**GPS定位实验报告**

邱能21307130310

**一、PJ介绍：**

**任务：**给定高采样的轨迹序列（采样频率在10s~15s间）和路网的空间位置和拓扑结构，求得每个轨迹点实际所落在的路段号。

**准确率：**答案返回的路段序列记为，官方的结果序列为. 那么准确率为

**评分方式：**本次PJ主要考察匹配的效率。在达到平均准确率为的前提下，速度越快的程序分数越高。（暂定，即如果平均准确率没有达到，则可能无法得到分数）

**二、重要内容的实现：**

本次PJ使用Hidden Markov map matching through noise and sparseness论文中所给的隐马克洛夫模型方法实现，重要部分在于计算点到一个路段的最近点，路网上两点间的距离，已经状态转移等。分别就重要部分的实现进行阐释。

1. **计算轨迹点p到道路r上的最近点**

道路r可以看做是由一系列相连的线段组成的折线，我们可以对每一个线段求轨迹点到线段上的最近点，再保留所有线段最近点上最近的一个。在求到线段上最近点时，主要思想是利用数学上的公式进行求解。

我们设线段两端为（,）,(,)，轨迹点p（,）。首先考虑特殊情况，即或时，线段为平行于坐标轴的直线，我们过p点做线段所在直线的垂线即可。时，最近点q(,max(,))（）或(,min(,))（）或（,）（）;时同理。

当且时，我们设线段所在直线的方程为，其中, ，所以通过p点到的直线方程为,将两式联立

解得，我们比较该点是否在线段上，如果在就取该点值，如果不在取线段离最近的端点值，或或,。

如此操作后我们得到了p到线段上最近的点q，接下来我们根据p,q之间的距离选择所有线段上距离p最近的点q即可。

1. **计算路网上两点间的距离**

设表示轨迹点在路上的最近点，要计算的是点到点之间的距离dis[i][j]，我们利用BFS（广度优先搜索）的方法搜索在起始线段周围的路段。

我们利用trans[]队列来储存BFS的遍历顺序，初始时将路段的终点与到该终点的路径长度作为结构体加入trans中，每次trans头部元素intersection（路口）出列，考虑以intersection作为起始点的所有路段，将这些路段的终点以及到达时的路径长度作为一个结构体都压入trans队列中。如果遍历深度大于3时，我们通常认为不会在短时间内经过3个及以上的路口，因此可以不必考虑这些情况，在算法中就是遍历深度大于3的点不必考虑。在考虑以intersection作为起始点的路段时，我们同时遍历所有可能匹配的路段,如果两者相同，我们就计算此时点到点的距离,若dis[i][j]之前不存在或,则更新。

这里要注意一个细节，即计算与的距离时，由于都是在同一条路段上，我们计算距离时应该用计算，而不是在BFS算法中的先从出去，再从返回的回路距离。但如果时，因为道路是单行道，不能掉头，所以我们可以假设是通过其他路段绕了一圈回来，此时依旧适用于BFS算法。

以上的算法可以计算出从点出发的所有路径长度dis[i][j]，对于不同的i，我们利用一个for循环全部重复一次上述的计算便可以得到。上述的方法其实和bellman-ford类似，每次进行松弛。

1. **计算测量概率（measured possibility）**

我们假设测量概率和到的距离之间成正态分布关系，由正态分布的概率公式建立模型，到的测量概率,其中。

1. **计算转移概率（transition possibility）**

基于论文的阐释，我们可以发现在大量真实数据的测量下，与间有着数据关系，其分布与 的值成指数函数关系，具体如下图1所示，基于这个性质我们可以构造出关于路径上距离差的转移概率：

其中,

)

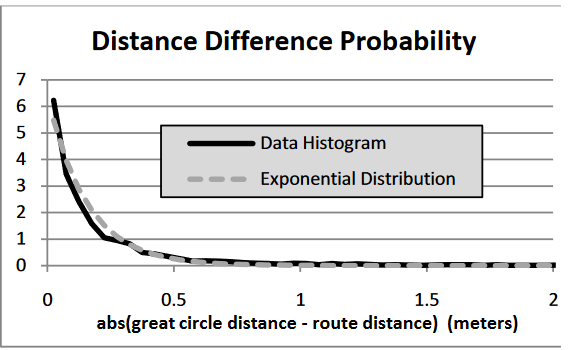
5.**Viterbi算法的动态转移**

图 1

Viterbi算法使用动态规划来快速找到HMM模型网格中的路径，使这条路径上测量概率和转换概率的乘积最大化。设乘积是轨迹点在路段上的投影点对应乘积之和，我们在进行动态转移时只需要考虑前面一个状态的乘积，转移方程为：

