December 21, 2011

Course 1.204 Project Report

Lu Lu

December 21, 2011

Abstract

This study is based on the paper *Parsing and Modeling Location Histories (Ramaswamy Hariharan and Kentaro Toyama)*. Using one year's GPS data of a person, the algorithms in this paper are enhanced and implemented to extract stays and locations in the trajectory. The probability models in the paper are established based on the GPS data. A fitness to data test is conducted and different models are compared in terms of fit to displacement distribution, fit to stay duration distribution as well as fit to location frequency and location stay time. By comparing the original GPS data and the trajectory prediction of CTRW, Non-Markovian probability model and Markovian probability model, enhanced models are proposed and proved to be able to better fit the GPS data.

I. Introduction

Location histories are records of entities location in geographical space over an interval of time. They can provide us with great help in understanding individual travel behaviors and human mobility patterns. In the past, location histories have been reconstructed by looking at migrating populations or tracking demographics. These trajectories have a temporal resolution of decades, even centuries and a spatial resolution of kilometers. However, advances in location-aware technologies enable us to record location histories at a dramatically increased resolution nowadays-seconds in temporal and meters in spatial.

With these high quality data, the paper *Parsing and Modeling Location Histories (Ramaswamy Hariharan and Kentaro Toyama)* proposes a number of rigorously defined data structures and algorithms for analyzing location histories, including the definitions of stays and destinations and the algorithms to extract the from trajectory data. Two probability models are established in the paper to model people's travel behavior-Non-Markovian and Markovian. Methodologies that train these models with GPS data, use these models to estimate the relative likelihood of a new location history and generate new trajectories are proposed. Authors of the paper test the models in terms of fit to real data with intuitive methods such as comparing the location based trajectory of a person in several days between the data and model prediction. Conclusions are drawn that Markovian model is better to generate (predict) trajectories but Non-Markovian model is better to estimate the relative likelihood of a new location history. The authors also did some other experiments using their extracted stays and locations from the GPS data.

The following of the report is focusing on the my work based on the paper. In section II, the algorithms of extracting stays and destinations are introduced, enhanced and implemented. Section III compares the existing models that can be used to generate trajectories with a more comprehansive way. In section IV, two enhanced models are proposed and evaluated. The report concludes in section V.

II. Extracting Stays and Destinations

A. Stays

A stay is a single instance of an objective spending some time in one place-time dependent. Identification of stays depend on two scale parameters. *roaming distance* is the spatial scale, which is the maximum distance that an object can stray from a point location to count as a stay. *Stay duration* is the temporal scale, which is the minimum duration an object must stay within roaming distance of a point to qualify as staying at that location. With different spatial and temporal scales, we can get different stays. For example, when the roaming distance is 10m, staying in the office can be viewed as a stay yet going to the bathroom, which is in the same building is not the same stay with the office one. However, when we use 100m as the roaming distance, they become a same stay. It is the same for the stay duration.

The algorithm in the paper to extract stays from a trajectory is shown in Figure**??**. The *Diameter* function computes the greatest distance between any two locations in a set, and the *Medoid* identifies the location in a set that minimizes the maximum distance to every other point in the set (i.e., it is the data point nearest to the center of the point set). This algorithm traverse all the GPS points, identifies a stay as the largest possible set with successive GPS point that satisfies that the time difference between the first point in a set and the last point in a set is larger than Stay Duration and the Diameter of the set is larger than Roaming distance. By using Diameter of the set instead of the geometry center of the set, we can prevent viewing a slow movement as a stay (since in a slow movement, the geometry center is moving slowly and the distance between the new point and the center grows in a very slow speed and may not be able to exceed Roaming Distance within Stay Duration).

