Trajectory Representation in Location-based Services: Problems & Solution

Nirvana Meratnia
Computer Science Department
University of Twente
P.O.Box 217, 7500 AE, Enschede
The Netherlands
Meratnia@cs.utwente.nl

Rolf A. de By
Intl. Inst. for Geo-information Science &
Earth Observation (ITC)
P.O. Box 6, 7500 AA Enschede
The Netherlands
Deby@itc.nl

Abstract

Recently, much work has been done in feasibility studies on services offered to moving objects in an environment equipped with mobile telephony, network technology and GIS. However, despite of all work on GIS and databases, the situations in which the whereabouts of objects are constantly monitored and stored for future analysis are an important class of problems that present-day database/GIS has difficulty to handle. Considering the fact that data about whereabouts of moving objects are acquired in a discrete way, providing the data when no observation is available is a must. Therefore, obtaining a "faithful representation" of trajectories with a sufficient number of discrete (though possibly erroneous) data points is the objective of this research.

1. Introduction

Recently, with the coming about of miniature and cheap GPS receivers and cellular phones, new horizons have been opened for services offered to moving objects in an environment equipped with mobile telephony, network technology and GIS. However, despite of all work on GIS and databases, the situations in which the whereabouts of objects are constantly monitored and stored for future analysis are an important class of problems that present-day database/GIS has difficulty to handle. Among all the issues in handling continuous change in these objects, such as data modeling, data structures and operations, large amount of generated data and storage issue, indexing methods, and query processing techniques; the main focus of this paper is on the issue of faithful representation of moving object trajectory.

One of the important problems in dealing with moving point objects is the acquisition of continuous data in a discrete way. The uncertainty of the representation of such object movement is affected by the frequency with which position samples are taken, i.e., the *sampling rate*. Special attention should be paid to situations in which the connection between object and data acquisition device is lost and data cannot be updated [4]. On the other hand, there exist retransmission protocols since computer systems cannot directly represent continuous phenomena of arbitrary nature, such phenomena must be approximated using finite means. Thus, unambiguous and optimal representation of moving objects is crucial.

The goal of this paper is to faithfully represent the moving object trajectories in the context of restricted movement on a road network. Such faithful representations can open the door for recognition of patterns of movement and object classification and consequently introduce information about the environment where the objects move.

The rest of this paper is organized as follows. In Section 2, the scenario with which we assumed to work is shortly described. In Section 3 a number of current trajectory representation techniques are discussed and their incapability is shown. Our proposal to faithfully represent and register trajectories is also addressed. Results of experiments and description of data used in such experiments are presented in Section 4. Finally, the paper concludes with recommendations for future work.

2. The Scenario

As data about moving objects is acquired in a discrete way, any further computation requires the use of positional interpolation techniques, in which one of the fundamental issues is the "medium" via which the objects are traveling. The medium may or may not impose restrictions on where objects go.

In the scenario we assume to work with, observed object, which is called the moving object throughout the paper, plays an important role. We use the phrase as a container term for various organic and inorganic entities that demonstrate mobility. A moving object can be a car equipped with a GPS receiver or a person traveling in a car carrying a cellular phone that enables positioning. Regardless of the data acquisition method, the main part of the moving object data is in the form of time-stamped positions, i.e., quadruples of the form <obj_id, x, y, t>. For an object obj_id, (x,y) represents the position of the moving object, at time t. The moving objects send such sequence of spatio-temporal data together with the queries (if any) to a central base station and subsequently receive the query results.

Another important componenet in our scenario is the road network data, as we are talking about restricted movement on a road network. For the sake of simplicity, <road_id, from, to, maxspeed> are assumed to be the main attributes of a road network data set. "From" and "to" indicate start and end nodes, "maxspeed" is the maximum allowed speed for a road segment with identification "road_id". We assume road network segments are one way roads. Other attributes can be added. The geometry is stored in the nodes.

3. Trajectory representation

As data about moving objects is acquired and represented in a discrete way, an appropriate technique is needed to provide the data when no observation is available, to determine the historical path, or future locations. In such cases, positional interpolation is used. As the method of interpolation is entirely dependent on the characteristics of the moving object, i.e., its maximum speed, maximum acceleration, maximum angular speed, etc, an important question is which interpolation technique should be chosen [1]. On the other hand, concerning our case other factors, such as traffic jams, traffic lights, should be also taken into account to determine the trajectory of a moving object as accurately as possible.

