij}(t)]^{\beta}}{\sum\limits_{s\subset allowed_k}[\tau_{is}(t)]^{\alpha}\cdot[\eta_{is}(t)]^{\beta}} & j\in allowed_k\\ 0 & 其他\end{cases} \quad (10)$$

式中，$allowed_k=\{C-tabu_k\}$—表示在 t 时刻蚂蚁 k 下一步允许选择的景点（即还没有访问的景点）；

$tabu_k$ （k=1，2，…，m ）—表示禁忌表，记录蚂蚁 k 当前已走过的景点；

α—信息启发式因子，反映了蚁群在运动过程中所残留信息量相对重要程度；

β—表示期望启发式因子，反应了期望值的相对重要程度；

η_{ij}—表示景点 i 转移到景点 j 的期望程度，具体计算公式如下

$$\eta_{ij}(t)=\frac{1}{d_{ij}} \tag{11}$$

对蚂蚁 k 而言，d_{ij}越小，则$\eta_{ij}(t)$越大，$p_{ij}^{k}(t)$也就越大。

为了避免残留信息素过多而淹没启发信息，在每只蚂蚁走完一步或者完成对所有 n 个景点的遍历后，要对残留信息素进行更新处理。（t+n）时刻在路径（i，j）上信息量可按式（3）和式（4）所示的规则进行调整。

$$\tau_{ij}(t+n)=(1-\rho)\cdot\tau_{ij}(t)+\Delta\tau_{ij}(t) \tag{12}$$

$$\Delta\tau_{ij}(t)=\sum_{k=1}^{m}\Delta\tau_{ij}{}^{k}(t) \tag{13}$$

式中，ρ—表示信息素挥发系数。模仿人类记忆特点，旧的信息将逐步忘却、削弱。

为了防止信息的无限积累，ρ的取值范围为[0，1)，用 1－ρ表示信息的残留系数。

$\Delta\tau_{ij}(t)$—表示本次循环中路径（i，j）上的信息量增量，初始时刻$\Delta\tau_{ij}(t)=0$。

$\Delta\tau_{ij}{}^{k}(t)$—表示第 k 只蚂蚁在本次循环中留在路径（i，j）上的信息量。

根据信息素更新策略

$$\Delta\tau_{ij}{}^{k}(t)=\begin{cases}\dfrac{Q}{L_K} & 第k只蚂蚁在本次循环中经过（i,j） \\ 0 & 其他\end{cases} \tag{14}$$

L_k—表示第 k 只蚂蚁在本次循环中所走路径的总长度。

这里的基本蚁群算法具体算法步骤为：

第 1 步：初始化参数。时间 t=0，循环次数$N_c=0$，设置最大循环次数$N_{c\max}$，令路径（i，j）的初始化信息量$\tau_{ij}(t)$=const，初始时刻$\Delta\tau_{ij}(t)=0$。