This algorithm is very computational ineffective, though. First, when diameter is larger than Roaming Distance, it's not neccessary to go back from j* to i all the time. We can check if the distance between j*-1 and j* is larger than Roaming Distance, if not, we can set i to j*-1, which

Input: raw location history, $\mathbf{R} = \{r_i\}$ **Output:** a set of stays, $\mathbf{S} = \{s_i\}$

Initialize: $i \leftarrow 1$, $\mathbf{S} \leftarrow \emptyset$
while $i < R$
  $j^* \leftarrow \min j$ s.t. $r_j \geq r_i + \Delta t_{dur}$;
  if ($Diameter(\mathbf{R}, i, j^*) > \Delta l_{roam}$)
    $i \leftarrow i + 1$;
  else
  begin
    $j^* \leftarrow \max j$ s.t. $Diameter(\mathbf{R}, i, j) \leq \Delta l_{roam}$;
    $\mathbf{S} \leftarrow \mathbf{S} \cup (Medoid(\mathbf{R}, i, j^*), t_i, t_{j^*})$;
    $i \leftarrow j^* + 1$;
  end
end

Figure 1: Original algorithm to extract stays from a trajectory

can accelarate the process a lot since the calculation of Diameter is very time consuming. Then, the Diameter function in the begin loop can be replaced by a more efficient function, which calculates the distance between the new added point and the the four "corner boundaries" of the existing point set. It yields the same results and prevents the $O(n^2)$ complexity of calculating Diameter.

These modifications greatly enhance the efficiency of stay extraction. Using a set of 10,000 GPS points to test, the original algorithm spends 195.85 seconds to generate all the stays and the enhanced algorithm only spends 5.68 seconds. The enhanced method is 35 times faster and yield exactly the same result.

With a roaming distance of 100m and a stay duration of 10min, Figure **??** shows the result of stays extraction from 6000 GPS points and FIgure **??** shows the result of stay extraction from 200,000 GPS points (one person, one year, from September 2001 to September 2002).