Having two consecutive position samples, the best we can do is to limit the possibilities of where the moving object could have been [3]. Therefore, a parametric distance function of time, $D(t)$, is used to express the relationship of the distance and time along the road network. As we are dealing with restricted movement along the road network, this function is sufficient to express the location of the object at any time. Moreover, object velocity $V(t)$ and acceleration $A(t)$ can be obtained by differentiating the distance function. However, an important question is how to choose the parameters representing the relationship between distance *(D)* and time *(t)*. It is important to note that to estimate the parameters involved in the distance function, in addition to the corresponding time values, the distances along the road network between the census points should be used. The importance of using the distances along the road network between the sample points is illustrated in Figure 1.

In this figure, the object is moving from road segment R_1 to road segment R_2. By using the measured distance along the road network, the distance D_{i+2} is being forced to pass through the intersection node of R_1 and R_2, instead of passing through a straight line connecting data points P_{i+1} and P_{i+2}.

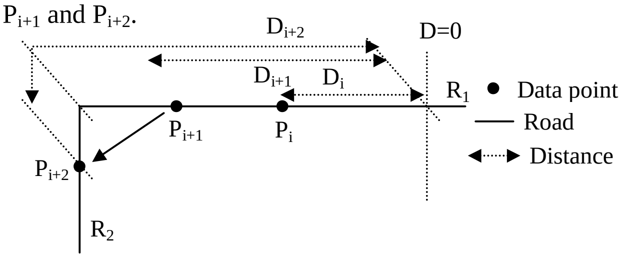

Figure 1. Constraining distance functions to the road network. As it can be seen for points P_i and P_{i+1}, using a plain interpolation without considering the road network characteristics is not appropriate.

3.1. Problems in trajectory representations

Most of the previous work on trajectory representation used linear interpolation as a simple interpolation technique. However, linear interpolation may not be suitable enough to describe a movement, as we will show shortly. In our previous work [2], a spline interpolation technique was used to represent the movement of a moving object, in which the distance function (D) was constrained along the road network. In this paper, we will closely analyze the behavior of both linear interpolation and spline.

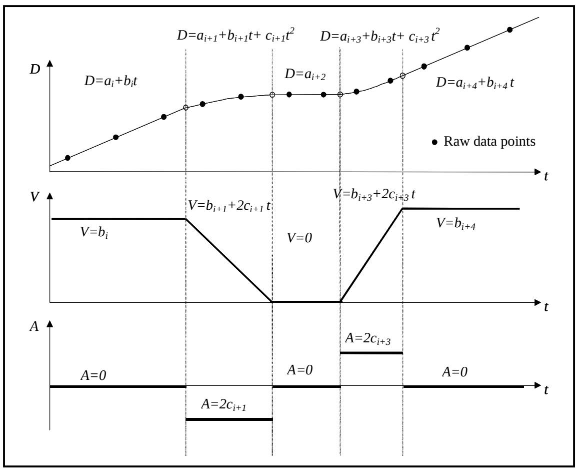

Figure 2. Profiles of distance (D), velocity (V) and acceleration (A) functions representing a single moving object

Figure 2 shows profiles of distance (*D)*, velocity (*V)* and acceleration (*A)* functions versus time (*t)* for a normal movement, in which a moving object travels with constant speed, decelerates to stop, waits, accelerates and travels again with constant speed.

Using linear interpolation for representing such movement results in functions shown in Figure 3. It can be clearly seen from the figure why linear interpolation is only suitable for situations in which object moves with constant speed.

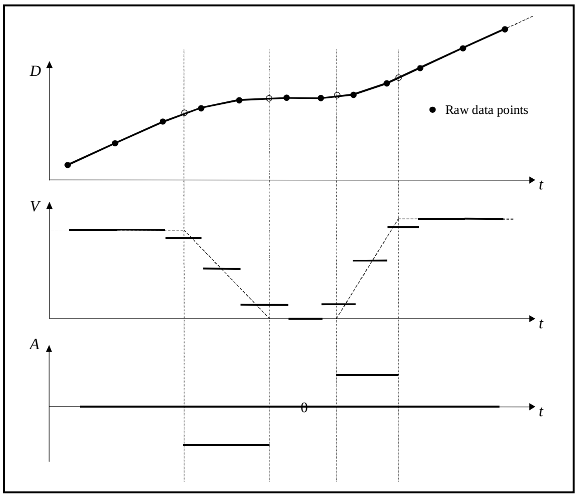

Figure 3. Linear interpolation functions of data points in Figure 2.

