DETECTING AND CORRECTING MAP-MATCHING ERRORS IN LOCATION-BASED INTELLIGENT TRANSPORT SYSTEMS

*Nagendra R. Velaga**
Mohammed A. Quddus
Abigail L. Bristow

Transport Studies Group, Department of Civil and Building Engineering, Loughborough University, Loughborough, Leicestershire, LE11 3TU, UK.
*(*Corresponding Author's Email: n.r.velaga@lboro.ac.uk)*

ABSTRACT

Map-matching (MM) algorithms integrate the data from positioning sensors (such as GPS) with a digital map in order to identify firstly, the road link on which a vehicle is travelling from a set of candidate links; and secondly, to determine the vehicle's precise location on that segment. Due to errors in positioning sensors, digital maps and the map-matching process, MM algorithms sometimes fail to identify the correct road segment from the candidate segments. This phenomenon is known as mismatching. Identification of the wrong road link could mislead users and may degrade the performance of the ITS services and reduce their efficiency. Therefore, the main objective of this paper is to improve a topological map-matching (tMM) algorithm by error detection, correction and performance re-evaluation.

Errors in a tMM algorithm are determined using data (62,887 positioning points) collected in three different countries (UK, USA and India). After map-matching, each mismatched case was examined to identify the primary causes of the mismatches and a number of strategies were developed to correct these mismatches enabling enhancement of the tMM algorithm. An independent dataset (5,256 positioning points) collected in and around Nottingham, UK, was employed to re-evaluate the performance of the enhanced tMM algorithm. The result suggests that the enhanced tMM algorithm correctly identified road segments 97.8% of the time. With the same positioning data, the success rate was found to be 96.5% before the enhancement. The enhanced tMM algorithm developed in this research is simple, fast, efficient and easy to implement. Since the accuracy offered by the enhanced algorithm is found to be high, the developed algorithm has potential to be implemented in real-time location-based ITS applications.

Keywords: Map-matching, GPS, Location-based ITS services, Genetic Algorithm.

INTRODUCTION

Many Intelligent Transport System (ITS) applications require real-time vehicle positioning data. A navigation system that provides such positioning data consists of three key components: (1) a positioning system such as Global Positioning System (GPS) or GPS integrated with a dead-reckoning (DR) system (2) a Geographic Information System (GIS) based road map and (3) a map-matching (MM) algorithm (Greenfeld, 2002; Taylor and Blewitt, 2006). A map-matching (MM) algorithm is used to augment positioning data from a navigation system with spatial road network data. A MM algorithm makes use of a range of positioning and navigational data including position, heading, speed and road network topology to identify the correct road segment on which a vehicle is travelling and the vehicle's location on that road segment (Quddus et al., 2007). The key task for a MM algorithm is to identify the correct road segment from a pool of candidate road segments (Velaga et al., 2009).

The raw positioning data from a navigation system contain errors due to satellite orbit and clock bias, atmospheric (ionosphere and troposphere) effects, receiver measurement error and multipath error (Kaplan and Hegarty, 2006). GIS based road maps include errors which can be geometric (e.g., displacement and rotation of map features) or topological (e.g., missing road features) (Goodwin and Lau, 1993; and Kim et al., 2000). Even when the raw positioning data and map quality are good, MM techniques sometimes fail to identify the correct road segment, especially at roundabouts, level-crossings, Y junctions, dense urban roads and parallel roads (White et al., 2000; Quddus et al., 2007). Any error associated with either the raw positioning fix, the digital map or the MM process employed can lead to wrong link identification. This phenomenon is known as mismatching. Identification of the wrong road link may mislead the users and may reduce the effectiveness of the ITS service. It is therefore important to identify the reasons for mismatches and use this information to develop strategies for the purpose of increasing the performance of MM algorithms so as to enable the further enhancement in navigation modules of ITS.

Navigation modules of ITS may be enhanced by reducing the errors in the positioning sensors or improving the quality of the spatial road network map or enhancing the map-matching process. For instance, the Russian Global Orbit Navigation Satellite System (GLONASS) and the upcoming European Galileo system, along with a DR system, can enhance the performance of existing vehicle navigation systems. A good map-matching algorithm would however be fundamental to physically locate a vehicle on a road network. Current map-matching algorithms have many constraints and limitations, especially in typical operational environments (such as dense urban areas) where highly accurate positioning data are essential (see Quddus et. al., 2007). The enhancement in vehicle positioning can also be obtained by improving the quality of spatial network data at a national level. Many countries are investing resources for the purpose of improving the quality of GIS road maps. This however takes a long time to become available to users. Therefore, further improvement in map-matching algorithms can be considered as a viable option to enhance vehicle navigation modules.