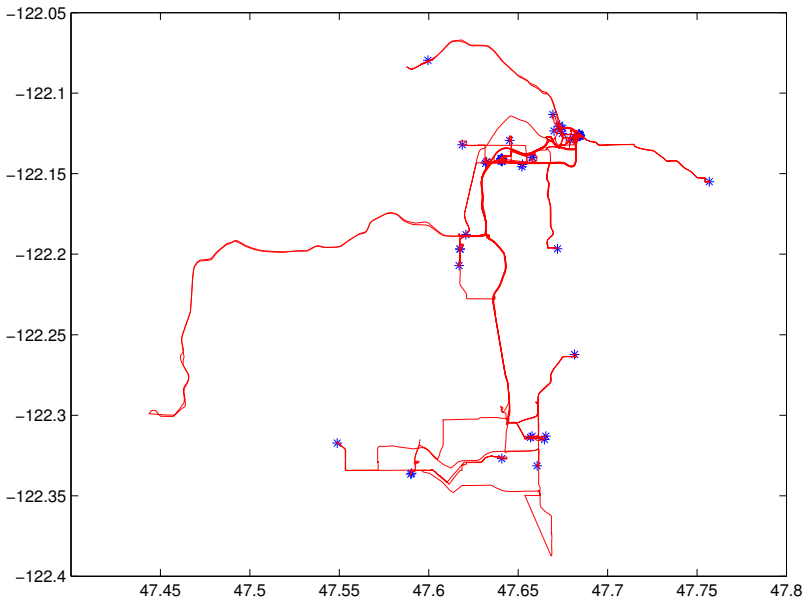


Figure 2: Stays from a trajectory with 6000 GPS points

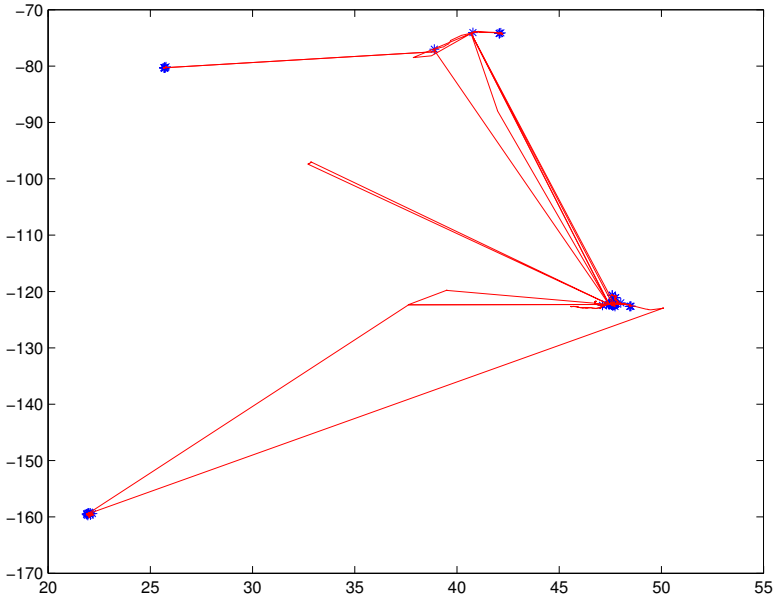


Figure 3: Stays from a trajectory with 200,000 GPS points

B. Destinations

A destination is any place where one or more tracked objects have experienced a stay. Destinations are dependent on geographic scale, but not on temporal scale (i.e., beyond the temporal scales used to identify stays). The scale determines the maximum distance between stays in a destination.

Extracting destinations from stays is a clustering task. As we don't know the number of clusters(destinations) a priori, agglomerative clustering is applied here. Figure **??** shows the algorithm to extract destinations.

Input: a set of point locations, $\mathbf{L} = \{\mathbf{l}_i\}$ **Output:** a set of destinations, $\mathbf{D} = \{d_j\}$

Initialize: $\mathbf{c}_i \leftarrow \mathbf{l}_i$, for $1 \le i \le L$, and $\mathbf{C} = \{\mathbf{c}_i\}$
loop
  $(\mathbf{c}_i, \mathbf{c}_j) \leftarrow FindClosestPair(\mathbf{C})$;
  if $Radius(\mathbf{c}_i, \mathbf{c}_j) \le \Delta l^{dest}$
    $\mathbf{c}_i \leftarrow Merge(\mathbf{c}_i, \mathbf{c}_j)$;
    $\mathbf{C} \leftarrow \mathbf{C} - \mathbf{c}_j$;
  else
    exit
end
foreach $\mathbf{c}_i \in \mathbf{C}$, create destination $d_i = (Medoid(\mathbf{c}_i), \Delta l^{dest})$;

Figure 4: The algorithm to extract destinations from a set of stays

The generation result is shown in Figure **??**. Blue points are the centers of destinations and red points are stays. FIgure **??** shows the person's trajectory based on location IDs. We can easily identify where he live and where he work from the trajectory information.

Using the one year GPS trajectory, which has 216,110 GPS points, 629 stays and 144 destinations are identified.