On the other hand, Figure 4 shows that the spline method generally gives more realistic results for representing the whole movement. However, in this figure we observe unusual behavior of velocity and acceleration functions in the period that the object is likely stationary.

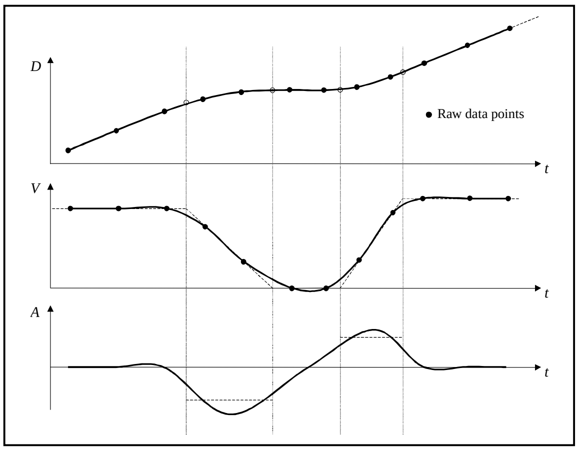

Figure 4. Spline interpolation functions of data points in Figure 2.

Depending on how long the moving object is in stationary mode, the unusual behavior of velocity and acceleration functions differ. If moving object waits for a relatively long period of time, a spline will represent the object movement in a back and forth order to compensate for waiting time. This means that the object is assumed to be moving backward and forward during the waiting time. On the other hand, as a spline is a smooth function, in order to keep the continuity of the function, it may misrepresent object movement during the waiting time, which means to present a traveled distance while the object is stationary. This case happens when the waiting time is relatively short.

Figure 5 and 6 illustrate the unusual behavior of spline in case of long and short waiting times, respectively.

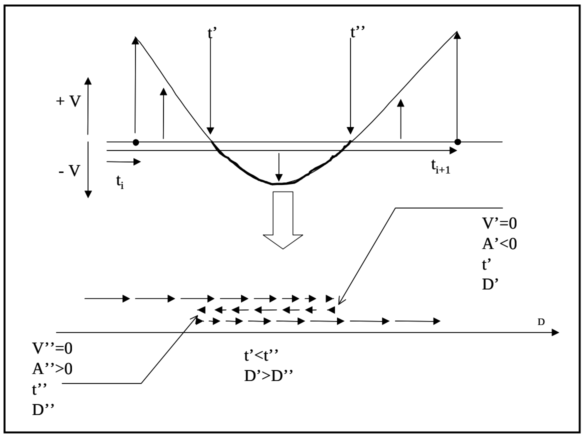

Figure 5. Spline forces the moving object to move back and forth to compensate relatively long waiting time

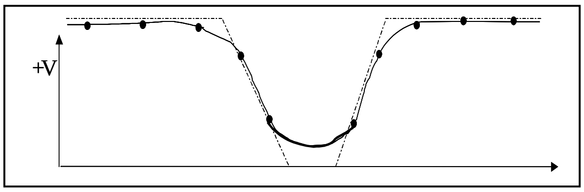

Figure 6. Spline may misrepresent the moving object velocity for relatively short waiting time

Therefore, using the same general function to describe the complete movement should be avoided. This means that faithful representation of a moving object trajectory requires a clear distinction between different patterns of movement (states of the trajectory). Movement in each of such states should be described by a different function which suites the movement best.

On the other hand, we can also see from Figure 3 and 4 that most of the errors in trajectory representation occur at specific points (break points), in which there is an abrupt change in movement. Such abrupt change may occur due to the approach of obstacles, bends or turns. Regardless of the reason for abrupt change, the moving object will not follow the same routine pattern of movement as before, therefore a smooth interpolation function such as a spline cannot describe the changes that actually occur. These break points are in fact the distinction points between different states of movement. Different distance functions might be used between each two consecutive break points. Due to the importance of such break points, determining them is a must. To do so, a *break point extraction technique* is proposed and discussed in the following section.