第 2 步：将 m 只蚂蚁随机放在 n 个景点上。

第 3 步：循环次数$N_c\leftarrow N_c+1$。

第 4 步：令蚂蚁禁忌表索引号 k=1。

第 5 步：k=k+1。

第 6 步：根据状态转移概率计算蚂蚁选择景点 j 的概率，$j=\{C-tabu_k\}$。

第 7 步：选择最大状态转移概率景点，将蚂蚁移动，把该景点计入禁忌表中。

第 8 步：若没有访问完集合 C 中的所有景点，即$k<m$，跳转至第 5 步；否则，转第 9 步。

第 9 步：根据式（13）和式（14）更新每条路径上的信息量。

第 10 步：若满足结束条件，循环结束输出计算结果；否则清空禁忌表并跳转到第 3 步。

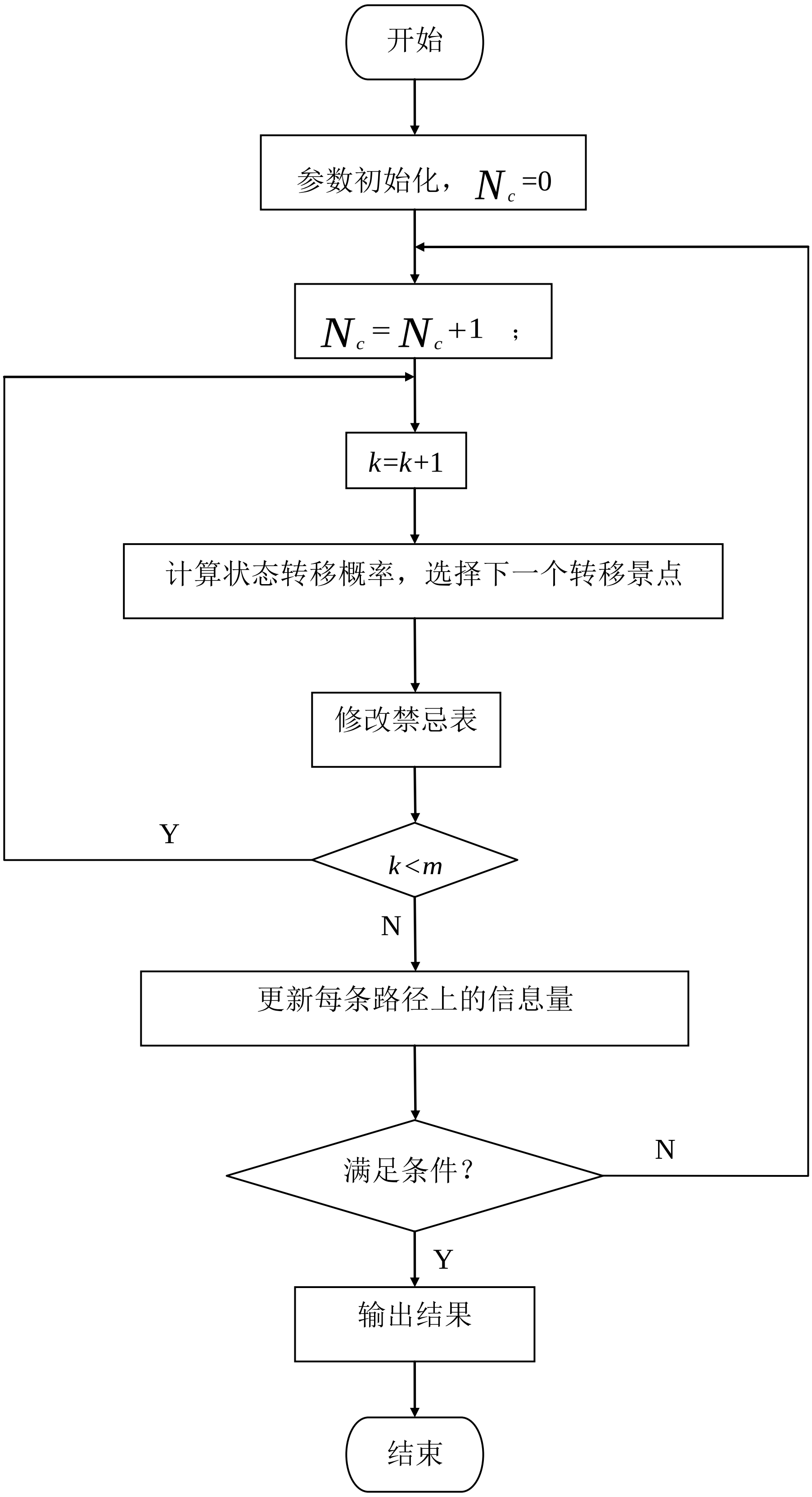


图 5　基本蚁群算法的算法框图

根据以上算法我们利用 matlab 软件编程（程序见附录 4），模拟出了游览十个景点最短的旅游线路，见下图。其中每一个‘*’表示每个城市，折线表示旅游线路得到最优旅行路径为