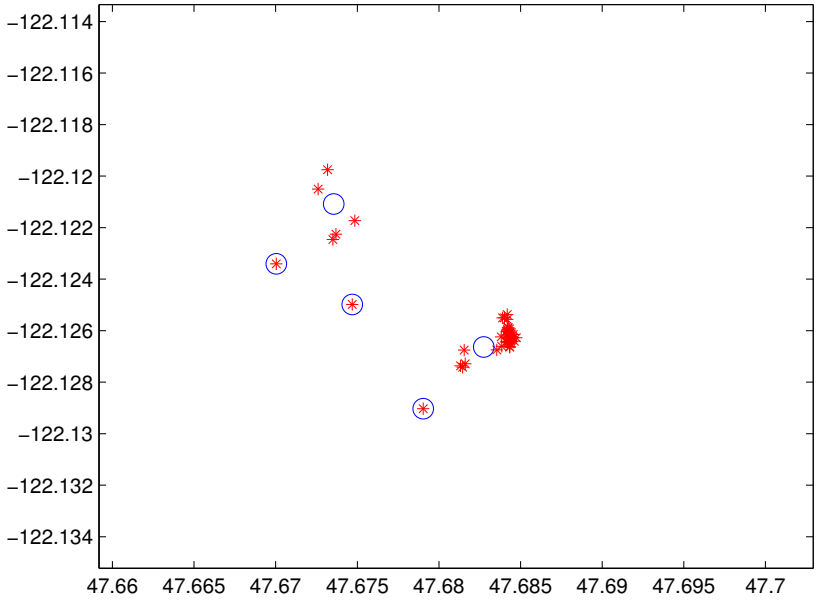


Figure 5: Results of extracting destinations from a set of stays

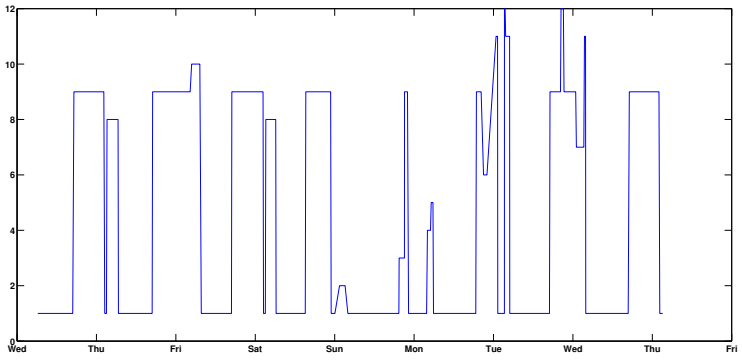


Figure 6: One week's location ID based trajectory of a person

III. Trajectory Generating Models

There are many models that can be used to generate trajectories, such as Eigen Behavior, Continuous Time Random Walk (CTRW) model, and the two models proposed in the paper: Non-Markovian probablity model and Markovian probability model. In order to evaluate these models in term of the ability of reproduce reality, an experiment is conducted to test the fit to real GPS data of the last three models. GPS trajectories are generated with different models and distribution of travel distance, distribution of stay duration, location frequency and location total time of these trajectoris are compared with real GPS data.

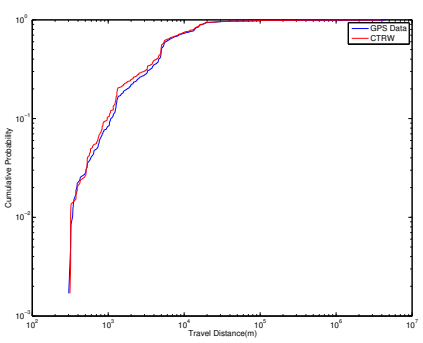
A. Continuous Time Random Walk

According to the paper *Scaling Laws of Human Travel*, CTRW can model the antagonistic interplay between scale-free displacements and waiting times of human mobility with a high accuracy. The distribution of displacement and rest time in the model

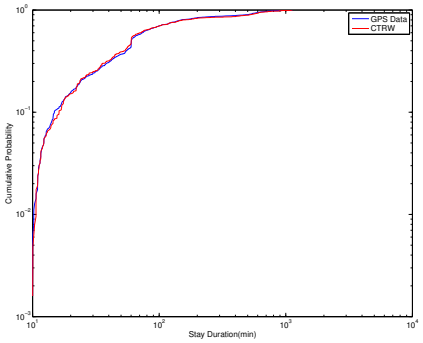
t well the band notes data and the author also proved that the bank note data can fairly reflect peoples travel.

However, when we consider human trajectory, a major dierence betwee the CTRW model and human motion is that individual human returns to several places frequently but the CTRW traces wonder around the whole area randomly. The CTRW failed to reproduce the simple reproducible patterns that are followed by an individual.

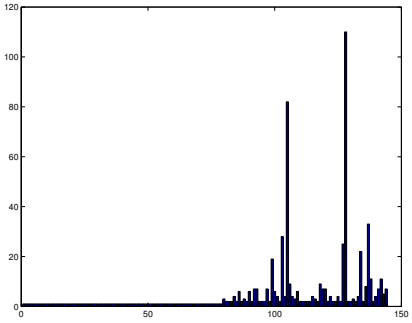
In the CTRW model, we use the distribution of travel distance and stay duration we get from the GPS data as the model input. The fit to GPS data test is shown in Figure **??**.