3.2. Break point extraction

As mentioned earlier, '*break points*' are points in which abrupt changes in velocity and acceleration occur. These points are important as they are points in which moving object does not continue its routine pattern of movement. Ignoring such points while using any sort of interpolation results in failure of the interpolation technique to faithfully represent the moving object trajectory at these points. These are points that require especial attention and we propose the following steps determine them:

- *Data point grouping:* to detect and group consecutive data points that together represent a "constant" movement pattern (constant speed, constant acceleration, constant deceleration, or waiting). The overall knowledge about spatial error (uncertainty associated with the data) is used in this step.

- *Parametric function fitting:* to fit a parametric function through the data points in each detected group.

- *Intersecting consecutive functions:* to intersect the functions representing consecutive groups, to determine the break points.

We explain each of these steps more elaborately in the sections below.

3.2.1. Data point grouping. In order to group the data points, velocity and acceleration values in consecutive data points are compared. We assume not having measured the speed values directly. By comparing two consecutive velocity values from three data points $<D_i,t_i>$, $<D_{i+1},t_{i+1}>$, $<D_{i+2},t_{i+2}>$, we will have:

$$\Delta V = V_{i+1} - V_i = \frac{D_{i+2} - D_{i+1}}{t_{i+2} - t_{i+1}} - \frac{D_{i+1} - D_i}{t_{i+1} - t_i} \quad (1)$$

If we assume no error in the data, $\Delta V=0$ would mean a constant speed for the three data points. However, as data may be contaminated by error, an error propagation analysis should be performed. Therefore, the uncertainty of ΔV, denoted as $\sigma_{\Delta V}$, can be expressed on the basis of uncertainty in the location and time values as:

$$\sigma^2_{\Delta V} = K_{\Delta V} \Sigma_{D,t(i,i+1,i+2)} K^T_{\Delta V} \quad (2)$$

, where $K_{\Delta V}$ is the Jacobean matrix representing the partial derivatives of ΔV with respect to D_i, D_{i+1}, D_{i+2}, t_i, t_{i+1}, and t_{i+2} as:

$$K_{\Delta V} = \begin{bmatrix} \frac{\partial \Delta V}{\partial D_i} & \frac{\partial \Delta V}{\partial D_{i+1}} & \frac{\partial \Delta V}{\partial D_{i+2}} & \frac{\partial \Delta V}{\partial t_i} & \frac{\partial \Delta V}{\partial t_{i+1}} & \frac{\partial \Delta V}{\partial t_{i+2}} \end{bmatrix} \quad (3)$$

and $\Sigma_{D,t\,(i,\,i+1,\,i+2)}$ is the variance covariance matrix for D_i, D_{i+1}, D_{i+2}, t_i, t_{i+1} and t_{i+2}, which can be expressed as:

$$\Sigma_{D,t(i,i+1,i+2)} = \begin{bmatrix} \sigma_D^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_D^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_D^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_t^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_t^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_t^2 \end{bmatrix} \quad (4)$$

In this variance covariance matrix, no correlation is assumed. We also assume equal distance and time variances, which means $\sigma_D = \sigma_{Di} = \sigma_{Di+1} = \sigma_{Di+2}$ and $\sigma_t = \sigma_{ti} = \sigma_{ti+1} = \sigma_{ti+2}$.

To detect the sequence of points having similar speed, a procedure of "sequence growing" is used. In this procedure, each point in the sequence is compared to its two previous data points. For the three points, ΔV is computed using Equation 1, and is compared to its expected uncertainty (σ_Δ) as computed from Equation 2. Based on such comparison, either the new point is classified into the same group, or is used to start a new group.

It is important to note that while a group of points with similar velocity (different than zero) can be classified as a group of similar acceleration (zero acceleration), a group with constant acceleration (different than zero) cannot be classified as having similar velocity. Therefore, the groups of constant velocity are detected first. Then, for the rest of the unclassified data points we will detect groups of similar acceleration.