新乐遗址 ↔ 方特 ↔ 沈阳怪坡 ↔ 沈阳国家森林公园 ↔ 沈阳世博园

旅游最短路径线路图

北陵公园 ↔ 辽宁省博物馆 ↔ 张氏帅府 ↔ 沈阳故宫 ↔ 九一八历史博物馆

旅行路线总长为 92.12138 千米。

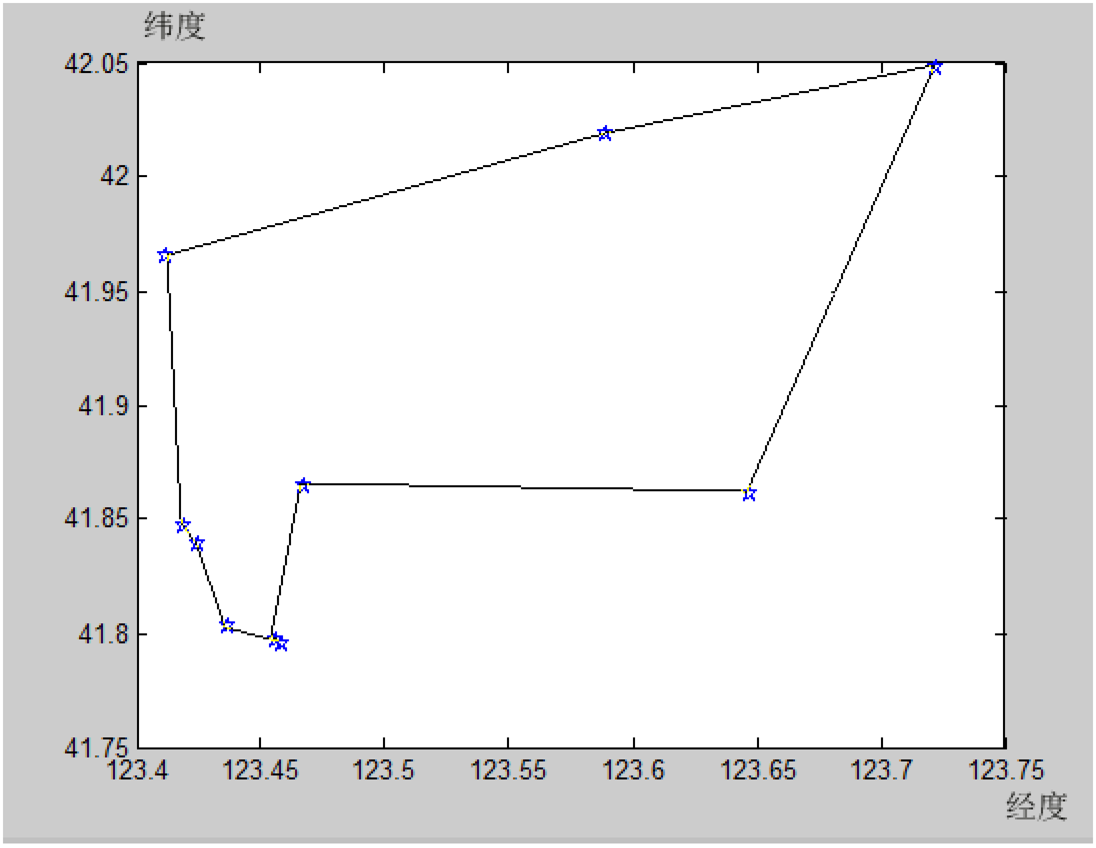


图 6　蚁群算法的最优路线图

4.2 问题二模型的建立与求解

4.2.1 旅行商问题的基本理论

某旅行商欲往 n 个城市推销货物，从某个城市出发，沿途经过各个城市一次后返回出发城市，要确定一条行走的路线，使得总路径最短。这个问题称为旅行商问题（TSP）[1]。用图论的术语说，就是在一个赋权完全图中，找出一个有最小权的 Hamilton 圈 C。称这种圈为最优圈。与最短路问题及连线问题相反，尽管目前还没有求解旅行商问题的

有效算法。但是却有一个可行的办法是求一个 Hamilton 圈，然后适当修改以得到具有较小权的另一个 Hamilton 圈。修改的方法叫做改良圈算法。设初始圈 $C = v_1 v_2 \cdots v_n v_1$。

（1）对于 $1 < i+1 < j < n,$ 构造新的 Hamilton 圈：

$C_{ij} = v_1 v_2 \cdots v_i v_j v_{j-1} v_{j-2} \cdots v_{i+1} v_{j+1} v_{j+2} \cdots v_n v_1,$

它是由 C 中删去的边 $v_i v_{i+1}$和$v_j v_{j+1}$,添加边$v_i v_j$和$v_{i+1} v_{j+1}$ 而得到的。若 $w(v_i v_j) + w(v_{i+1} v_{j+1}) < w(v_i v_{i+1}) + w(v_j v_{j+1})$，则以 C_{ij}代替C，C_{ij}叫做C的改良圈。

（2）转（1），直至无法改进，停止。

用改良圈算法得到的结果几乎可以肯定不是最优的。为了得到更高的精确度，在不给定起始位置的前提下，可以选择不同的初始圈，重复进行 n 次算法，以求得精确的结果。

4.2.2 旅行商问题的数学表达式

设城市的个数为 n，d_{ij} 是两个城市 i与j 之间的距离，$x_{ij} = 0$或1（1表示走过城市 i到城市j 的路，0 表示没有选择走这条路）。则有

$$
\begin{aligned}
&\min \sum_{i \neq j} d_{ij} x_{ij} \\
&s.t. \quad \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, 2, \cdots, n, (\text{每个点只有一条边出去}) \\
&\qquad \sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, 2, \cdots, n, (\text{每个点只有一条边出去}) \\
&\qquad \sum_{i, j \in s} x_{ij} \le |s| - 1, \quad 2 \le |s| \le n-1, \quad s \subset \{1, 2, \cdots, n\}
\end{aligned}
$$