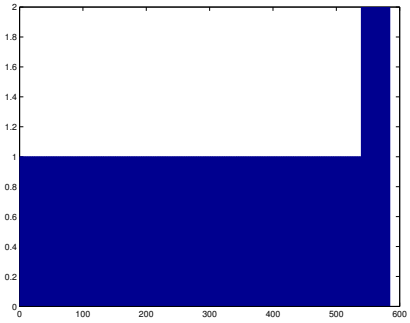
(a) Cumulative Displacement Distribution



(b) Cumulative Stay Duration Distribution



(c) GPS data location distribution



(d) CTRW data location distribution

Figure 7: Continuous Time Random Walk Test

We can see from the figures that CTRW can perfectly match the displacement and stay duration distribution of the GPS data, because the distribution of the GPS data is the model input. In other words, if we have a very good understanding to these two distributions in the real world, CTRW model can generate trajectories perfectly fit these two distributions. However, as we analyzed before, the key weakness of CTRW model is that it fails to capture people's behavior that frequently returning to a relatively small number of locations. Figure **??** and **??** show the poor performance of CTRW in replicating people's location-based trajectories.

B. Non-Markovian Probability Model

For the Non-Markovian Probability Model (Non-Markovian) and Markovian Probability Model (Markovian), we use the concept of *recurring time*, which is the set of all time intervals that represents a regularly recurring interval of time. For example, a recurring time interval might be the set of all times occurring between 18:00 and 19:00, regardless of date. In my model, I divided a day into 144 recurring time intervals and each last 10 minutes. This division is based on the stay duration setting we used in the extracting of stays. There will be no stays shorter than 10 minutes, so it is reasonable to set 10 minute as the minimum time interval.

For both Non-Markovian and Markovian, there are two basic assumptions:

- At the beginning of a given time interval, an object is at exactly one destination.
- During any given time interval, an object makes exactly one transition between destinations. A transition may occur from a destination to itself (a self-transition).

The basic idea for Non-Markovian is that for each recurring time interval, there is a probability that the person is staying at a certain location. Therefore, from the GPS data, we can calculate a matrix that contains the probability that the person at every loction in every recurring time interval. This is called "Training of the model" in the paper.

With this matrix, we can generate a trajectory from a start time interval for any length of time. We update the location of the person every 10 minutes based on the matrix.

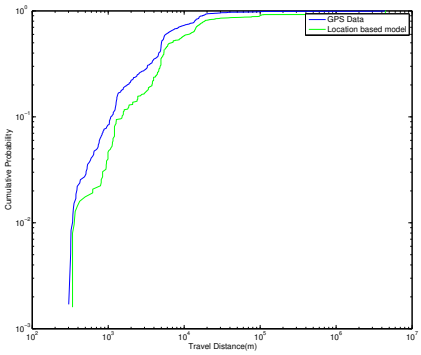
To be comparable to the GPS data, we generate exacly 629 stays with Non-Markovian(and Markovian). The fit to GPS data test is shown in Figure **??**.

We can see from the figures that the Non-Markovian model fits good with the total frequency at each location, which is hardly be acchieved by CTRW. The fit to displacement distribution is OK but the displacements in the trajectory produced by Non-Markovian Model is more likely to be large than those in the GPS data. The reason for this is that the Non-Markovian Model failed to capture the sequential nature of human travel. For instance, people will not go to a resteraunt in San Francisco, then go back to his home in Seattle, and then go to a movie theater in San Francisco. This kind of trajectory can be produced by the Non-Markovian model, which is unreasonable in reality and results in the distribution that long trips are more likely to happen than in reality. For the fit to stay duration distribution, the model has a poor performance. Lack of long duration time also causes the result shown in Figure **??**: although the frequency of being at different locations matches well, the total time at each time has a poor match.

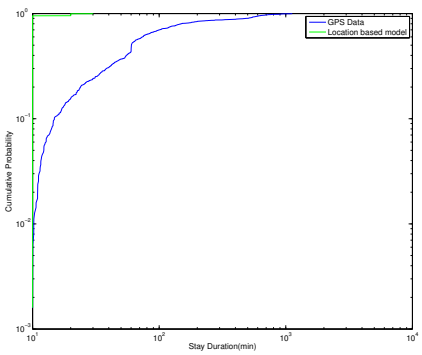
C. Markovian Probability Model

The Markovian model uses destination transition probability matrix instead of destination distribution probability matrix. This matrix contains the probability of the transition from one destination to another(including itself) at every recurring time interval. Training the model with the GPS data, the fit to GPS data test is shown in Figure **??**.