Here, a weight-based tMM algorithm developed by the authors (Velaga et al., 2009) is used for detecting and correcting map-matching errors. After map-matching of an extensive positioning dataset, each mismatched case was examined carefully. A thematic analysis of the mismatching was carried out to identify errors due to: (1) positioning sensor (2) digital map and (3) the map-matching process. A number of strategies were identified to correct these mismatches enabling enhancement of the tMM algorithm. The tMM algorithm was modified accordingly, and its performance is re-evaluated using an independent positioning dataset. The enhanced algorithm performance, with respect to correct link identification and horizontal accuracy, is reported for each of the improvement strategies separately.

The paper is organised as follows: the next section provides a brief description of the MM algorithm used in this study. This is followed by an outline of the positioning data used in this study. The process of detecting and correcting map-matching errors is then explained. This includes investigating the main reasons for each mismatch, developing the strategies to enhance the map-matching algorithm, correcting the algorithm accordingly and finally re-evaluating the enhanced algorithm using an independent positioning data set. The paper ends with conclusions.

THE MAP-MATCHING ALGORITHM

A MM algorithm that uses historical data (such as the previously matched road segment), vehicle speed and topological information on the spatial road network (such as link connectivity) is called a topological map-matching (tMM) algorithm (Greenfeld, 2002; Li et al., 2005; Quddus et al., 2003). A tMM algorithm is fast, simple and easy to implement (Velaga et al., 2009). An algorithm that assigns weights for all candidate links based on different criteria such as the similarity in vehicle movement direction and link direction, the nearness of the positioning point to a link, and the connectivity of a candidate road link to the previously travelled road link is known as a weight-based topological MM algorithm (Quddus et al., 2003; Velaga et al., 2009). A weighting approach in selecting the correct road segment from the candidate segments improves the accuracy of correct road link identification (Greenfeld, 2002; Quddus et al., 2003).

The map-matching process of the tMM algorithm used in this research is divided into three stages: (1) initial matching, (2) matching on a link and (3) matching at a junction. The aim of the initial MM process is to identify the correct link for the first positioning point. Initially, the algorithm creates an error circle around the first positioning point. Then, the algorithm selects all the links which are inside and touching the error circle as candidate links. Among these candidate links, the correct link identification is based on the total weight score, which is the sum of the heading (similarity in vehicle movement direction and link direction) and proximity (nearness of positioning point to a link) weights. The vehicle's location on that selected link is then estimated. This is achieved by a perpendicular projection of the positioning point onto the link. After successful completion of the initial MM process, the algorithm calculates the distance to the downstream junction. If the vehicle is at or near a junction, then the algorithm goes to the stage of 'matching at a junction' otherwise the algorithm goes to the 'matching on a link' stage. In case of 'matching on a link', the algorithm snaps the current positioning point to the previously selected road segment. If the vehicle is at or near a junction, it is necessary to identify a new road segment. The procedure for the identification of a set of candidate segments for a positioning point at a junction is similar to that of the initial MM process. The correct link is selected, from candidate links, based on the total weight score (TWS). However, at this stage, two additional weights are introduced on turn restrictions at junctions and link connectivity. Then the perpendicular projection to the selected link gives the vehicle location on that link. A detailed description of the map-matching process can be found in Velaga et al. (2009).

It is important to identify the relative importance of the weight scores used in the weight-based tMM algorithm. Moreover, the relative importance of each weight score used in the weight-based tMM algorithm may vary with operational environments. Earlier studies, by Greenfeld (2002) assumed equal importance for various factors and Quddus et al. (2003) determined relative importance empirically based on a true input-output dataset. Velaga at al. (2009) introduced an optimisation algorithm to identify the relative importance of weights for three operational environments: urban, suburban and rural. Using three positioning data sets from each operational environment, the relationship between percentage of wrong link identification and coefficient of weight scores were identified. Regression equations were developed for each of the operational environments. These three functional relationships (i.e., map-matching error vs weight coefficients) were optimised using the constrained nonlinear minimisation method. The sample sizes (i.e., number of junctions) used in the regression analysis, to find the relationship between map-matching error and the weight coefficients, in urban, suburban and rural areas were 175, 440 and 40 respectively. Further details on the optimisation algorithm and the optimal weight scores for three different operational environments can be found in Velaga at al. (2009).