To group data points based on acceleration value, a similar procedure could be performed, except that in this stage four consecutive data points would be needed. Therefore we will have:

$$\Delta A = A_{i+1} - A_i = 2\frac{\begin{vmatrix} 1 & t_{i+1} & D_{i+1} \\ 1 & t_{i+2} & D_{i+2} \\ 1 & t_{i+3} & D_{i+3} \end{vmatrix}}{\begin{vmatrix} 1 & t_{i+1} & t_{i+1}^2 \\ 1 & t_{i+2} & t_{i+2}^2 \\ 1 & t_{i+3} & t_{i+3}^2 \end{vmatrix}} - 2\frac{\begin{vmatrix} 1 & t_i & D_i \\ 1 & t_{i+1} & D_{i+1} \\ 1 & t_{i+2} & D_{i+2} \end{vmatrix}}{\begin{vmatrix} 1 & t_i & t_i^2 \\ 1 & t_{i+1} & t_{i+1}^2 \\ 1 & t_{i+2} & t_{i+2}^2 \end{vmatrix}} \quad (5)$$

This value should be compared to its expected uncertainty ($\sigma_{\Delta A}$), which can be expressed by means of error propagation as:

$$\sigma^2_{\Delta A} = K_{\Delta A} \Sigma_{D,t(i,i+1,i+2,i+3)} K^T_{\Delta A} \,, \quad (6)$$

where $K_{\Delta A}$ is the Jacobean matrix representing the partial derivatives of ΔA with respect to D_i, D_{i+1}, D_{i+2}, D_{i+3}, t_i, t_{i+1}, $t_{i+2}$and t_{i+3} as:

$$K_{\Delta A} = \begin{bmatrix} \frac{\partial \Delta A}{\partial D_i} & \frac{\partial \Delta A}{\partial D_{i+1}} & \frac{\partial \Delta A}{\partial D_{i+2}} & \frac{\partial \Delta A}{\partial D_{i+3}} & \frac{\partial \Delta A}{\partial t_i} & \frac{\partial \Delta A}{\partial t_{i+1}} & \frac{\partial \Delta A}{\partial t_{i+2}} & \frac{\partial \Delta A}{\partial t_{i+3}} \end{bmatrix} \quad (7)$$

and $\Sigma_{D,t\ (i,\ i+1,\ i+2,\ i+3)}$ is the variance covariance matrix for D_i, D_{i+1}, D_{i+2}, D_{i+3}, t_i, t_{i+1}, $t_{i+2}$and t_{i+3} , which can be expressed as:

$$\Sigma_{D,t(i,i+1,i+2,i+3)} = \begin{bmatrix} \sigma_D^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_D^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_D^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_D^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_t^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_t^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sigma_t^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma_t^2 \end{bmatrix} \quad (8)$$

Again, no correlation, but equal distance variances, and equal time variances are assumed. The same procedure as for detecting constant speed data points would be used.

3.2.2. Parametric function fitting. After classifying all data points and detecting the groups of constant velocity and the groups of constant acceleration, simple distance functions can be used to fit through the data of each group. To do so:

- For points belonging to a constant speed sequence, a linear function in the form of $D = a + b\ t$ will be defined. When more than two points are detected in such a group, a least squares function fitting is used.
- For points belonging to a constant acceleration sequence, a cubic spline function in the form of $D = a + b\ t + c\ t^2$ will be defined. When more than three points are detected in such a group, a least squares function fitting is used.

It is important to note that least square fitting reduces the uncertainty in the data. The larger the number of points in each group is, the lower the uncertainty in trajectory representation.

3.2.3. Intersecting consecutive functions. After the parameters of the functions have been determined in the previous step, an intersection of the functions representing consecutive groups is computed. It provides us with the time values of the break points (see Figure 7). These break points can be used to restrict the time range for each of the functions.

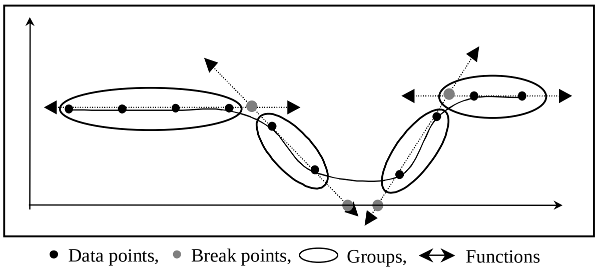

Figure 7. Identifying break points by fitting functions through groups of data points

4. Experiments

The proposed method was tested using a simulation technique. The evaluation procedure starts by simulating so-called true data for both the moving object and the road network. In the next two sections, the data description and the analysis of the results are discussed.