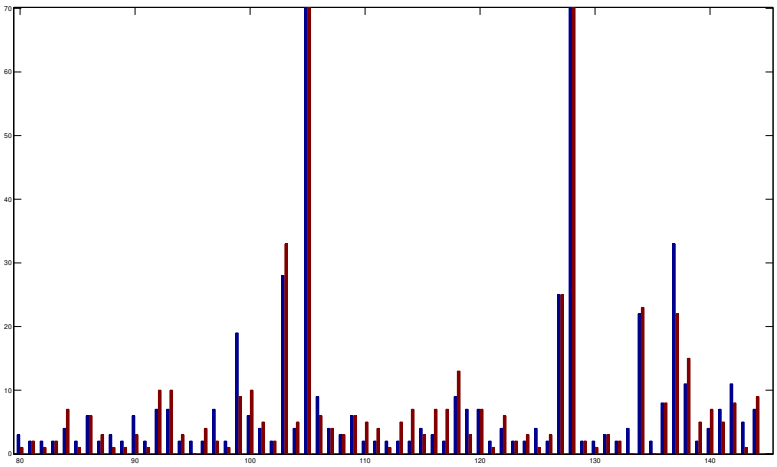
Markovian model is a way to "simulate" people's desision taking process: at this time, at this location, where do I want to go? We can see from the results that the displacement distribution of Markovian trajectory fits better to the GPS data than Non-Markovian trajectory does. However, the stay duration distribution and the location total time are still not consistent with the data and the location frequency distribution is not as good as the Non-Markovian model.



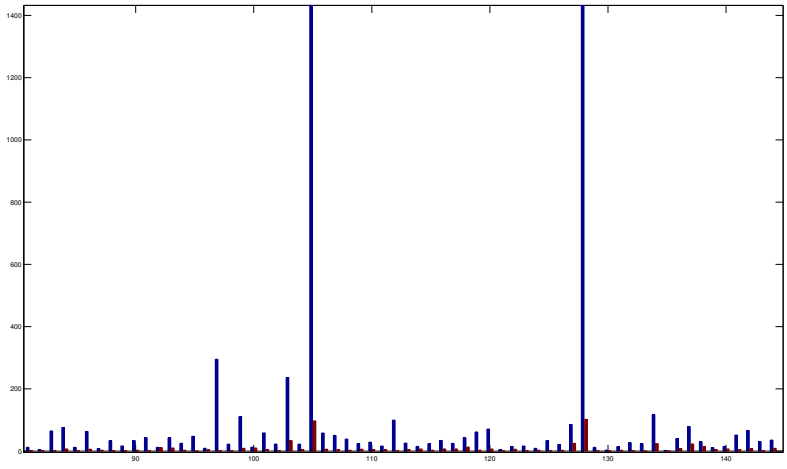
(a) Cumulative Displacement Distribution



