A method of probabilistic map distribution of path likelihood

Michel Bierlaire
Jeffrey Newman
Jingmin Chen

STRC 2009 **September 2009**

STRC 2009

A method of probabilistic map distribution of path likelihood

Michel Bierlaire
Transport and Mobility Laboratory
Ecole Polytechnique Fédérale de Lausanne
Lausanne
phone: +41 21 6932537
fax: +41 29 6938060
michel.bierlaire@epfl.ch

Jeffrey Newman
Transport and Mobility Laboratory
Ecole Polytechnique Fédérale de Lausanne
Lausanne
phone: +41 21 6932532
fax: +41 29 6938060
jeffrey.newman@epfl.ch

Jingmin Chen
Transport and Mobility Laboratory
Ecole Polytechnique Fédérale de Lausanne
Lausanne
phone: +41 21 6932532
fax: +41 29 6938060
jingmin.chen@epfl.ch

September 2009

Abstract

A method is proposed to probabilistically map location observations to the underlying network. Instead of generating a single path as the map matching algorithms do, this method aims at calculating a likelihood for each potentially true path to have been the actual path. The result can be used in route choice modeling to avoid biases introduced by a deterministic map matching algorithm. Both spatial and temporal relationships existing in the location data trace and network are taken into account in the method. An algorithm is designed to calculate path probability, starting by defining the measurement for the topological relationship between location observation and network data. Results from the algorithm for a simulated trip are presented to demonstrate the viability of the algorithm.

Keywords

path probability, spatial-temporal, route choice modeling, map matching

1 Introduction

Discrete route choice modeling requires data describing the traveled path. That data can be collected from surveys in many ways. The use of portable GPS devices is becoming more popular because of their accuracy in recording geographical information. However, to serve as an input to the models, the actual path needs to be deduced from discontinuous location data. Map matching algorithms are the traditional way to estimate a single true path from a trace of location data. Significant advancements have been made in the research of map matching algorithms (see, for instance, Quddus *et al.* (2007); White *et al.* (2000)) and, with auxiliary sensors such as dead reckoning, such algorithms have already been used in commercial navigation tools.

However, in route choice modeling, such deterministic map matching is neither desirable nor necessary. Firstly, although the selected algorithm may perform generally well, it is still possible that errors are introduced into the data, because neither the geographical data nor the underlying network data is always accurate. With some systematic errors brought by algorithms (e.g. preferring to match to main roads, or expected path), biases can be introduced in the parameters for discrete choice models. Especially in particularly poor circumstances, map matching will sometimes produce very bad matches if the algorithms parameters are calibrated badly (Marchal *et al.* (2005)). Secondly, route choice modeling frameworks accept a probabilistic representation of the actual path (Bierlaire and Frejinger (2008)). Instead of using a unique true path, a set of potential true paths, along with a likelihood for each proposed path to have been the actual path, is used as the paths observation. However, a detailed method of probabilistic network mapping of location data to paths has not been proposed. Although some existing map matching algorithms employ probabilistic approach (Ochieng *et al.* (2003)), they still ultimately generate a unique result from observations. This paper gives detailed techniques for the theoretical framework proposed in Bierlaire *et al.* (2009), which retains the complete probabilistic representation as an output of the data interpretation process.

In the next section, the input for the algorithm will be introduced, which are location observations and the transportation network. In section 3, the equations will be derived for measuring the topological relationship between a single observation and two kinds of network elements, position and arc, in a probabilistic way. Then, the temporal relationship reflected in observations will be introduced to the network to design an algorithm for calculating the path probability. Finally, an application of the algorithm to a synthetic scenario will be illustrated and some conclusions and future work will be discussed.