4.1. Data description

A road network was simulated, consisting of 49 nodes and 51 links. Each link connects two nodes and has attributes such as direction and maximum speed limit. Links were chosen to be straight lines for simplicity; however, the algorithm is not limited to such cases. As links are straight lines, all geometry is stored in the nodes. Nodes are classified as either dead-ends or intersections.

The moving objects were randomly simulated in the network. Start points, end points, speed and acceleration values were chosen randomly respecting the road network characteristics. Homogeneous spatial errors were introduced with a mean of zero and standard deviation of 10.0 meters. Data points, in form of <obj_id,x,y,t>

together with the road network geometry and topology formed the input data.

4.2. Results

An important observation is that both spline and linear interpolation methods do not use errors or uncertainty measures. On the other hand, our proposed method requires uncertainty measures of the location along the road network. The uncertainty value along the link is assumed to be equal to the error perpendicular to the link. This error is computed based on our spatial registration method proposed in [2]. After detecting the data sequences with similar characteristics, least squares polynomial fitting was implemented. Intersecting the functions representing the consecutive groups results in break point detection. Positional errors were plotted on the road networks for the three trajectory representation methods (spline, linear interpolation and our proposed method).

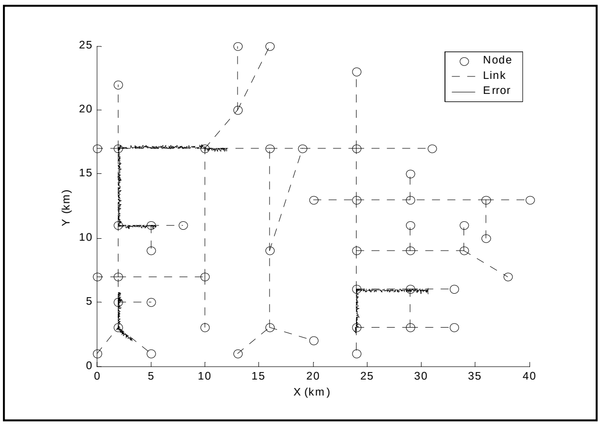

Figure 8. Interpolation error in spline

Corresponding root mean square errors were computed and were found to be 7.4 m, 7.8 m and 0.019 m, for spline, linear interpolation and our proposed method, respectively. Out of all simulated moving objects, obtained results of applying linear interpolation, spline and our proposed trajectory representation methods on three moving objects are shown in Figures 8, 9 and 10.

As can been seen in these figures as well as from comparing the root mean square error values, spline and linear interpolation methods are highly affected by the noise contamination in the data. On the other hand, the break point extraction method suppresses the noise in the data due to the large number of data points between two consecutive break points, which enhances the estimated parameters and consequently reduces the uncertainty.

As Spline already performs better than linear interpolation, to enable a better comparison, much more zoomed views of interpolation error in spline and our proposed method for one of the trajectories are shown in Figure 11.

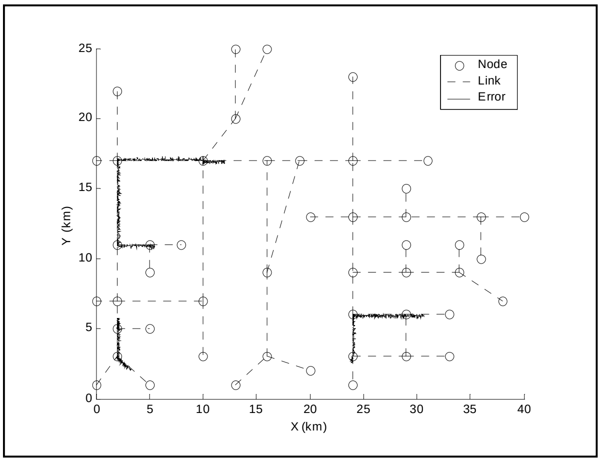

Figure 9. Interpolation error in linear interpolation

Our method, unlike the spline method, deals with each data point individually. Therefore, the time complexity involved in computations of our proposed method is O(N), where N is the number of data points. Another important advantage of our method is its contribution to data compression. For instance, applying our break point extraction algorithm on 2511 simulated raw data resulted in 43 break points, 16 constant speed intervals (two parameters each) and 24 constant acceleration intervals (three parameters each).

