

测试规格说明书

目录

1. 测试环境搭建	1
1.1 硬件环境	1
1.2 软件环境	1
2. 测试数据准备	2
2.1 地图数据	2
2.2 两点之间的最短距离.....	4
2.3 最短路径测试值.....	9
2.4 最短时间测试值.....	13
3. 测试方法及结果	16
3.1 兼容性测试	16
3.2 UI 测试.....	18
3.3 随机测试	19
3.4 概率统计	20
4. 测试评估	23
4.1 存在缺陷	23
4.2 分析与改进	24
5. 拟定使用说明	25

1.测试环境搭建

1.1 硬件环境

(1) PC 机 1

表 1-1-1 PC 机 1

名称	信息
制造商	Lenovo
型号	Lenovo Win7 PC
处理器	Intel(R) Core(TM) i7-2670QM CPU @ 2.20Hz 2.19Hz
内存	8.00GB
硬盘大小	320GB
系统类型	64 位操作系统

(2) PC 机 2

表 1-1-2 PC 机 2

名称	信息
制造商	DEEPIN
型号	Deepin233Y
处理器	Pentium(R) Dual-Core CPU E5200 @ 2.50GHz 2.50GHz
内存	1.99GB
硬盘大小	40GB
系统类型	32 位操作系统

(3) PC 机 3

表 1-1-3 PC 机 3

名称	信息
制造商	ACER
型号	Aspire4750
处理器	Intel(R) Core(TM) i5-2410M CPU @ 2.30GHz 2.30GHz
内存	6.00GB
硬盘大小	320GB
系统类型	64 位操作系统