(b) Cumulative Stay Duration Distribution

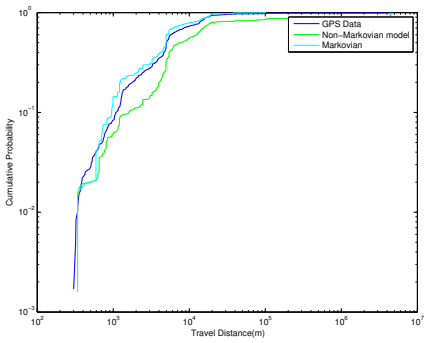


(c) Fit to location frequency

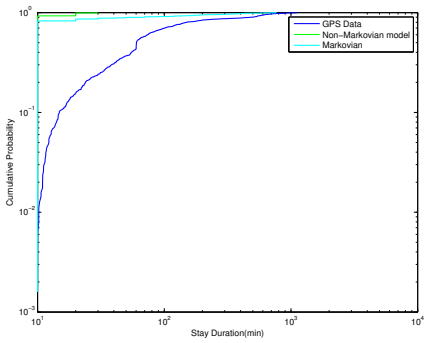


(d) Fit to location total time

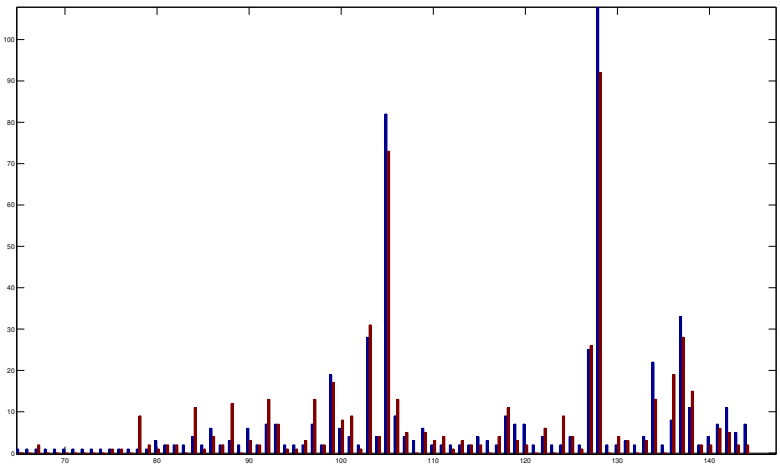
Figure 8: Non-Markovian Probability Model Test



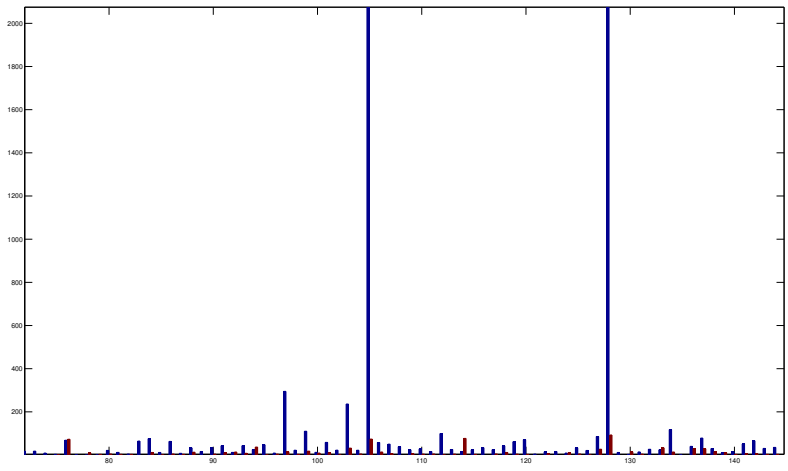
(a) Cumulative Displacement Distribution



(b) Cumulative Stay Duration Distribution



(c) Fit to location frequency



(d) Fit to location total time

Figure 9: Markovian Probability Model Test

The trajectories generated with the Markovian model is more unstable than others. In other words, the statistic results of Markovian trajectories in different run of simulations have larger difference than others. This may be caused by the random nature of the model and probably is a good characteristic to model human behavior.

IV. Enhanced Probability Model

We notice that although the Markovian Model can match the GPS data well in terms of fit to displacement distribution and location frequency distribution, it fails to generate reasonable stay durations and therefore has a poor fit to stay duration as well as location total time distribution. To solve this problem, two enhanced probability models are proposed: Enhanced Non-Markovian Probability Model and Enhanced Markovian Probability Model.

In these two models, instead updating location with a fixed time step, variable time steps are used. We introduced a new matrix that contains the stays duration distribution for each location. The model is also trained with the GPS data.

The Enhanced Non-Markovian Model uses a time-dependent location matrix and a location-dependent duration matrix to generate trajectories and the Enhanced Markovian Model uses a time-dependent location transition matrix and a location-dependent duration matrix to generate trajectories. Their fit to GPS data is shown in Figure **??**.

From the figures we can see that the enhanced method does improve the model performance in reproducing stay duration related distribution. The Non-Markovian model fits better to the time distributions than the Markovian model.

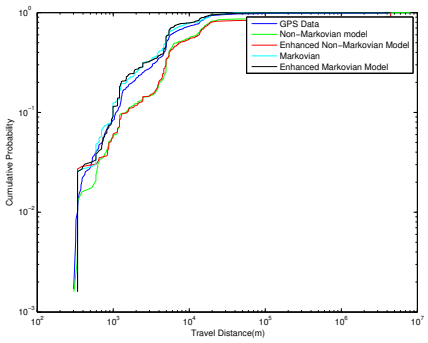
V. Conclusions

In this report, stays and destinations are extracted from one year's GPS data of one subject using the algorithms (or enhanced algorithms) in the paper. Then, different trajectory generation models are tested in terms of fit to GPS data. Their performance is summarized in Table 1.