（各起点和终点外，各边不构成圈）

4.2.3 模型一求解

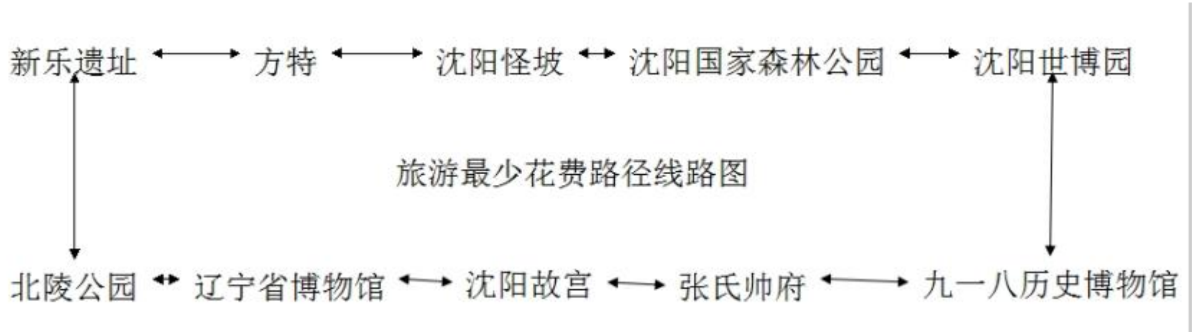
利用下面的分段函数求出花费最小的矩阵 $F(i, j)_{10 \times 10}$：

$$
\begin{cases}
F(i, j) = d(i, j) \times 2, & d(i, j) <= 30 \\
F(i, j) = 1.7 \times d(i, j), & 30 < d(i, j) <= 70 \\
F(i, j) = 1.4 \times d(i, j), & 70 < d(i, j)
\end{cases}
$$

按所给各城市的顺序编号 $1, 2, \cdots, 10$，按照上一问求出的任意两景点间距离作为两景点间距离，我们以任意两点之间的最少花费矩阵为权重矩阵，利用 $w_1(i, j)_{10 \times 10}$ 构造无向图 UG_1，据题意并不知起始地点，因此利用 Matlab 软件重复进行 10 次改良圈算法，尝

试以每一个城市为出发点（算法见附录 5），首先设 $C=v_1v_2\cdots v_nv_1$，按改良圈算法求出此时的最优圈后，改变初始圈 $C=v_2v_1\cdots v_nv_2$，依次进行下去，求出符合要求的最经济的最优圈 $circle_1$，保证了从终点返回到出发点的花费最小，即最经济旅行方案，如下：

我们运用 Matlab 软件模拟出了最经济的旅游线路，见下图。其中每一个‘*’表示每个城市，折线表示旅游线路，标有 1 的城市是旅行的起点，标有 100 的城市是旅游的终点，由终点 100 返回起点 1 所构成的闭合回路就是最经济的旅行路线，其路途最少花费为 185 元。