1.2 软件环境

表 1-2-1 软件环境

主机号	操作系统
PC 机 1	Windows7

PC 机 2	Microsoft Windows XP professional 版本 2002
PC 机 3	Windows8

2.测试数据准备

2.1 地图数据

系统的坐标系采用 Window 设备坐标系（客户区）。其原点在对话框的左上角，横轴正方向向右，纵轴正方向向下，单位为像素。具体情况见下图：



(1) 1-49 普通结点（道路与道路的交叉点）、51-71 地址结点（可以标记仓库和客户的结点）、1-75 条道路的标号即 ID 的具体分布如下图：



图 2-1-1 ID 分布

(2) 1-49 普通结点（道路与道路的交叉点）、51-71 地址结点（可以标记仓库和客户的结点）的 Window 设备坐标如下图：

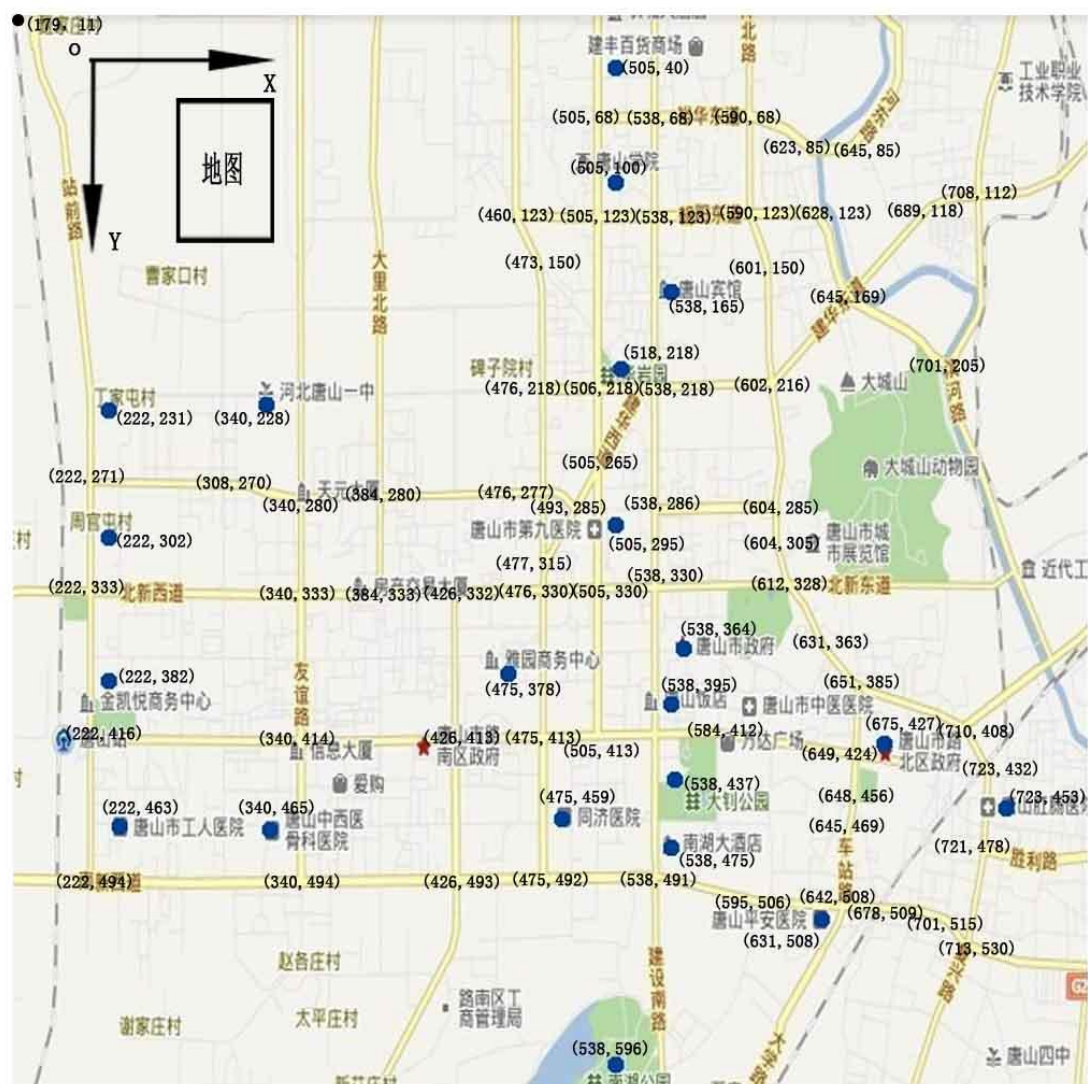
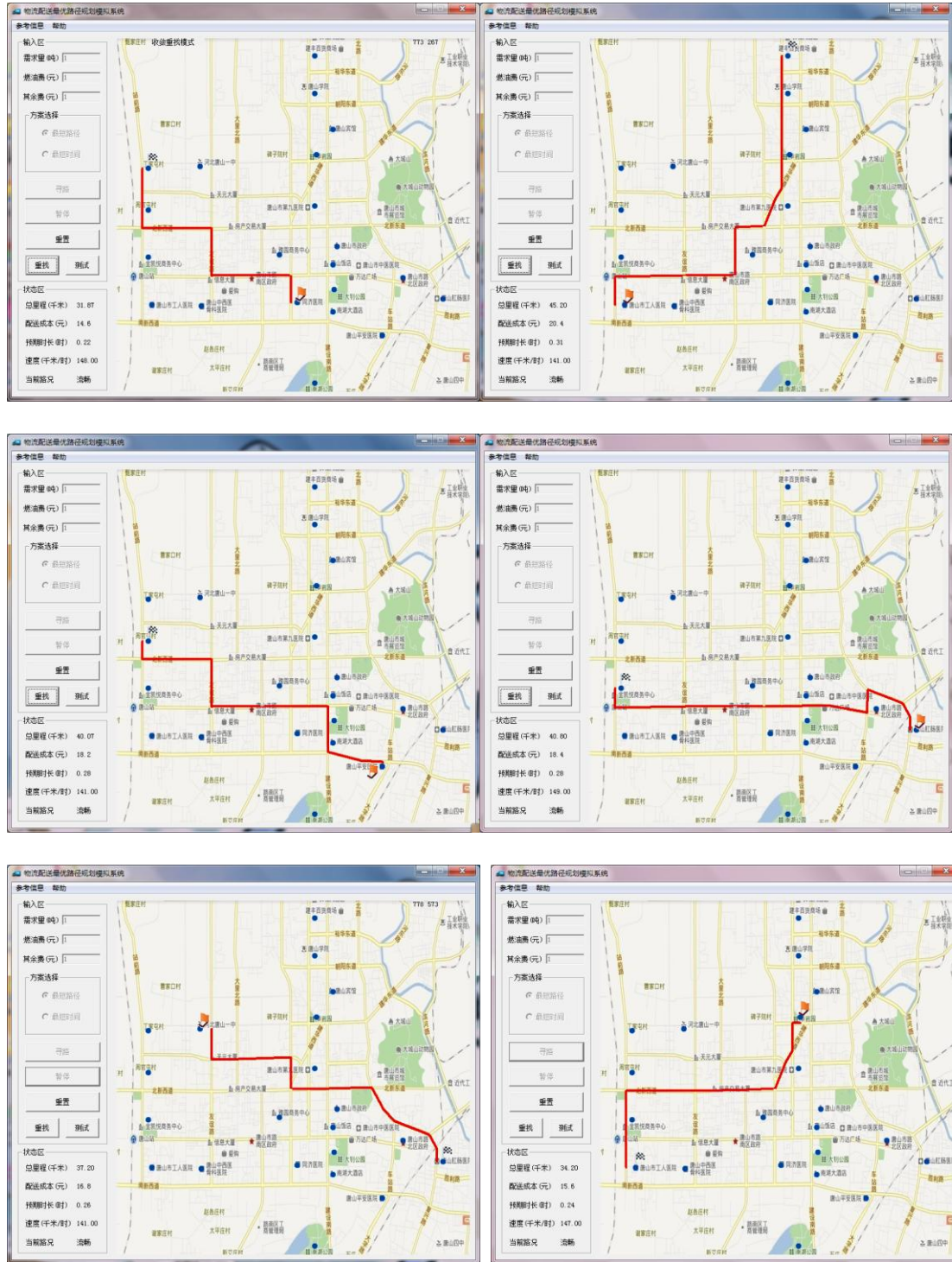
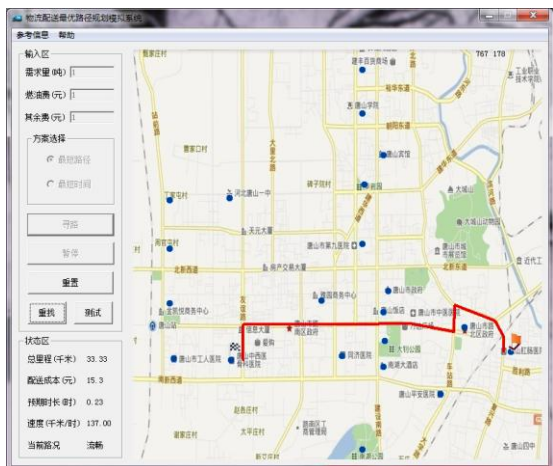
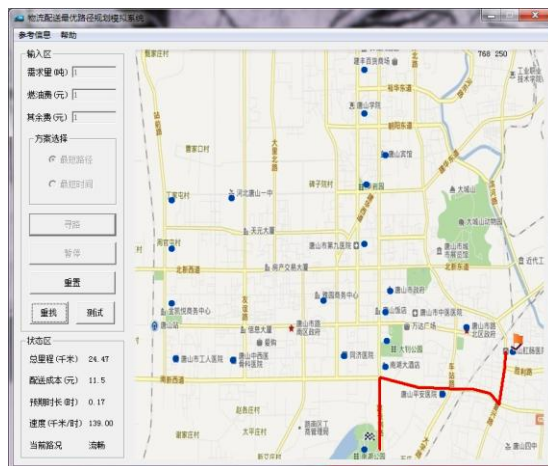
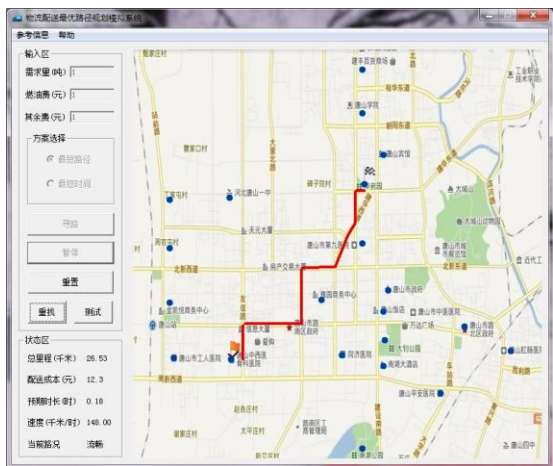
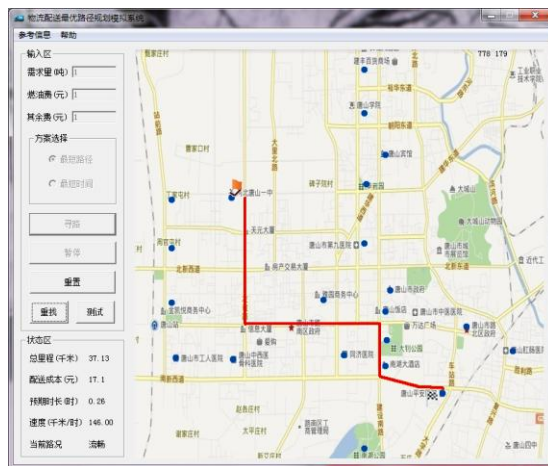
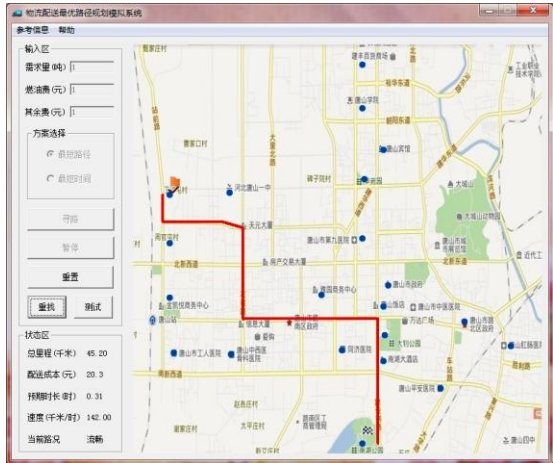
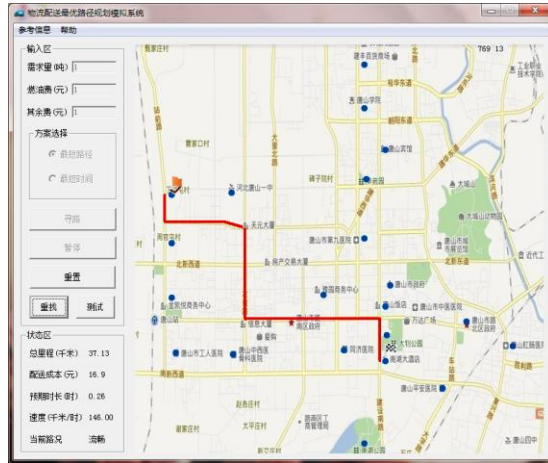