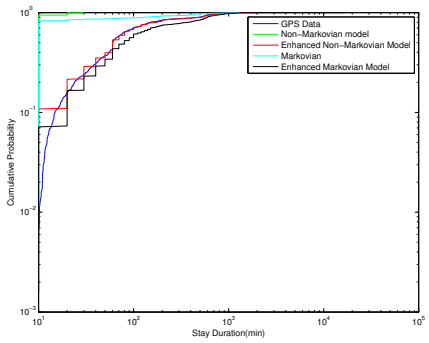
Table 1: Performance Summary of Different Models

	CTRW	NM	EnNM	M	EnM
Displacement Distribution	A+	B	B	A	A
Duration DIstribution	A+	C	A	C	A-
Location Frequency	C	A	A	A-	A-
Location Time	C	C	A	C	A-

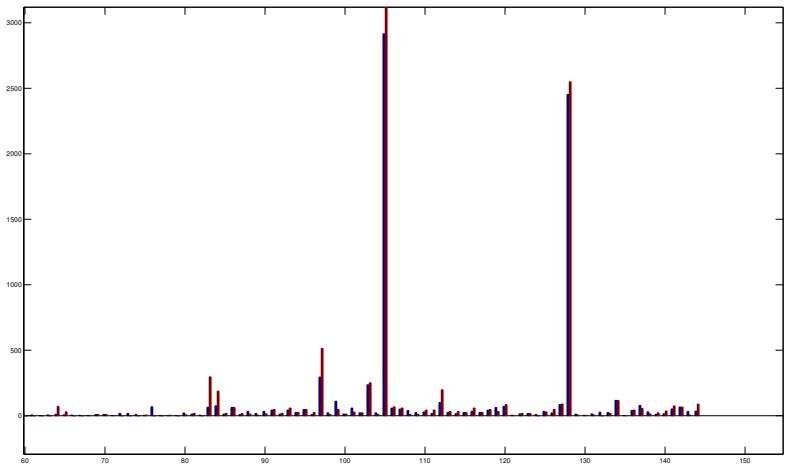
One flaw of this research is that the same GPS data set is used to both train and test the models. This may weaken the validity of the results and should be avoided in future studies. At the same time, although this study uses more comprehensive way to evaluation the models, it is still necessary to have a quantified index to measure the fitness to real data which enable us to have a more clear and persuasive comparison.



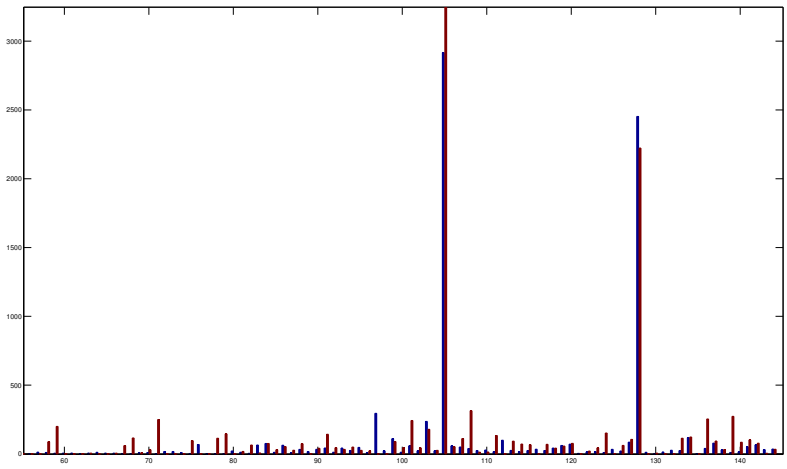
(a) Cumulative Displacement Distribution



(b) Cumulative Stay Duration Distribution



(c) Fit to location time (Enhanced Non-Markovian Model)



(d) Fit to location time (Enhanced Markovian Model)

Figure 10: Enhanced Non-Markovian and Markovian Probability Model Test