2 Input and Notation

2.1 The network

Let $G = (N, A)$ be a network where N is the set of nodes and A the set of arcs. For each node n in N, we know a pair of coordinates $x_n = \{lat, lon\}$, which are the latitude and longitude of n. The application $\ell_a : [0 : 1] \rightarrow \mathbb{R}^2$ describes the trajectory of the physical route corresponding to an arc a. For each physical point x on the arc, there exists a unique ϵ between 0 and 1, representing its position on the arc, such that $x = \ell_a(\epsilon)$. In particular, $\ell_a(0)$ is the coordinate of the up-node, and $\ell_a(1)$ is the coordinate of the down-node of arc a. The speed profile can be described as follows. When ϵ goes uniformly from 0 to 1, the point $\ell_a(\epsilon)$ reflects the trajectory and speed of the traveler. For example, if the arc a is a straight line between node u and node d, and the traveler travels at constant speed we have

$$\ell_a(\epsilon) = (1 - \epsilon) \cdot x_u + \epsilon \cdot x_d. \tag{1}$$

2.2 The location observations

A recording device is carried by the traveler to collect location data. The device makes observations on a variety of direct and indirect location information sources from its sensors, including GPS readings, GSM cell tower information, WLAN base stations, etc. We can generalize each of these location observation data sources to be observations of noisy location information paired with non-noisy (i.e., error-free, at least to the level of relevance for transportation planning) time stamp information. We can merge and arrange these in chronological order to obtain an observed series of points $\check{G} = (\check{g}_k)_{k=1}^{K}$, where $\check{g}_k = (\check{t}_k, \check{x}_k, \check{\sigma}_k^x, \check{v}_k, \check{\sigma}_k^v)$, a tuple containing:

- a time stamp $\check{t}_k$;
- a coordinates $\check{x}_k$, and the standard deviation of the error in the measurement of that coordinates $\check{\sigma}_k^x$;
- a speed measurement $\check{v}_k$, and the standard deviation of the error in the measurement that speed $\check{\sigma}_k^v$.

Henceforth, to provide clarity, any notation referring to raw location information is marked with the ˇ check symbol.

3 Measurement Formulations

3.1 Location measurement errors

We assume that the precision of the location measurements is represented by a random variable in polar coordinates,

$$z(r,\theta) = r \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}, \tag{2}$$

with density $f_z(r,\theta)$. A typical simplification would be to assume that the error is independent of θ. This is incorrect in a urban environment, where the effect of buildings differs with the direction. However, if we accept it, we simply need a distribution on $f_z(r)$ on r. It is typically represented by a Rayleigh distribution.

The probability that a GPS point $\check{x}$ is produced by a device at location $\bar{x}$ is given by

$$\Lambda(\check{x},\bar{x}) = \int_{\theta=0}^{2\pi} \int_{r=\|x-\bar{x}\|}^{+\infty} f_z(r,\theta) dr d\theta. \tag{3}$$

If the distribution is independent of θ, it simplifies to

$$\Lambda(\check{x},\bar{x}) = \Pr(r \geq \|\check{x}-\bar{x}\|) = \int_{r=\|x-\bar{x}\|}^{+\infty} f_z(r) dr. \tag{4}$$

Note that this probability is monotonically decreasing when the distance between the observed $\check{x}$ and the hypothesized true $\bar{x}$ increases.

3.2 Single location measurement

Since $x = \ell_a(\epsilon)$, for any position in the transportation network, the probability density of recording a GPS observation at that position is

$$f_{\boldsymbol{g},\boldsymbol{\varepsilon_a}}(\check{g},\epsilon_a) = f_{\boldsymbol{g},\boldsymbol{x}}(\check{g},\ell_a(\epsilon)) = \Lambda(\check{x},\ell_a(\epsilon)). \tag{5}$$

By Bayes, given that a traveler is on some arc in the transportation network when a location observation $\check{g}$ is recorded, the probability density of the traveler's location can be expressed as

$$f_{\boldsymbol{\epsilon_a}}(\epsilon_a|\check{g},a) = \frac{f_{\boldsymbol{g},\boldsymbol{\varepsilon_a}}(\check{g},\epsilon_a)}{\Pr(\check{g},a)}, \tag{6}$$

where

$$\Pr(\check{g}, a) = \int_{x \in a} f_{\boldsymbol{g},\boldsymbol{x}}(\check{g}, x) dx \tag{7}$$

$$= l_a \cdot \int_0^1 f_{\boldsymbol{g},\boldsymbol{\epsilon_a}}(\check{g}, \epsilon_a) d\epsilon. \tag{8}$$