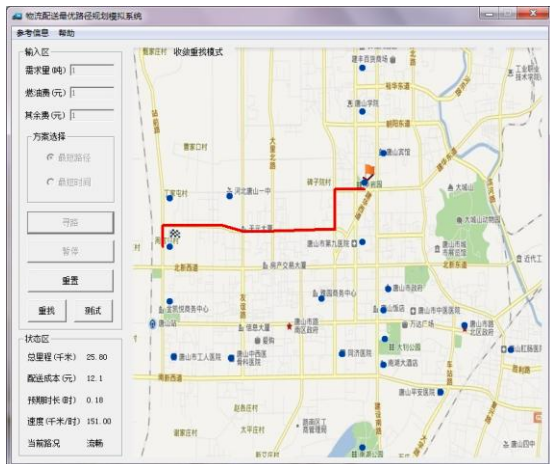
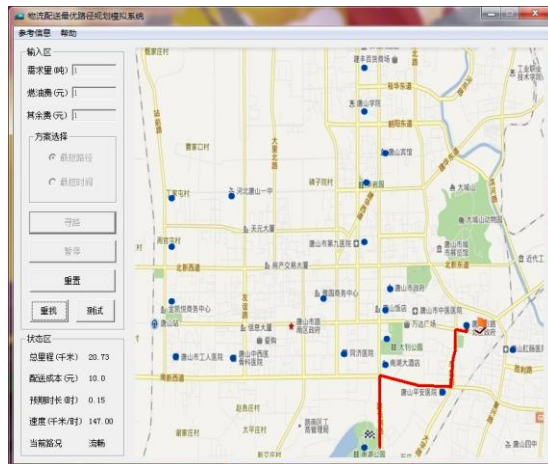
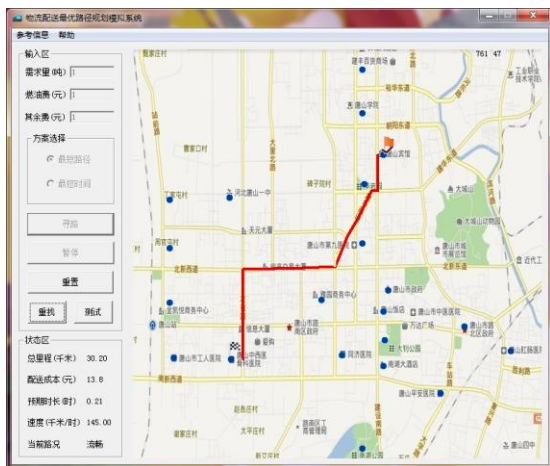
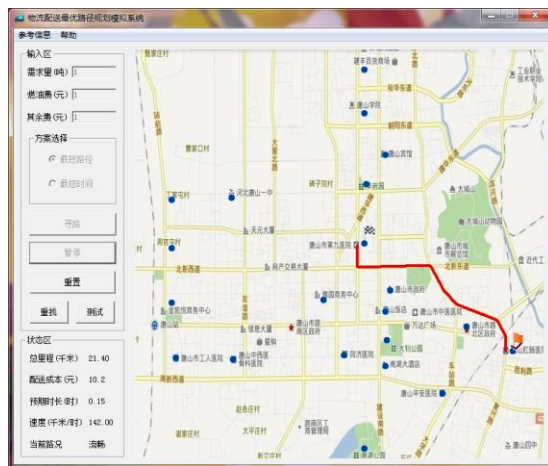
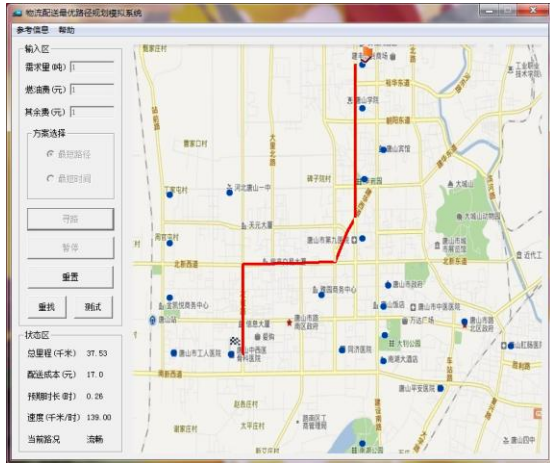
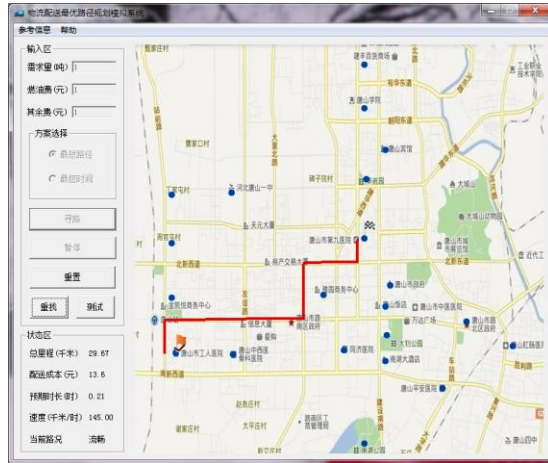
图 2-1-2 71 个结点的 Window 设备坐标