The algorithm performance was then tested in dense urban environments in London and Washington, D.C, and a suburban area of London. The algorithm success rate in correct link identification in dense urban areas of central London, Washington and suburban area of London was found to be 96.8%, 95.9% and 96.7% respectively (Velaga et al., 2009).

POSITIONING DATA

Four positioning datasets collected from three different countries were used in the research reported here (see Table 1).

Table 1 - Positioning datasets used in this research

Data set	Test location	Date	Equipment used	Sample size (hours)	Route Length (Km)	Location characteristics
1	Loughborough to London; Central London and South part of London, UK	May-08	AEK - 4P and AEK-4R	42,231 points (11.7 hours)	700	Mix of dense Urban, Suburban, and Rural
2	An urban area with narrow congested roads in Mumbai, India	Dec-08	AEK-4P	16,756 points (4.6 hours)	95	Urban
3	Dense urban roads in Washington, DC, USA	Jan-09	AEK-4P	3,900 points (1.1 hour)	17	Dense urban
4	In and around Nottingham, UK	May-09	AEK - 4P, AEK-4R, and a Carrier-phase GPS with INS	5,256 points (1.5 hours)	56	Urban and Suburban

Dataset 1, 2 and 3 were collected in the United Kingdom (a trip from Loughborough to London, central London and south part of London), Mumbai metropolitan area, India and urban roads of Washington, D.C. respectively. These three datasets were obtained from controlled field tests, which were carried out on pre-planned routes. The test routes were selected carefully to ensure that the vehicle travelled through a good mix of characteristics such as tall building, bridges, flyovers, and dense road network, congested urban roads, open areas, construction sites, motorways, rural roads etc. These three datasets (i.e., 1, 2 and 3) were used for the map-matching error detection process. Dataset 4 was collected in and around Nottingham, UK. For this dataset, a true (reference) trajectory was obtained from a carrier phase GPS. Dataset 4 was used to evaluate the performance of the map-matching algorithm (before and after enhancing the algorithm). All the positioning points are obtained for every second. The test trajectory for the four data sets is shown in Figure 1. In this research, to analyse the above data sets three separate digital road maps, UK map (map scale 1:2,500), Washington DC map (map scale 1:1,250), and Mumbai map (map scale 1:25,000), were used. For all these maps, roads are represented with the road central line.