Then Equation 6 becomes

$$f_{\boldsymbol{\epsilon}}(\epsilon_a | \check{g}, a) = \frac{\Lambda(\check{x}, \ell_a(\epsilon)}{l_a \cdot \int_0^1 \Lambda(\check{x}, \ell_a(\epsilon) d\epsilon} \tag{9}$$

The probability that the traveler is on arc a when the $\check{g}$ is recorded is given by

$$\Pr(a|\check{g}) = \frac{\Pr(\check{g}, a)}{\sum_{b \in A} \Pr(\check{g}, b)} \tag{10}$$

$$= \frac{l_a \cdot \int_0^1 \Lambda(\check{x}, \ell_a(\epsilon)) d\epsilon}{\sum_{b \in A} l_b \cdot \int_0^1 \Lambda(\check{x}, \ell_b(\epsilon)) d\epsilon}. \tag{11}$$

For the vast majority of arcs in a realistic transportation network, this probability approaches zero. Therefore it is reasonable to exclude those arcs with very low probability from our consideration. This results in a suitably small domain of data relevance, D, which includes only those arcs which are at least partially inside a circle circumscribed about the observed point.

4 Path Likelihood Algorithm

4.1 The framework

Along a path, the true location where a GPS observation is recorded is dependent on the true location of the previous observation and the traveler's trajectory in the intervening time. By accounting for the dependency between GPS observations, we derive a measurement equation for calculating the probability of making the observations on a path,

$$\Pr(\check{g}_j, \check{g}_{j-1}, \cdots, \check{g}_1 | p) = \Pr(\check{g}_j | \check{g}_{j-1}, \cdots, \check{g}_1, p) \cdot \Pr(\check{g}_{j-1}, \cdots, \check{g}_1 | p), \tag{12}$$

and then, by Bayes, we can find the likelihood that each possible path is the true path:

$$\Pr(p | \check{g}_1, \ldots, \check{g}_k) = \frac{\Pr(\check{g}_1, \ldots, \check{g}_k | p)}{\sum_{\forall p'} \Pr(\check{g}_1, \ldots, \check{g}_k | p')}. \tag{13}$$

For each GPS observation, since a path consists of connecting arcs, we have

$$\Pr(\check{g}_j|\check{g}_{j-1},\cdots,\check{g}_1,p) = \sum_{a\in(D_j\cap p)} \Pr(\check{g}_j,a|\check{g}_{j-1},\cdots,\check{g}_1,p), \tag{14}$$

$$= \sum_{a\in(D_j\cap p)} l_a \cdot \int_0^1 f_{\boldsymbol{g},\boldsymbol{\epsilon^j}}(\check{g}_j,\epsilon_a)\cdot f_{\boldsymbol{\epsilon^j}}(\epsilon_a|\check{g}_{j-1},\cdots,\check{g}_1,p)d\epsilon_a, \tag{15}$$

where $\boldsymbol{\epsilon^j}$ denotes the random variable of the true position on an arc where $\check{g}_j$ is recorded. $f_{\boldsymbol{\epsilon^j}}(\epsilon_a|\check{g}_{j-1},\cdots,\check{g}_1,p)$ is the probability density function for the distribution of the current state given the previous GPS observations, which we term the "state function". At the first observation, there isn't a previous observation, so

$$\Pr(\check{g}_1|P) = \sum_{a\in(D_j\cap p)} \Pr(\check{g}_1,a). \tag{16}$$

4.2 The state function

The underlying dependency in Equation 15 is actually a result of the traveler's movement from the domain of the previous observation $\check{g}_{j-1}$ to the domain of the current observation $\check{g}_j$. Decomposing the movement into arc transitions will result in the state function becoming