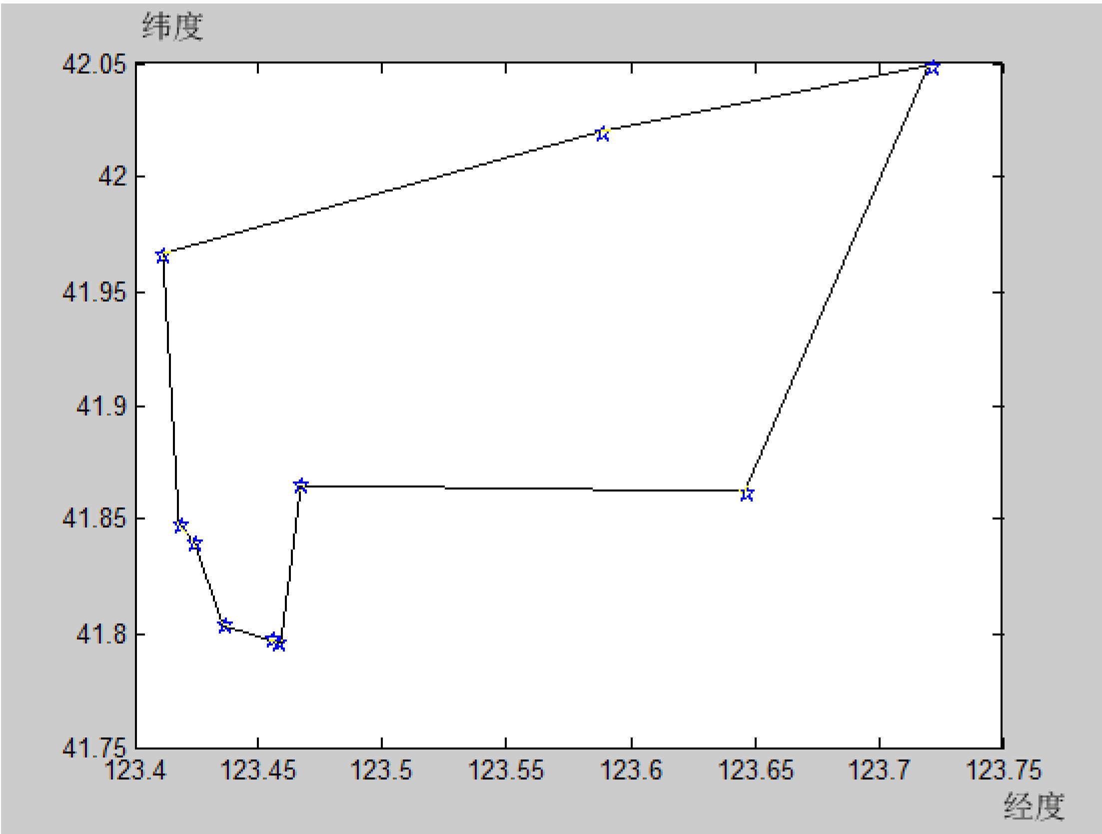


图 7　模拟最经济旅游线路示意图

4.3 问题三模型的建立与求解

4.3.1 多目标规划思想

基于选择既省钱、省时又方便的最佳旅游方案，我们建立了用序贯式算法求解多目标规划[2]的模型。序贯式算法的核心是根据优先级的先后次序，将目标规划问题分解成一系列的单目标规划问题，然后再依次求解。

求解目标规划的序贯算法：

对于 $k=1,2,\cdots,q,$ 求解单目标规划

$$\min \quad z=\sum_{j=1}^{l}(w_{kj}^{-}d_{j}^{-}+w_{kj}^{+}d_{j}^{+}) \tag{1}$$

$$s.t. \quad \sum_{j=1}^{n}a_{ij}x_{j}\le(=,\ge)b_{i},\quad i=1,\cdots,m \tag{2}$$

$$\sum_{j=1}^{n}c_{ij}x_{j}+d_{i}^{-}-d_{i}^{+}=g_{i},\quad i=1,2,\cdots,l \tag{3}$$

$$\sum_{j=1}^{l}(w_{sj}^{-}d_{j}^{-}+w_{sj}^{+}d_{j}^{+})\le z_{s}^{*},\quad s=1,2,\cdots,k-1 \tag{4}$$

$$x_{j}\ge 0,\quad j=1,2,\cdots,n \tag{5}$$

$$d_{i}^{-},d_{i}^{+}\ge 0,\quad i=1,2,\cdots,l \tag{6}$$

其最优目标值为 z_k^*，当 $k=1$ 时，约束（4）为空约束。当 $k=q$ 时，z_q^* 所对应的解 x^* 为目标规划的最优解。

4.3.2 模型的求解

所选各个景点之间距离都不大于 30 公里，故乘坐出租车为最省钱、省时的旅行方式。

景点	方式	景点	方式	景点	方式	景点	方式
1→2	出租车	7→8	出租车	6→4	出租车	9→1	出租车
2→10	出租车	8→5	出租车	4→3	出租车		
10→7	出租车	5→6	出租车	3→9	出租车		

表 2　最方便旅行景点之间交通方式选择

沈阳市景点	成人票	沈阳市景点	成人票	沈阳市景点	成人票
沈阳故宫	65	沈阳怪坡	36	沈阳国家森林公园	25
张氏帅府	50	方特欢乐世界	201	辽宁省博物馆	2
北陵公园	45	沈阳世博园	46	九一八历史博物馆	2
新乐遗址	14				

表 3　各景点门票价格　　单位：元

再根据表 3 中各景点门票费用计算出在不包含伙食费、住宿费情况下的一个正常家庭（按三口之家算）在最省时、最省钱情况下的总花费为 1643 元。

5 模型的科学地可行性、误差分析

上述算法思想是理想化的模型，在实际的旅游问题中，不仅要考虑路程的长短，还要考虑时间和费用。通过建立数学模型的机理进行了深入剖析后，我们有效的掌握了解“旅游线路”问题，即寻找出一条既省钱、省时又方便的最佳旅游方案。现在就根据上述算法做出较科学可行性和误差分析。