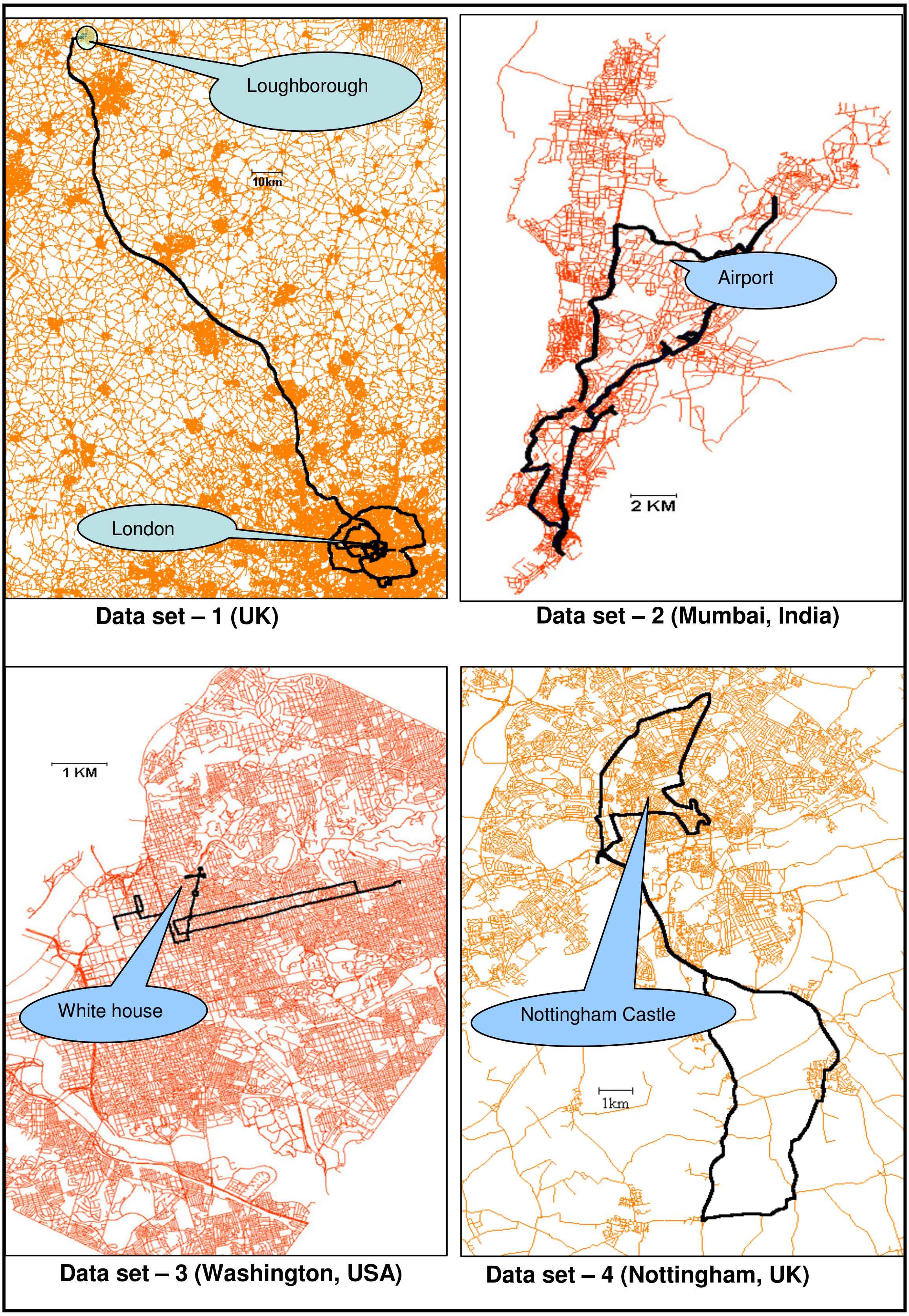

Figure 1 – Positioning data

ALGORITHM ENHANCEMENT METHODOLOGY

A step-by-step process of the enhancement of the tMM algorithm, which consists of error detection, correction and re-evaluation process, is shown in Figure 2.