$$f_{\boldsymbol{\epsilon^j}}(\epsilon_a|\check{g}_{j-1},\cdots,\check{g}_1,p) = \sum_{b\in(D_{j-1}\cap p)} f_{\boldsymbol{\epsilon^j}}(\epsilon_a|b,\check{g}_{j-1},\cdots,\check{g}_1,p)\cdot f_{\boldsymbol{a}}(b|\check{g}_{j-1},\cdots,\check{g}_1,p), \tag{17}$$

in which

- the probability $f_{\boldsymbol{a}}(b|\check{g}_{j-1},\cdots,\check{g}_1,p)$ was calculated when for $\check{g}_{j-1}$,

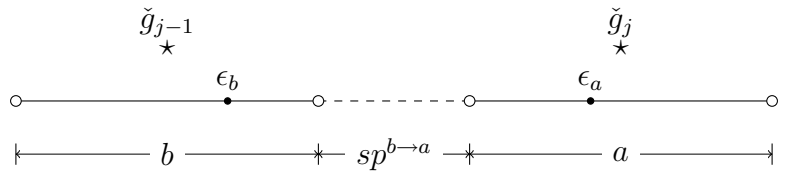
  $$f_{\boldsymbol{a}}(b|\check{g}_{j-1},\cdots,\check{g}_1,p) = \frac{\Pr(\check{g}_{j-1},b|\check{g}_{j-2},\cdots,\check{g}_1,p)}{\Pr(\check{g}_{j-1}|\check{g}_{j-2},\cdots,\check{g}_1,p)}; \tag{18}$$

- the probability $f_{\boldsymbol{\epsilon^j}}(\epsilon_a|b,\check{g}_{j-1},\cdots,\check{g}_1,p)$ represents the likelihood of being at position ϵ_a at the moment when $\check{g}_j$ is recorded, given the trace of GPS observations before $\check{g}_j$. This implies that we know the time $\check{t}_j$ of recording $\check{g}_j$, but we don't have other information in $\check{g}_j$ yet. For simplification, only the previous $\check{g}_{j-1}$ is taken into account, and the probability simplifies to $f_{\boldsymbol{\epsilon^j}}(\epsilon_a|b,\check{g}_{j-1})$. By considering each possible position on arc b, it becomes

  $$f_{\boldsymbol{\epsilon^j}}(\epsilon_a|b,\check{g}_{j-1}) = \int_{\epsilon_b=0}^1 f_{\boldsymbol{\epsilon^j}}(\epsilon_a|\epsilon_b,\check{g}_{j-1},b)\cdot f_{\boldsymbol{\epsilon^{j-1}}}(\epsilon_b|\check{g}_{j-1},b)d\epsilon_b. \tag{19}$$

  Before deriving the equation for position transition probability $f_{\boldsymbol{\epsilon^j}}(\epsilon_a|\epsilon_b,\check{g}_{j-1},b)$ in Equation 19, we have to reconstruct traveler's movement.

Figure 1: Arc transition between adjacent domains



4.3 Spatial-temporal relationship between observations

From Figure 1, we have that the travel time from ϵ_b to ϵ_a is

- if $a \neq b$,

$$\boldsymbol{t}_{b\to a} = (1-\epsilon_b)\cdot \boldsymbol{t}_b + \sum_{c\in sp^{b\to a}} \boldsymbol{t}_c + \boldsymbol{t}_w^{b\to a} + \epsilon_a \cdot \boldsymbol{t}_a = \check{t}_j - \check{t}_{j-1}, \tag{20}$$

  where $sp^{b\to a}$ is the sub-path from the down-node of arc b to the up-node of arc a, t_c is time cost on traveling on its component arc c and t_w is the total waiting time at intersections or other transportation facilities which might cause stops;
- if $a = b$,

$$\boldsymbol{t}_{b\to a} = -\epsilon_b \cdot \boldsymbol{t}_b + \epsilon_a \cdot \boldsymbol{t}_a = \check{t}_j - \check{t}_{j-1}. \tag{21}$$