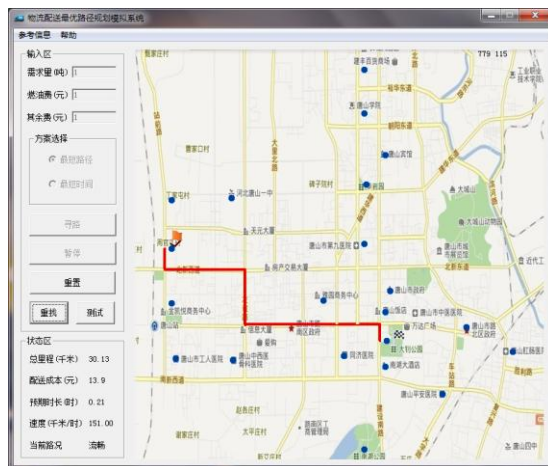
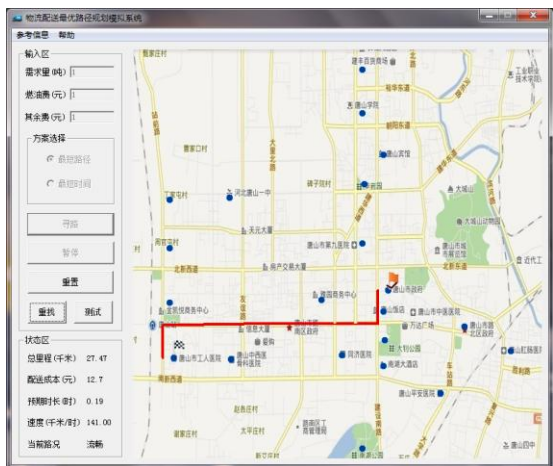
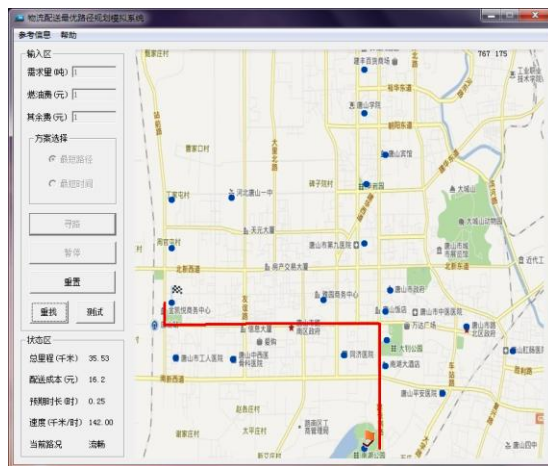
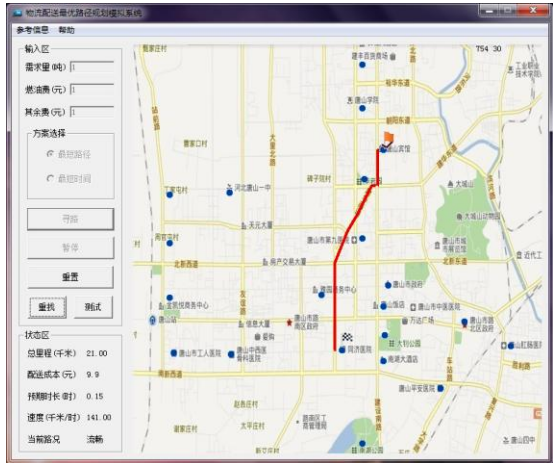
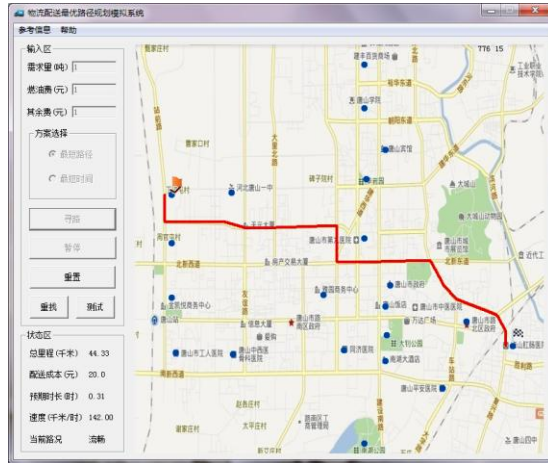
2.2 两点之间的最短距离

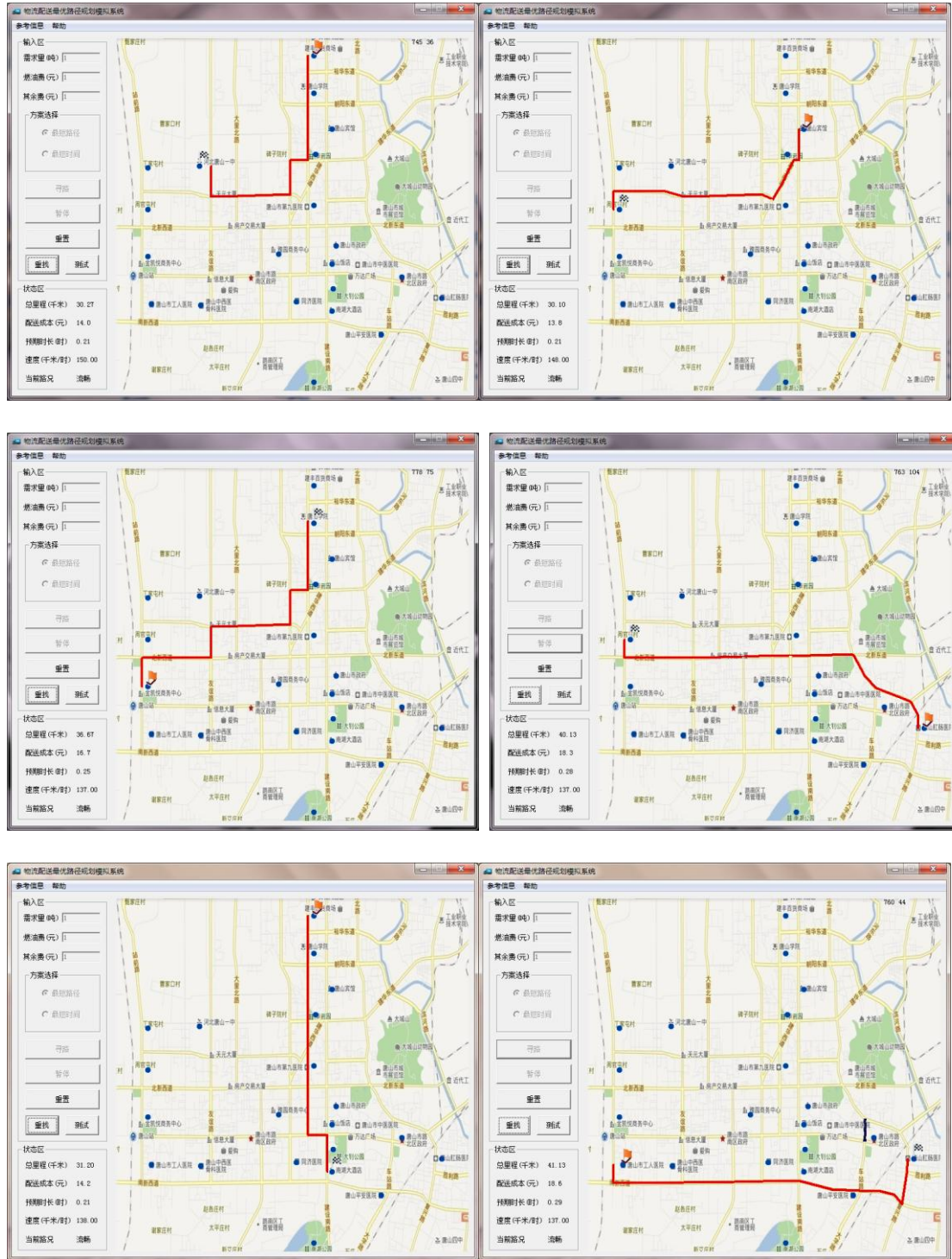
系统的整体思路是先由 A 星算法（启发式算法，非随机算法）得到任意两个客户地址间的最短路径，再利用退火算法求得大于等于 2 个客户之间巡回往返的最短路线。下面选取任意两点进行其最短路线的判定，判定依据通过肉眼对路线的判断。（共 30 组）