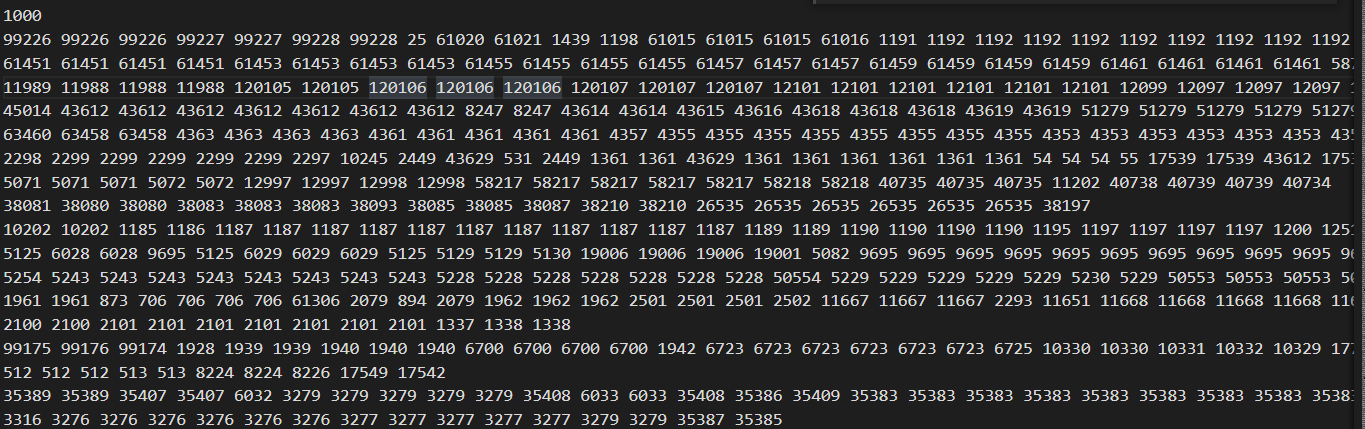
在转移后要注意一点，即所有的概率都是小于1的数字，如果过多的概率相乘，最后得到的概率积是一个很小的数，可能不能够在计算机中存储，于是我们可以在每一层t的转换后给所有乘上一个放大因子，使数据保持在合理的范围内。这里我选择每次乘以,这样最大的就被还原为1，所有步骤中的概率也在1附近的小数部分，便于计算。

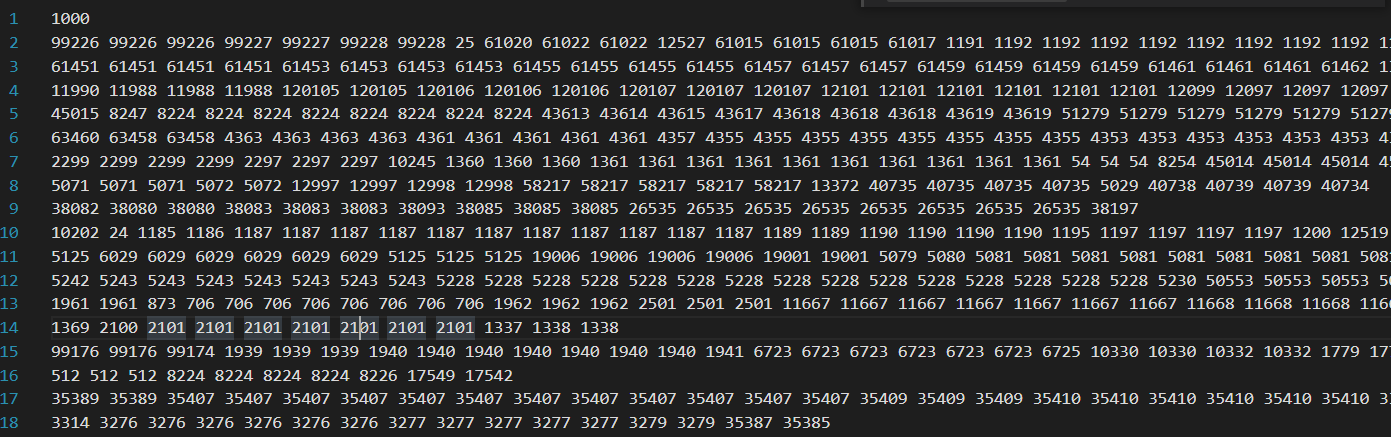
**三、创新点**

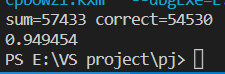
**记忆化存储**

利用map<pair<int,int>,double>实现记忆化存储路口间距离,当第一次计算后，后续使用都不必重新计算。通过实际检验，我们发现这样计算后的时间复杂度大大降低。同时为了更好使用这一方法，当需要两个路口间距离时，我们选取局部做Dijkstra算法求两个路口之间的距离，随后把产生的所有结果都给压入map中。

**四、实验结果与分析**







程序最后的准确度接近于95%，通过打印出每一时刻的转移概率表格与测量概率表格，发现在发生分歧的点上，几乎都是由于产生了另一条道路，其与给出的标准解答离两个测量点间的距离更近，于是这一条道路就取代了原本的道路，成为了新的分叉点。

但如果使用打印当前最佳点的方法，我们又可以在一定程度上弥补这些遗漏的点，可以在将来试图寻找一种权衡方法使得能够同时考虑到乘积最大的路径和当前乘积最大的点，得到的算法能够在最后有更高的准确度。