Now, we have a basic idea about how two adjacent observing positions ϵ_a and ϵ_b are related. And,

$$f_{\boldsymbol{\epsilon}^j}(\epsilon_a|\epsilon_b, \check{g}_{j-1}, b) = f_{\boldsymbol{t}_{b\to a}}(\check{t}_j - \check{t}_{j-1}|\epsilon_a, \epsilon_b, \check{g}_{j-1}, b). \tag{22}$$

In the following part of this section, the random variables in Equation 20 and Equation 21 will be discussed.

4.3.1 Travel time cost on arc a and b

The speed is assumed to be constant during the traveler travels on an arc, so travel time on arc is calculated by

$$t_c = \frac{l_c}{v_c}, \qquad \forall c \in A. \tag{23}$$

This assumption also allows us to take advantage of speed data $\check{v}$ recorded in $\check{g}$ to estimate the arc speed, given the condition that traveler is on an arc. $\check{v}$ is measured with error and its standard deviation $\check{\sigma^v}$ is also given in $\check{g}$. It is convenient and applicable to assume that the speed of traveler is normal distributed with mean $\check{v}$, and standard deviation $\check{\sigma^v}$. Since traveler's true speed lies in a continuous bound between 0 and the maximum capable speed of mean of transport which he/she is using, the speed distribution should be truncated within that bound.

4.3.2 Travel time on $sp^{b\to a}$'s component arcs

An assumption that the speed on arc $c \in sp^{b\to a}$, v_c, equals to v_a or v_b might be too strong to be reasonable. However, if the traveler's traveling pattern is stable, there exists a relationship between observed speeds and unobserved speeds during the time between observations. This relationship is also dependent on the underlying transportation network. In traffic theory, the free flow speed ratio reflects the traffic conditions. The inverse free flow speed ratio is

$$\varpi = \bar{v}/v, \tag{24}$$

in which $\bar{v}$ is the free flow speed or expected speed given in the network data, and v is the actual speed. At each GPS observation, the inverse free flow speed ratio is calculated by

$$\varpi_j = \sum_{a\in D_j} \frac{\Pr(a|\check{g}_j)}{\Pr(\check{g}_j)} \cdot \frac{\bar{v}_a}{\check{v}_j}. \tag{25}$$

We assume that within a certain geographical area and time period (Θ_j is a set of GPS observations which satisfy this condition), the traffic condition is stable to some extent, then normal distribution is used to depict $\boldsymbol{\varpi}$. The estimator for the mean is

$$\bar{\varpi} = \frac{1}{n} \sum_{\check{g}_i\in\Theta_j} \varpi_i \tag{26}$$

And estimator for the variance is,

$$\delta^2_\varpi = \frac{1}{n-1} \sum_{\check{g}_i\in\Theta_j} (\varpi_i - \bar{\varpi})^2. \tag{27}$$

For each $c \in sp^{b\to a}$,

$$\boldsymbol{t}_c = \frac{l_c}{\bar{v}_c} \cdot \boldsymbol{\varpi} \tag{28}$$

follows normal distribution.

4.3.3 Waiting time caused by stops

Due to traffic control devices (lights, stop signs, etc) existing in the transportation network, sometimes travelers are stopped, during the interval time between two location observations. So the possible waiting time should be taken into account if there are intersections between arc b and a. Actually, the meaning of the waiting time is introducing a penalty to those unlikely arc transitions, because if the observation interval is small enough, an abnormal speed profile could be observed. For example, if a device records data in every 10 seconds, there is at least 50% possibility that a stop is observed, if a traveler has been waiting for his/her green light for 5 seconds, 100% if 10 seconds. The incorporation of GPS observations with very low speed is a topic for further research. Within this paper, the distribution of waiting time is assumed to be uniform.