在实际问题中，我们考虑的不只是问题中所给的交通工具，现实中各交通工具的速度、票价也各不相同，甚至有些景点之间没有直达车，从省钱的角度考虑，可以全部选择飞机旅行，由于所有客运价格均由铁道部规定，而模型一采用的是根据景点的经纬度计算出两景点之间的距离，不考虑突发事件干扰出租车或豪华大巴的行程的情况下，可以在出行之前按照最短路径，可以预算出旅行所花费的最短时间和费用，具有一定的可行性，实际性。但是一般情况下路程越长花费的时间越多，所花的路费也越多，但在实际情况中，按照出租车、豪华大巴运行的轨迹来看并不一定是最短的，因此所花费的时间和金钱就不一定是最少的。而模型一是按照两景点之间的直线距离来求最短路径，这样模型一中必然存在着较大误差。一般情况下，所采用的交通工具确定，两城市间的旅行时间就确定了。所以用时间来计算既省时又经济的旅行路径，产生的误差更小，因此模型二与实际联系更加紧密。使模型贴近实际，通用性强。

6 模型的复杂性分析

6.1 蚁群算法的复杂度分析

蚁群算法的复杂度分析是在理论上对蚁群算法的算法效率的分析。对于上述模型中的蚁群算法，10 个城市是 TSP 的规模，m 为蚂蚁的数量，N_c 为算法的循环次数，从蚁群的算法过程我们得到各环节的时间复杂度，如下表所示：

Step	内容	T(*n*)
1	初始化：set t=0；set N_c=0 set $\tau_{ij}(t)=const$，$\Delta\tau_{ij}(t)$=0；	$O(n^2+m)$
2	设置蚂蚁禁忌表 set s=0； for k= 1 to m do 置第 k 个蚂蚁的起始城市到禁忌表 $tabu_k$	O(m)
3	每只蚂蚁单独构造解 循环计算直到禁忌表满 set s=s+1； for k= 1 to m do 根据转移概率 $p_{ij}^k(t)$ 选择下一个城市 将城市序号 j 加入禁忌表 $tabu_k$	O(n^2m)

4	解的评价和轨迹更新量的计算 将第 k 只蚂蚁在从禁忌表 $tabu_k(n)$ 转移到 $tabu_k(1)$ 计算第 k 只蚂蚁在本次循环中的路径长度 更新最优路径 计算各条路径的信息数反馈量 $\Delta\tau_{ij}(t)$	O（n^2）
5	信息数轨迹浓度的更新 计算各条路径在下一轮循环前的信息数强度 $\tau_{ij}(t+n)$ set t =t+n; set $N_c = N_c + 1$	O（n^2）
6	判定是否达到终止条件 如果 $N_c \le N_{c\max}$，且搜索出现停止现象 清空全部禁忌表 返回 step2 否则 输出最短路径 结束	O（nm）

6.2 动态规划算法的复杂度分析

问题三中，所旅游的景点数目较多，我们用动态规划方法求解，毫无疑问，计算量和储存量都很大，在求解时耗费了较多的时间。

7 模型的评价与改进

7.1 模型的评价

本文研究的是最佳旅游路线的选择问题，此问题属于旅行商问题，我们建立了路径最短，耗时最少，经济、省时、方便三个模型，根据不同需求，我们用蚁群算法、改良圈算法和多目标规划解决了选择最佳路线的问题。

针对问题一，知此问题属于旅行商问题。首先，我们按表中各城市的顺序编号 $1,2,\cdots,10$，我们根据实际的地理位置经纬度，利用几何知识计算出每两个景点之间的距离，简化问题。然后，我们运用蚁群算法求解旅行商问题，以任意两点之间的最短距离矩阵为权重，利用 Matlab 软件编程计算得到最短的旅行路线。针对问题二，我们运用改良圈算法，将权重矩阵设置为“最少花费”，建立模型二。同样利用 Matlab 软件进行 10 次改良圈算法，就会得到最优圈，即花费最少的旅行路线。针对问题三，这里在模型一、模型二结果的基础上，我们设定原则：优先考虑方便，当两地乘坐出租车所用的费用比乘坐豪华大巴所用费用高不出一个范围时，则乘坐出租车。我们给任意两地之间耗时最少的矩阵和花费最少的矩阵赋不同的权值。通过动态规划来实现，最终得到最佳的旅行方案。

之后我们结合实际情况对三个模型进行科学误差分析，并分析了所用算法的复杂

性，同时对我们解决旅行商所采用的算法进行了评价，这使我们对旅行商问题有了更深一步的理解。

7.2 模型的改进

模型一中我们把问题简单化，我们是根据经纬度计算出两个景点的距离，而实际上两地间的距离是因实际情况曲直多变的，以计算的距离代替实际距离，这种假设只是为了解题方便，模型进一步完善就要把路程更接近于实际得到的路程，这样才更有说服力。