2.3 最短路径测试值

通过小范围、中范围、最大范围的地址分布图与系统中的手动测试出来的结果进行比对。（下面只是示例，实际测量需要较大数量组）

表 2-1 小范围测试

地址分布			
状态栏	<div>状态区</div> <div>总里程(千米) 28.60</div> <div>配送成本(元) 13.3</div> <div>预期时长(时) 0.20</div> <div>速度(千米/时) 139.00</div> <div>当前路况 流畅</div>	<div>状态区</div> <div>总里程(千米) 17.17</div> <div>配送成本(元) 9.9</div> <div>预期时长(时) 0.15</div> <div>速度(千米/时) 140.00</div> <div>当前路况 流畅</div>	<div>状态区</div> <div>总里程(千米) 24.37</div> <div>配送成本(元) 11.5</div> <div>预期时长(时) 0.17</div> <div>速度(千米/时) 145.00</div> <div>当前路况 流畅</div>

表 2-2 中范围测试

状态区	
总里程(千米)	99.43
配送成本(元)	48.8
预期时长(时)	0.80
速度(千米/时)	148.00
当前路况	流畅

2.4 最短时间测试值

为了测试动态规划效果的准确性，我组通过一次地图分布如下图

的从“原始路线--动态规划 1 次后的路线--动态规划 2 次后的路线--动态规划 3 次后的路线”的变化中，对比每次的“状态栏”的各项指标，发现其规划效果确实同人工规划效果一致。

表 2-4 动态规划测试

原始最短路线	
动态规划 1 以后的路线	

动态规划2 以后的路线											
动态规划3 以后的路线											
原始状态栏	<div>状态区</div> <table><tr><td>总里程(千米)</td><td>55.03</td></tr><tr><td>配送成本(元)</td><td>26.1</td></tr><tr><td>预期时长(时)</td><td>0.41</td></tr><tr><td>速度(千米/时)</td><td>143.00</td></tr><tr><td>当前路况</td><td>流畅</td></tr></table>	总里程(千米)	55.03	配送成本(元)	26.1	预期时长(时)	0.41	速度(千米/时)	143.00	当前路况	流畅
总里程(千米)	55.03										
配送成本(元)	26.1										
预期时长(时)	0.41										
速度(千米/时)	143.00										
当前路况	流畅										

动态规划1以后的状态栏	<div>状态区</div> <div>总里程(千米) 58.43</div> <div>配送成本(元) 26.1</div> <div>预期时长(时) 0.41</div> <div>速度(千米/时) 147.00</div> <div>当前路况 流畅</div>
动态规划2以后的状态栏	<div>状态区</div> <div>总里程(千米) 62.23</div> <div>配送成本(元) 27.5</div> <div>预期时长(时) 0.43</div> <div>速度(千米/时) 139.00</div> <div>当前路况 流畅</div>
动态规划3以后的状态栏	<div>状态区</div> <div>总里程(千米) 67.70</div> <div>配送成本(元) 29.9</div> <div>预期时长(时) 0.47</div> <div>速度(千米/时) 143.00</div> <div>当前路况 流畅</div>