5 Application to A Synthetic Network

To examine the capability of the algorithm described above, tests were conducted on simulated data. In this paper, results from simulations on a small synthetic network are presented. Figure 2 gives the simulated scenario, in which the network is comprised of two parallel horizontal lines, each of which contains 10 arcs with length $94m$ and vertical lines connecting them (north and east directions are scaled differently with ratio $1 : 9.4$). The dashed pattern on the vertical links indicates their length is changeable in various scenarios, so that the performance of the algorithm can be tested at various resolutions. All arcs in the network are bidirectional. In the simulation, a traveler drives a car departing from node o, traveling along the bottom line at a constant speed $40km/h$, arriving at his destination node d. A recording device with him records a location observation in every $10s$. The green solid points are those true locations where the observations are recorded. Errors are introduced to each observation:

- offsets for latitude and longitude are drawn from normal distributions, which both have zero mean and different standard deviations randomly and independently selected from $[0, 30m]$. However, only the root mean square of two standard deviation is recorded in each observation as $\breve{\sigma}_x$. In order to make the simulation close to reality, the latitude's standard deviation is multiplied by 1.5 to simulate a systematic error and the latitude's offset is truncated in a bound $[-13m, 30m]$, while longitude's offset is truncated in $[-20m, 20m]$;
- offset for speed is drawn from normal distribution as well, with zero mean, and a fixed standard deviation $6km/h$.

Figure 2: Simulation Scenario

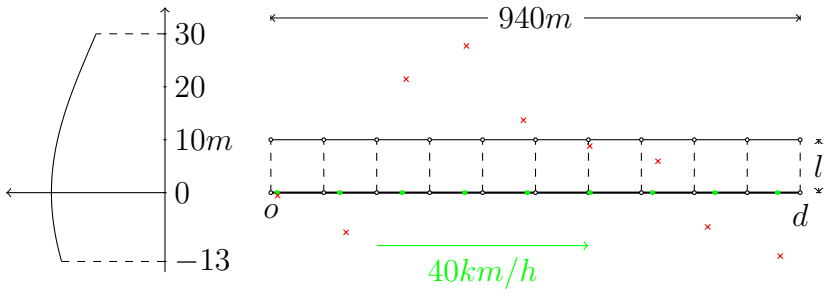
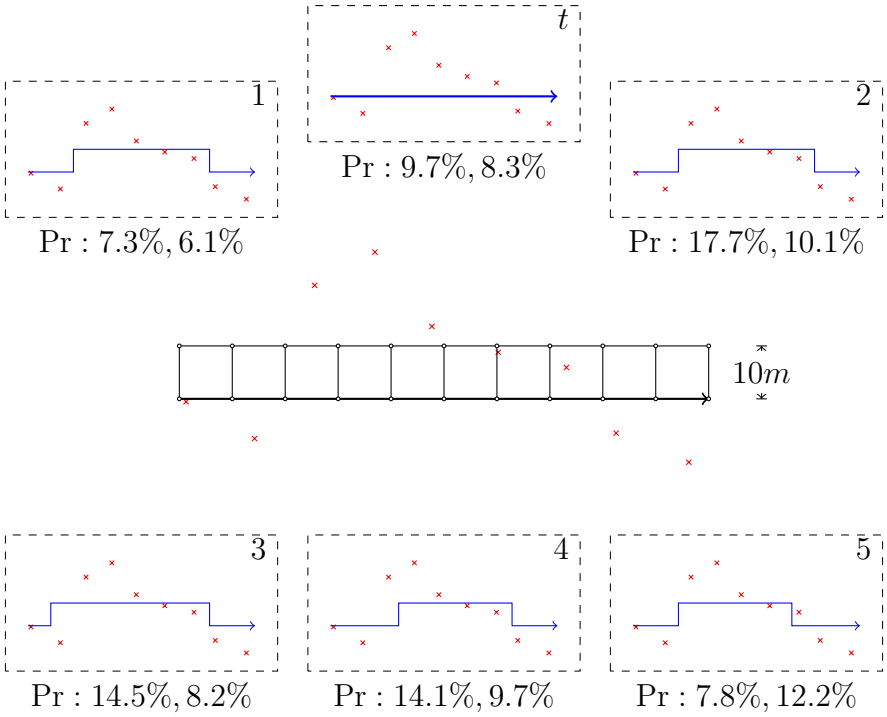