模型二中我们得出的结果考虑了旅游的交通费用和景点费用，都忽略了政府的宏观调控、市场经济、石油价格变化等对票价的的影响和天气变化对旅游路程和时间的影响，事实上，在我国政府的策略，市场经济，石油价格等因素都会对旅游票价有影响，天气的变化也会影响旅游进程，所以，模型的改进要考虑政策的影响以及天气变化的影响。

模型三中综合了模型一和模型二的结果，在此的基础上再考虑旅游交通方式的问题得出最优解，因此模型三中有模型一、二的影响，并且没有考虑景点的交通问题，所以，模型的改进就是要考虑景点交通问题的影响。

8 参考文献

[1]司守奎.数学建模算法与程序[M]，第五章图与网络，2007.
[2]王振龙.时间序列分析，北京：中国统计出版社，2002 年。
[3]韩中庚.长江水质综合评价与预测的数学模型，工程数学学报第 22 卷第 7 期，2005 年。
[4] 姜启源.谢金星，叶俊.北京:高等教育出版社,2005.
[5] 郑宝东.线性代数与空间解析几何.北京:高等教育出版社,2003.
[6] 张志涌.杨祖樱.matlab 教程.北京：北京航空航天大学出版社,2006.
[7] 叶其孝.大学生数学建模辅导教材.长沙：湖南教育出版社，1993.
[8] 韩中庚.实用运筹学.北京：清华大学出版社，2007.
[9]谢金星，薛毅，《优化建模与 LINDO/LINGO 软件》，北京：清华大学出版社，2005 年 7 月
[10]韩中庚，《数学建模方法及其应用》，北京：高等教育出版社，2006 年 7 月
[11]中北大学学报(自然科学版)，《用遗传算法求解旅行商问题》，2007 年第 28 卷第 1 期

附录：

附录 1：

```
x1=123.45574;x2=123.45819;x3=123.42492;x4=123.41893;x5=123.58896;x6=123.41196;x7=123.64732;x8=123.72206;x9=123.43648;x10=123.46709;y1=41.79708;y2=41.79479;y3=41.83902;y4=41.84718;y5=42.01887;y6=41.96573;y7=41.86067;y8=42.04823;y9=41.80329;y10=41.86441;
plot(x1,y1,'p')
hold on;
plot(x2,y2,'p')
hold on;
plot(x3,y3,'p')
hold on;
plot(x4,y4,'p')
hold on;
plot(x5,y5,'p')
hold on;
plot(x6,y6,'p')
hold on;
plot(x7,y7,'p')
hold on;
plot(x8,y8,'p')
hold on;
plot(x9,y9,'p')
hold on;
plot(x10,y10,'p')
hold on;
```

附录 2：

（1）利用中国城市经纬度，计算任意两座城市之间的经纬度程序：

```
a=[41.79708 41.79479 41.83902 41.84718 42.01887 41.96573 41.86067 42.04823 41.80329 41.86441 ];%各个城市纬度
b=[123.45574 123.45819 123.42492 123.41893 123.58896 123.41196 123.64732 123.72206 123.43648 123.46709];%各个城市经度
for i=1:10,
    for j=1:10,
c(i,j)=6371.3*acos(cos(a(1,i)*pi./180)*cos(a(1,j)*pi./180)*cos(b(1,i)*pi./180-b(1,j)*pi./180)+sin(a(1,i)*pi./180)*sin(a(1,j)*pi./180))%利用经纬度计算地球上两点距离
    end
end
```

附录 3（10 个城市任意两地距离 km）：

	沈阳故宫	张氏帅府	北陵公园	新乐遗址	沈阳怪坡
沈阳故宫	0	0.3257	5.3173	6.3516	27.0151
张氏帅府	0.3257	0	5.6385	6.6727	27.1665
北陵公园	5.3173	5.6385	0	1.0342	24.1691
新乐遗址	6.3516	6.6727	1.0342	0	23.7138
沈阳怪坡	27.0151	27.1665	24.1691	23.7138	0
方特欢乐世界	19.101	19.3901	14.1309	13.1954	15.7771
沈阳世博园	17.378	17.2993	18.5786	18.9763	18.2422
沈阳国家森林公园	35.5739	35.6499	33.8413	33.591	11.4679
辽宁省博物馆	1.7395	2.0328	4.087	5.0926	27.0906
九一八历史博物馆	7.5459	7.7768	4.4913	4.4251	19.9155

	方特欢乐世界	沈阳世博园	沈阳国家森林公园	辽宁省博物馆	九一八历史博物馆
沈阳故宫	19.101	17.378	35.5739	1.7395	7.5459
张氏帅府	19.3901	17.2993	35.6499	2.0328	7.7768
北陵公园	14.1309	18.5786	33.8413	4.087	4.4913
新乐遗址	13.1954	18.9763	33.591	5.0926	4.4251
沈阳怪坡	15.7771	18.2422	11.4679	27.0906	19.9155
方特欢乐世界	0	22.7113	27.216	18.1771	12.1553
沈阳世博园	22.7113	0	21.7532	18.5981	14.9317