3.测试方法及结果

3.1 兼容性测试

通过将“物流配送最优路径模拟规划系统.exe”可执行文件传输到上述 PC 机 1、PC 机 2、PC 机 3 上，打开运行进行兼容测试。测试结果如下：

- (1) 在 PC 机 1 上无问题，能够正常运行。
- (2) 在 PC 机 2 上出现如下警告：



图 3-1 缺失文件 MSVCRTD.dll

解决方法：（最简单方法是安装文件夹“兼容补丁”中的“vc6_cn_full.exe”）

①打开光盘中的“兼容补丁”文件夹，解压“msvcrt”文件，根据主机处理器找到对应的 MSVCRTD.dll 文件。

② 直接拷贝该文件到系统目录里：

Windows NT/2000 系统复制到 C:\WINNT\System32 目录下

Windows XP 系统则复制到 C:\Windows\System32 目录下

Windows vista/7 系统也是复制到 C:\WINDOWS\system32 目录下

③然后打开“开始-运行-输入 regsvr32 MSVCRT.DLL”,回车即可解决错误提示

（3）在 PC 机 3 上出现如下警告：



图 3-2 缺失文件 MFC42D.DLL

解决方法：

①解压光盘中的“mfc42d”文件，找到其中对应的 mfc42.dll 文件。

②操作同上述，只不过最后一步输入的是 regsvr32 mfc42.dll。

3.2 UI 测试

通过打开软件，对软件的可见外观和底层与用户交互的部分进行测试，结果如下：

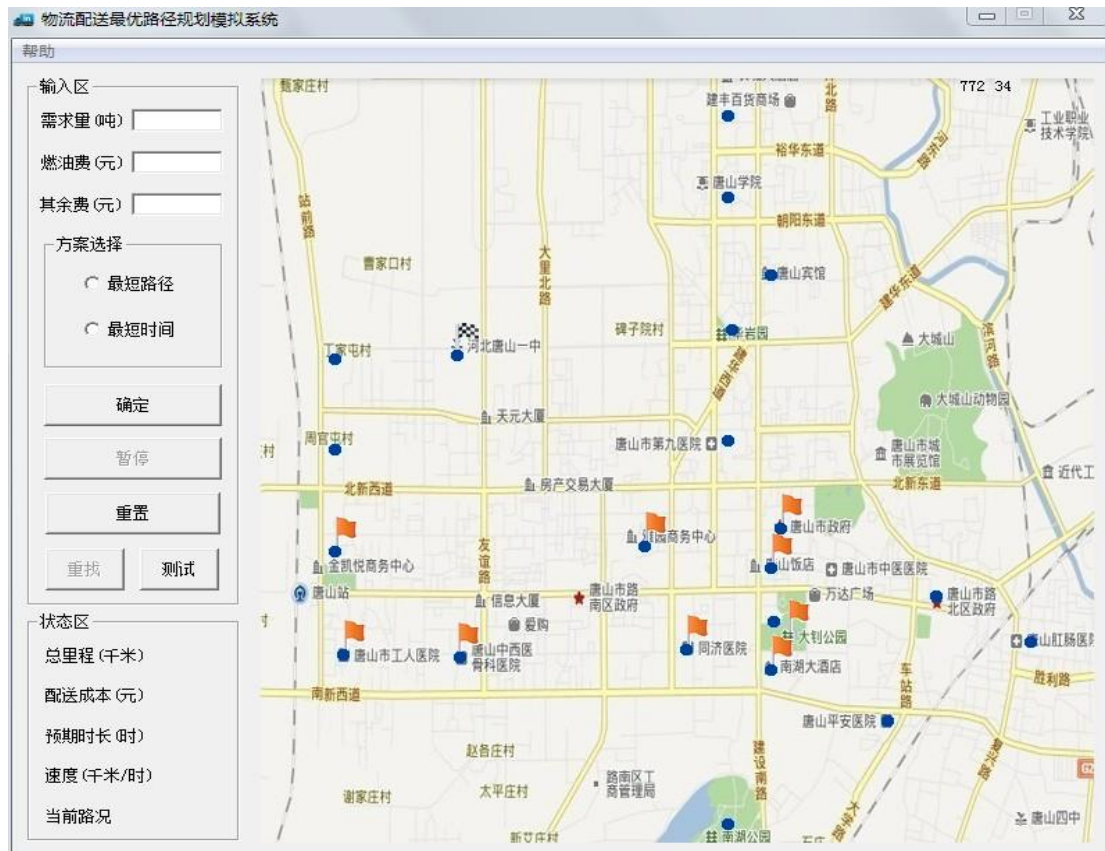


图 3-3 UI 测试界面

基本满足了用户界面的风格清晰明了，文字正确，页面美观，图片衔接到位，操作友好等。例如：所有按钮对齐，字体大小易于观看，出错信息内容清楚易懂，菜单帮助到位，图标寓意系统功能等。具有用户友好性、人性化、易操作性。

此外，按照需求规格书 2 需求分析中所列系统功能性需求、非功能性需求测试，结果显示全部达到预期效果。具有稳定性、持久性、可移植性。

3.3 随机测试

通过测试者对该系统的经验和主观判断对软件进行功能和性能的抽查。而本程序的随机测试应用到的测试数据是 2.1 中结点的坐标和 2.2 中的最短路径测试值。结果如下：

(1) 同手动测试后的状态栏显示进行比对。除了最后一个 99.43 千米路程的路线不同于真正最短的路径 92.20，其余路线的状态栏均一致。(详见视频)

(2) 手动测试同数学演算结果的比对。下面将 28.60 千米路程的这条路线进行手动测试，结果的最短路线图和状态栏如下所示，为了证明该条路线的总里程的准确性，下面用坐标进行数学演算求得最后的真实值。(比例尺 图上每像素：实际千米数=30：1)



图 3-4 手动测试路线及状态栏

结合 2.1 中 (1) (2) 两幅图中的坐标和 ID，和数学中两点间距离

公式 $|AB| = \sqrt{(X1 - X2)^2 + (Y1 - Y2)^2}$ 可以总结出下表：

表 3-1 坐标与距离演算

配送次序	结点 ID	Window 设备坐标	与上一点距离
1	51	(222, 231)	
2	1	(222, 271)	40
3	52	(222, 302)	31
4	2	(222, 333)	31
5	53	(222, 382)	49
6	3	(222, 416)	34
7	54	(222, 463)	47
8	4	(222, 494)	31
9	9	(340, 494)	118
10	56	(340, 465)	29
11	8	(340, 414)	51
12	7	(340, 333)	81
13	6	(340, 280)	53
14	55	(340, 228)	52
15	6	(340, 280)	53
16	5	(308, 270)	33.5
17	1	(222, 271)	86
18	51	(222, 231)	40

设备像素距离

$$=(40+31+31+49+34+47+31+118+29+51+81+53+52+53+33.5+86+40)=859.3$$

59.3（像素）

总里程=设备像素距离/比例尺= $859.3/30=28.60$ （千米）（一致）

（配送成本、预期时长、速度、和路况经检测也同样符合状态栏中的信息）

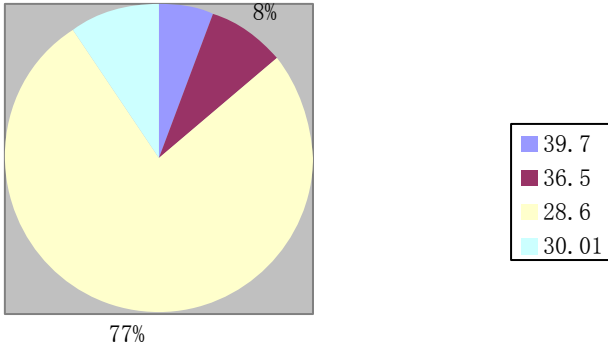
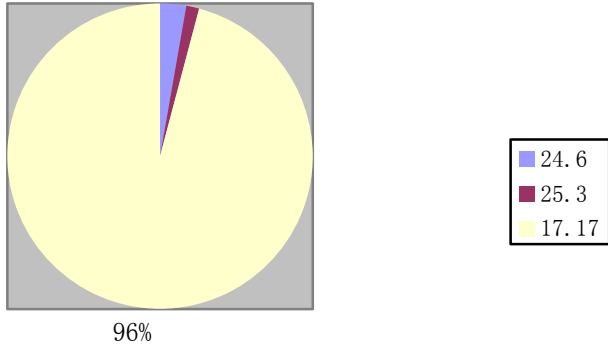
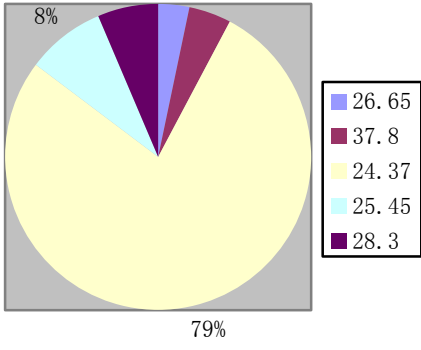
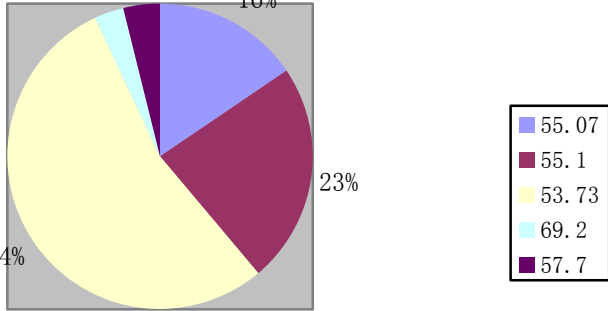
3.4 概率统计