Figure 3: Result with $l = 10m$



Among a large amount of simulated data, a typical scenario is selected for analysis here. The red x symbols in Figure 2 represents the coordinates observed. The coordinates' errors are so random that we can hardly tell which path is the true one intuitively. For comparison purpose, we run the algorithm with l (the distance between the longitudinal paths) being $10m$, $20m$, and $30m$ respectively.

Figure 3 shows the result when l is set to be $10m$. Ten paths are generated but only 6 most probable paths are presented. The two probability values under each path indicate the path probability calculated by different algorithms. The first one is calculated by the method describe above; while the second one is calculated by similar method but without state function,

Figure 4: Result with $l = 15m$

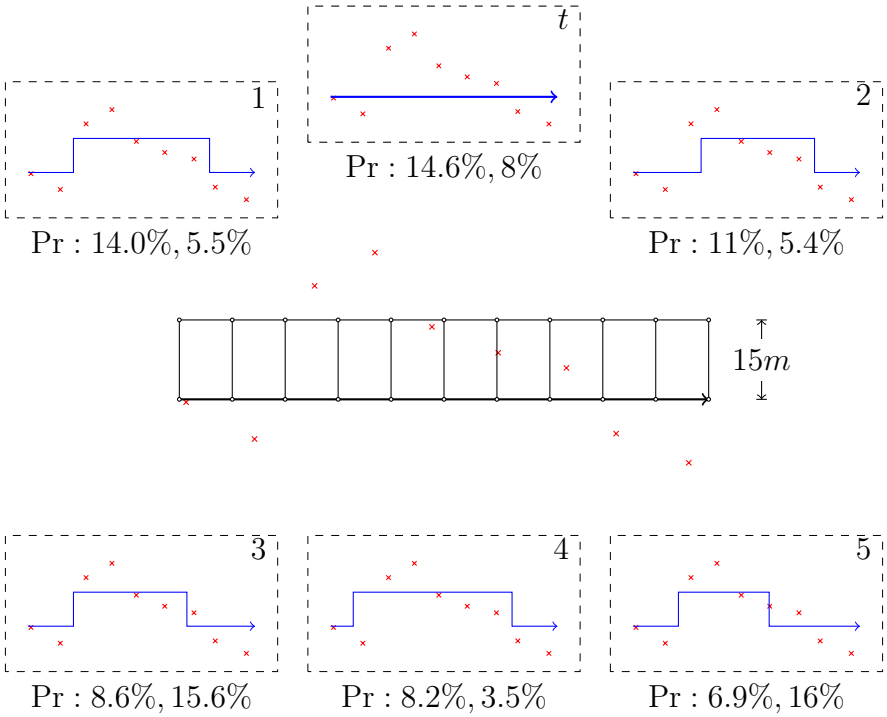
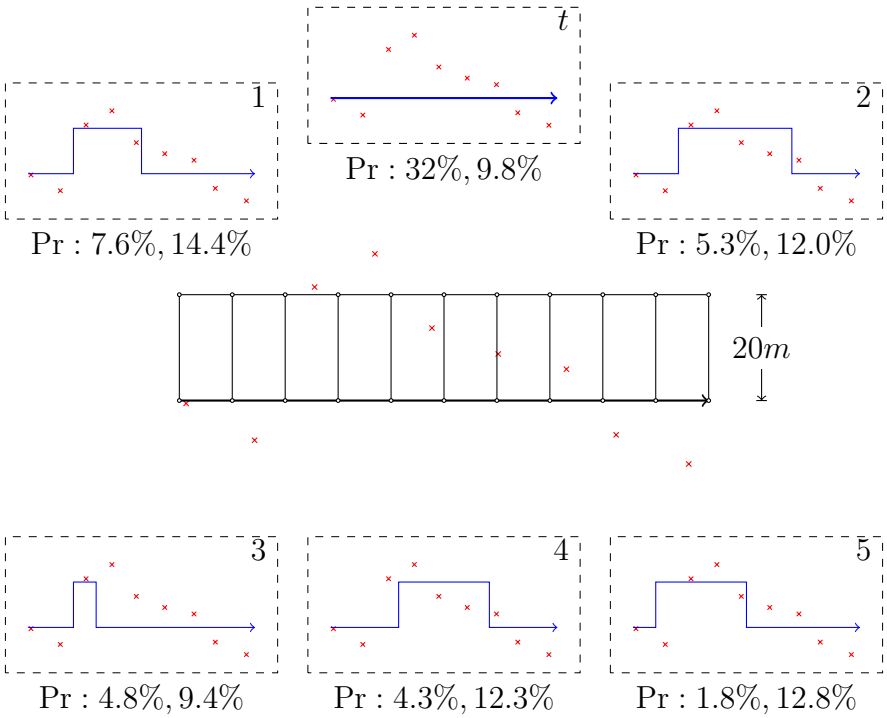