沈阳国家森林公园	27.216	21.7532	0.0001	36.0571	29.3664
辽宁省博物馆	18.1771	18.5981	36.0571	0	7.2543
九一八历史博物馆	12.1553	14.9317	29.3664	7.2543	0

附录 4

蚁群算法求解程序：

```
%蚁群算法选择路径
function
[R_best,L_best,L_ave,Shortest_Route,Shortest_Length]=ACATSP(d,NC_max,m,
Alpha,Beta,Rho,Q)
%d 为任意两城市间距离
%NC_max 最大迭代次数
%m 蚂蚁个数
%Alpha 表征信息素程度参数
%Beta 表征启发因子程度参数
%Rho 信息素蒸发系数
%Q 信息素增加强度
n=34;
Eta=1./d;
Tau=ones(n,n);
Tabu=zeros(m,n);
NC=1;    %迭代计数器，记录次数
R_best=zeros(NC_max,n);
L_best=inf.*ones(NC_max,1);
L_ave=zeros(NC_max,1);
while NC<=NC_max   %停止条件一，达到迭代次数
%将 m 个蚂蚁放到 n 个城市上
Randpos=[];
for i=1:(ceil(m./n))
Randpos=[Randpos,randperm(n)];
end
Tabu(:,1)=(Randpos(1,1:m))';
for j=2:n
for i=1:m
visited=Tabu(i,1:(j-1));
J=zeros(1,(n-j+1));
P=J;
Jc=1;
for k=1:n
```

```
if length(find(visited==k))==0
J(Jc)=k;
Jc=Jc+1;
end
end
for k=1:length(J)
P(k)=(Tau(visited(end),J(k))^Alpha)*(Eta(visited(end),J(k))^Beta);
end
P=P/(sum(P));
Pcum=cumsum(P);
Select=find(Pcum>=rand); %若计算的概率大于原来的就选择这条路线
to_visit=J(Select(1));
Tabu(i,j)=to_visit;
end
end
if NC>=2
Tabu(1,:)=R_best(NC-1,:);
end
%记录本次迭代最佳路线
L=zeros(m,1);
for i=1:m
R=Tabu(i,:);
for j=1:(n-1)
L(i)=L(i)+d(R(j),R(j+1));
end
L(i)=L(i)+d(R(1),R(n));
end
L_best(NC)=min(L);
pos=find(L==L_best(NC));
R_best(NC,:)=Tabu(pos(1),:);
L_ave(NC)=mean(L);
NC=NC+1        %迭代继续
%更新信息素
Delta_Tau=zeros(n,n);
for i=1:m
for j=1:(n-1)
Delta_Tau(Tabu(i,j),Tabu(i,j+1))=Delta_Tau(Tabu(i,j),Tabu(i,j+1))+Q/L(i
);
end
Delta_Tau(Tabu(i,n),Tabu(i,1))=Delta_Tau(Tabu(i,n),Tabu(i,1))+Q/L(i);
end
%禁忌表清零
Tau=(1-Rho).*Tau+Delta_Tau;
Tabu=zeros(m,n);
```

```
end
Pos=find(L_best==min(L_best));
Shortest_Route=R_best(Pos(1),:)
Shortest_Length=L_best(Pos(1))
subplot(1,2,2)   %绘制图形
plot(L_best)
```

附录 5

```
clc,clear
d=xlsread('juli.xls');
w=zeros(100)
for i=1:99
    for j=i+1:100
        if d(i,j)<=1500
            w(i,j)=1.5*d(i,j);
        end
        if d(i,j)>1500
            w(i,j)=1700+(d(i,j)-1000)*1.1;
        end
    end
end
w=w+w'
xlswrite('feiyong',w,'sheet1');

function main
clc,clear
load mydata
ug=xlsread('feiyong')
[i,j,y]=find(ug);
a=sparse(i,j,y);
a=tril(a);
L=size(ug,1);
c1=[1 2:99 100];
[circle,long]=modifycircle(c1,L);
c2=[1 100 2:99];%改变初始圈，该算法的最后一个顶点不动
[circle2,long2]=modifycircle(c2,L);
if long2<long
long=long2;
circle=circle2;
end
circle,long
x=[x0(circle) x0(1)]
```

```
y=[y0(circle) y0(1)]
scatter(x,y,'b','*')
hold on
plot(x,y)
```