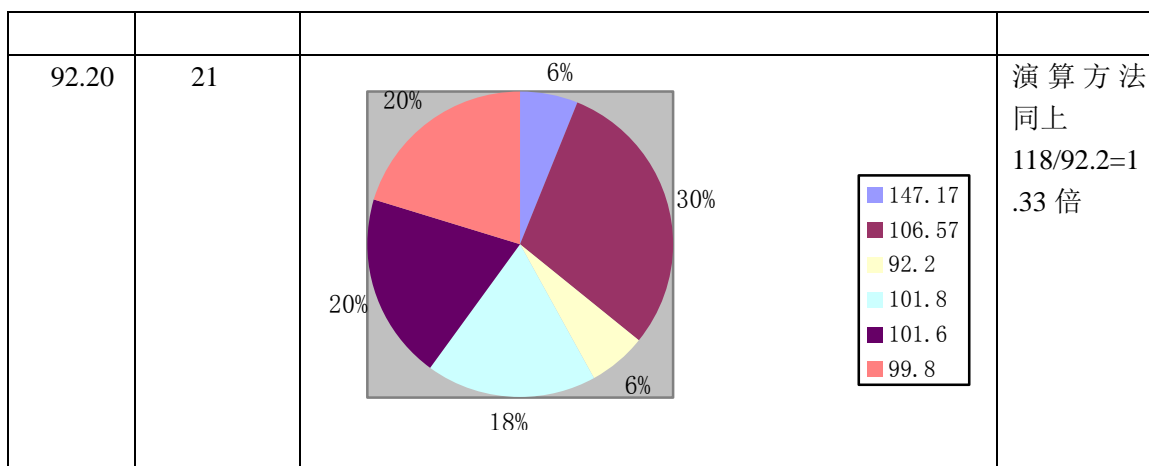
（1）多次点击重找按钮检测上述 5 种情况找出最短路径的概率大小。

下面用圆饼图表示分布（肉色表示实际最短路线的概率；测试次数均

为 300 次):

表 3-2 不同路线随机概率分布

实际最短	地址数	概率分布圆饼图	平均误差倍数
28.60	6		$\left(30.01 \times 9\% + 39.7 \times 6\% + 36.5 \times 8\% \right) / (9\% + 6\% + 8\%) = 29.79$ $29.79 / 28.6 = 1.04 \text{ 倍}$
17.17	7		$\left(24.6 \times 3\% + 25.3 \times 1\% \right) / (3\% + 1\%) = 18.113$ $18.113 / 17.1 = 1.02 \text{ 倍}$
24.37	6		演算方法同上 $29.33 / 24.37 = 1.09 \text{ 倍}$
53.73	13		演算方法同上 $56.93 / 53.73 = 1.17 \text{ 倍}$



由圆饼图分析，找到最优解的概率随着地址数目的增多而减小，这里反映出了随机算法的局限性。但我组为尽力解决该问题设置了重找按钮，通过对历史最优解的淘汰方法避免随机性。

(2) 数学处理：期望=Σ概率×数据

表 3-3 期望统计

实际最短	演算过程	期望
28.60	$0.77*28.6+0.09*30.01+0.08*36.5+0.06*39.7$	E1=30.01
17.17	$0.96*17.17+0.03*24.6+0.01*25.3$	E2=17.47
24.37	$0.79*24.37+0.08*25.45+0.06*28.3+0.04*37.8+0.03*26.65$	E3=25.29
53.73	$0.54*53.73+0.23*55.1+0.16*55.07+0.04*57.7+0.03*69.2$	E4=54.88
92.20	$0.3*106.57+0.2*99.8+0.2*101.6+0.18*101.8++0.06*147.17+0.06*92.2$	E5=104.9

①由以上数据可以求得该系统找到最优路径的概率：

$$P=\sum \text{各组找到最优路径的概率}/5$$

$$=(0.77+0.96+0.79+0.54+0.3)/5=3.36/5=67.2\%$$

②由以上数据还可以画出实际值和期望值的折线图进行比对：

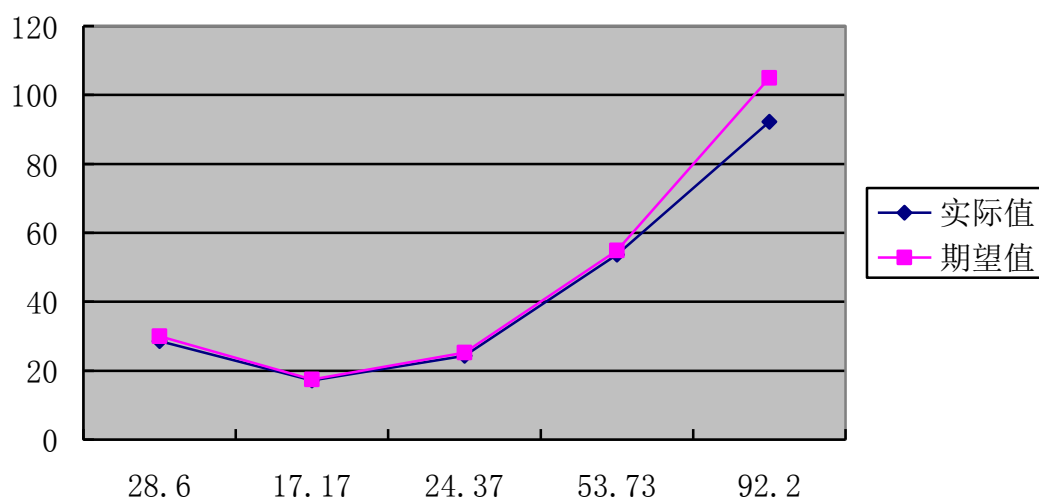


图 3-5 实际值与期望值的折线图

由数据分析得到找到最优解的平均概率为 67.2%，其概率较大。

由折线图分析得到实际值与期望值相差不大，在误差范围之内，基本达到了题目要求。

4.测试评估

4.1 存在缺陷

我组通过利用以上各种测试方法得到的结果，得到了系统存在的如下缺陷：

（1）随机算法的局限性

改进模拟退火算法与遗传算法本身是随机算法，不一定每一次都能找到最优解即最短路径。而系统目前采用“重找”按钮模拟生物进化的特点进行筛选更短的路线。（具体请见测试文档）