Figure 5: Result with $l = 20m$



and it basically just reflects the topological relationship between the observations and the network. For clarity, we call them spatial-temporal algorithm and spatial algorithm respectively . We can see from the figure that the two lines are so close that we can not recognize the true path from the results generated by both algorithm. Both algorithms fail in this case. However, when we extend l to $15m$ (see Figure 4), the spatial algorithm gives the highest probability values to the wrong paths (3 and 5). But the probability of the true path calculated by spatial-temporal algorithm is the highest. Further, l is extended to be $20m$ (see Figure 5). The true path gains the remarkable likelihood from spatial-temporal algorithm, but the spatial algorithm fails again. The failure of the spatial-temporal algorithm in the first case, and the only marginally highest result in the second case also show that it is not a panacea.

6 Conclusions

This paper proposed an algorithm for generating probabilistic matching of location data to paths in transportation network. It presented a theoretical framework for calculating path probability from a trace of location observations, as well as measurement formulations for defining the matching in a probabilistic fashion. Given a set of imprecise location observations, the algorithm generates the probability for each possible path having been the true one. An application of the algorithm to synthetic data showed the capability of the algorithm in recognizing the true path.

There are still much work to be done. Besides improving the algorithm by utilizing low speed GPS observations in a better way, real data should be collected and used to test the algorithm. A data collection campaign will be carried out soon in collaboration with Nokia Research Center in Lausanne, to collect GPS data from Nokia N95 mobile phones. Also the algorithm should be compared against the advanced map-matching algorithms. Estimating route choice models for travelers by using the path likelihoods for trips is our ultimate goal.

References

Bierlaire, M., J. Chen and J. Newman (2009) Using location observations to observe routing for choice models, Paper submitted to Transportation Research Boarding Annual Meeting 2009, August 2009.

Bierlaire, M. and E. Frejinger (2008) Route choice modeling with network-free data, *Transportation Research Part C: Emerging Technologies*, **16** (2) 187–198.

Marchal, F., J. Hackney and K. Axhausen (2005) Efficient map matching of large global po-

sitioning system data sets: Tests on speed-monitoring experiment in zürich, *Transportation Research Record: Journal of the Transportation Research Board*, **1935**, 93–100.

Ochieng, W., M. Quddus and R. Noland (2003) Map-matching in complex urban road networks, *Brazilian Journal of Cartography (Revista Brasileira de Cartografia)*, **55** (2) 1–18.

Quddus, M., W. Ochieng and R. Noland (2007) Current map-matching algorithms for transport applications: State-of-the art and future research directions, *Transportation Research Part C*, **15** (5) 312–328.

White, C. E., D. Bernstein and A. L. Kornhauser (2000) Some map matching algorithms for personal navigation assistants, *Transportation Research Part C: Emerging Technologies*, **8**, 91–108.