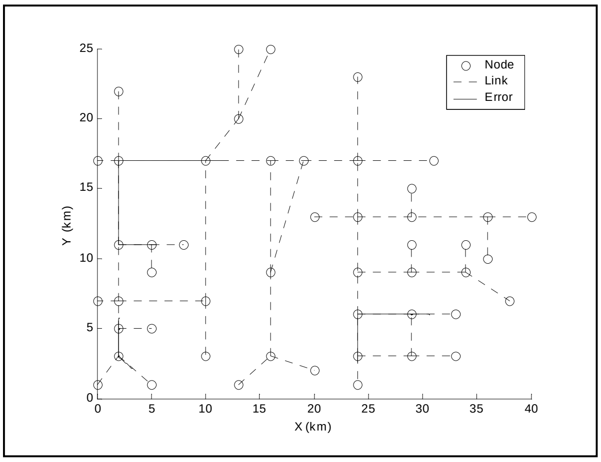

Figure 10. Interpolation error in our method

Assuming equal number of data bytes for restoring time and distance, the raw data are compressed by 98%, which is a very large compression rate.

5. Conclusions and recommendations

In this paper, by focusing on the case of restricted movement on a road network, different trajectory representation techniques to faithfully represent the trajectory of a moving object are discussed, compared and their incapability is shown.

The concept of "break points" is introduced as the points along the moving object trajectory in which an abrupt change in movement occurs. A method for extracting break points based on data sequence grouping is presented. The grouping method depends on the uncertainty in the data. The procedure includes least squares functional fitting and intersections. It has been shown that using our proposed method, a fair representation of moving object trajectory as well as a high compression rate with reasonably low error can be achieved.

Obtained information about individual objects can provide overall information about the road network itself, which enables global query analysis. This issue together with data compression are to be discussed in future work.

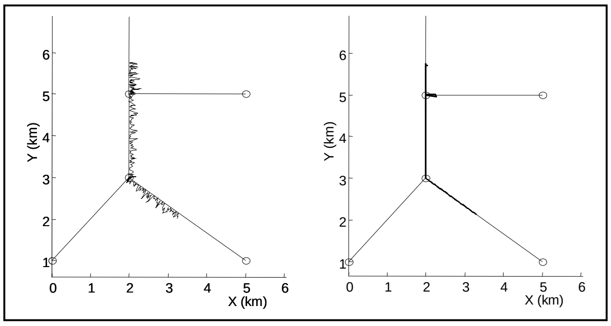

Figure 11. (Left) Interpolation error in spline, (Right) Interpolation error in our proposed method

6. References

[1]. Meratnia, N., De By, R. A., "Moving objects data modeling". Proceedings of ASPRS 2001 Annual Conference, St. Louis, USA, 2001

[2]. Meratnia, N., Kainz, W., De By, R. A. "Spatio-temporal Methods to Reduce Data Uncertainty in Restricted Movement on a Road Network". In book "Advances in Spatial Data Handling. 10th International Symposium on Spatial Data Handling", Richardson, D. and van Oosterom, P. (Eds.), 8-12 July, Ottawa, Canada, 2002, pp. 391-402, ISBN 3-540-43802-5

[3]. Pfoser, D., Jensen, C., "Capturing the Uncertainty of Moving-Object Representations", Proceedings of the 6th International Symposium, 1999, Springer-Verlag, Series LNCS 1651

[4]. Wolfson, O., Sistla, P., Xu, B., Zhou, J., Chamberlain, S. "Tracking Moving Objects Using Database Technology in DOMINO", Proceedings of the 4th Workshop on Next Generation Information Technologies and Systems (NGITS). Zikhron-Yaakov, Israel., 1999 Lecture Notes in Computer Science, number 1649, Springer-Verlag pp 112-119