（2）与真实数据的差距

模拟软件中的数据和实际生活中的真实值存在一定的差距。

(3) 兼容性问题

软件在不同硬件配置、操作系统的环境下的兼容性存在问题

(4) 调度多辆车参与配送

未能实现根据需求吨数分配车辆数目进行配送。

4.2 分析与改进

针对以上 4 个问题，我组对他们分别做出了分析与改进方案：

(1) 之所以还是存在随机性较强的问题，还是由于数组顺序的变异过程（客户先后次序）存在缺陷。解决的办法是通过对该系统大量的数据测试做出概率分布图，进行离散数学分析，拟定更好的变异方案。

(2) 解决此问题需要进一步深入了解物流配送的流程，对于其中更细的指标加深印象才能估算出更准确的权值。比如：每小时的时间损失费，需要通过对该物流公司配送服务水平与利润收益之间的关系进行统计分析才能得出该费用权值。

(3) 兼容性问题需要经过更多台不同版本型号的 PC 机测试才能进一步提供补丁，解决兼容问题。

(4) 解决调度多辆车进行配送问题可以转换为求一辆汽车的 TSP 完全问题上扩大成了多辆车的 TSP 问题，即 MTSP 问题。且当客户的需求不仅仅是被访问，而是有一定容积和重量的商品的装载和卸载，涉及到不同种类和型号或不同载重量车辆的调度策略时，MTSP 问题转换为 VRP 问题。我组通过查阅资料发现，可以把节约里程法同该系统本身的算法相结合解决。

5.拟定使用说明

兼容问题：具体请见 3.1 兼容性测试。

操作说明及注意事项：

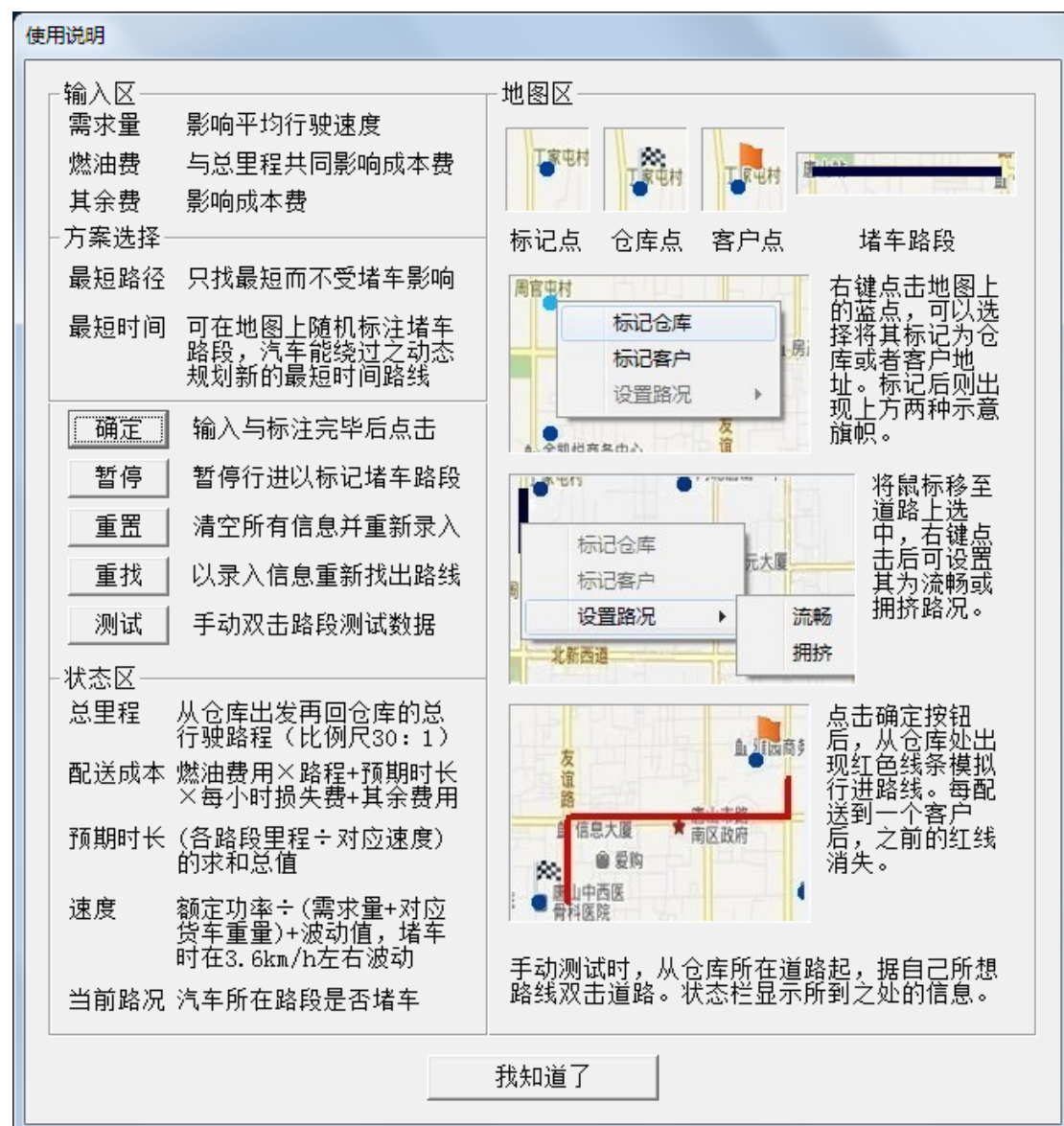


图 5-1 操作说明（上一页）



图 5-1 操作说明（下一页）

如图所示，各项指标的具体含义及其计算公式都有描述，更多详细信息请参见系统设计说明书。另外，需要强调有两点：

（1）按确定和测试按钮前都必须正确输入、正确标注，否则将弹出不合法对话框警告；测试按钮可以在确定按钮按下（即找到一条路线后）后的任意时刻按下；重置按钮可以在任意时刻按下；重找按钮必须在确定按钮按下后才能按下。

(2) 用户在测试时双击不应过于频繁，且双击的位置应该是朝着下次将要到达的路段或是下下次将要到达的路段。如果已经到达一条路段的一端，用户继续双击本路段，那么路线会向着该路段的另一端前进，这表明用户想要测试的路线是迂回路线。另外，手动双击测试时由于 MFC 消息机制的优先性问题可能存在双击不敏感问题，但并无大碍。

(3) 对于汽车行进中的路段无法设置堵车事件（即有红线存在的道路）。当配送到达一个客户地址后，红线消失，此时方可标注。

(4) 在汽车模拟行进的过程中，用户可以不按暂停按钮而标注堵车路段，但系统因有一个计算的时差而可能会暂停些许时间。因此，需要标注时用户最好点击暂停按钮后进行标注，完成后，再按启动按钮。