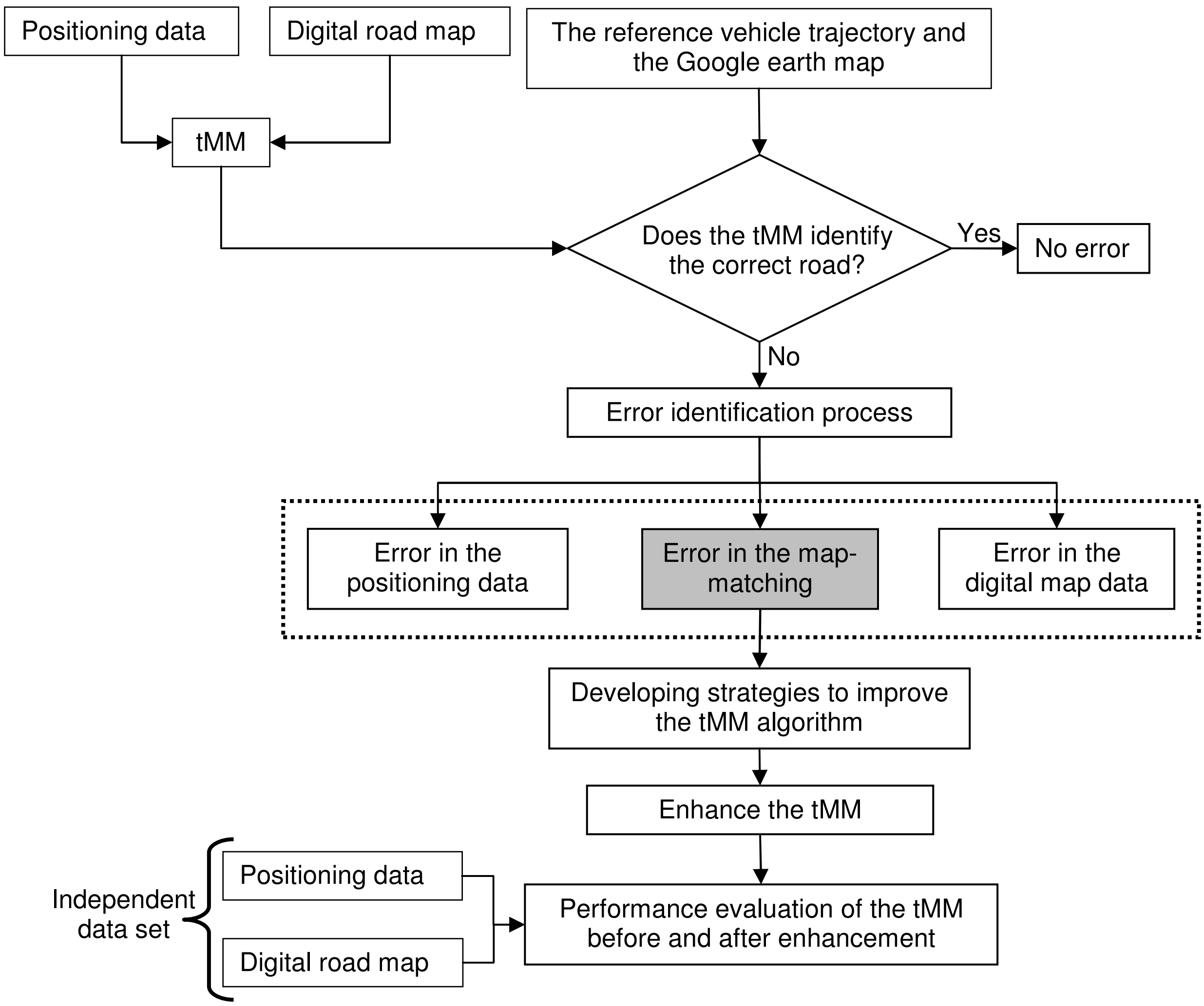

Figure 2 - Map matching error detection, correction and the performance re-evaluation

The output of the tMM algorithm provides a road segment on which a vehicle is travelling. If the road segment selected by the tMM algorithm is the actual (true) road segment for that particular positioning point, then it is assumed that there is no error in link identification. Otherwise, the map-matched point falls under the mismatching case. This process was conducted for all 62,887 positioning points to identify the main reasons for mismatching involving errors in the positioning data or the digital map or the map-matching process. The error detection process is followed by the development of different strategies to improve the performance of the tMM algorithm and then to modify the algorithm accordingly. Finally, the performance of the enhanced tMM algorithm (before and after enhancement) is examined using an independent positioning data set. The following section describes the mismatching identification process.

ERROR IDENTIFICATION PROCESS

The error detection process is carried out by classifying the mismatches due to errors in the positioning data, the digital map, and the map-matching process. The quality of raw position points is decided based on the number of visible satellites and the value of the Horizontal Dilution of Precision (HDOP) representing the quality of positioning solution. The UK Google earth satellite image is used as a base map to check errors in the GIS road maps (examples include, topological and geometric errors, missing links, extra links and digitisation errors). If the quality of a raw positioning point is good and no errors are identified in the digital map, then the reason for mismatching is assumed to be an error in the map-matching process. Each of mismatches due to the MM process is examined carefully to identify which part of the algorithm (i.e., the candidate link identification, the total weight score calculation and the consistence checks) caused the mismatching. Though the overall performance of the tMM algorithm is good, there are some mismatches in situations where the vehicle took a ‘U’ turn at junctions or at complex road configurations (i.e., Y junctions, roundabouts and parallel roads etc). The main cause of each mismatching case was identified through careful observation and critical judgement.

Table 2 shows the results from the tMM algorithm on three separate data sets collected in from UK, USA and India.

Table 2 – Reasons for mismatches

	UK (Data set 1)	Washington, DC, USA (Data set 3)	Mumbai, India (Data set 2)
Positioning data sample size	42,231	3,900	16,756
Mismatches due to positioning sensor error	472 (33.9%)	47 (29.6%)	159 (11.6%)
Mismatches due to digital map error	238 (17.1%)	27 (17.0%)	985 (71.7%)
Mismatches due to MM process errors	683 (49.0%)	85 (53.4%)	230 (16.7%)
Total number of mismatches	1,393	159	1,374

The tMM algorithm has 96.7%, 95.9% and 91.8% success rate of correct road link identification with data set 1, 3 and 2 respectively. From 62,887 map-matched positioning points, a total of 2,926 mismatches were discovered. In order to find out the reasons (i.e., errors in positioning data, map-matching process and digital map) of mismatching, each mismatching case was individually examined. In Table 2, the percentage contribution of mismatches due to the corresponding error is provided in parenthesis. It can be seen that about half of the mismatches in data sets 1 and 3 are due to errors in the map-matching process. The similar result for data set 1 and 3 suggest that the algorithm may be transferable. In case of data set 2, the major contribution of mismatches is digital map errors. This is due to the fact that the Mumbai GIS map has more missing links and digitisation errors. Clearly, the map-matching errors are predominant where the digital maps are good.

ENHANCEMENT OF THE MAP-MATCHING ALGORITHM

From 62,887 map-matched positioning points, a total 998 of mismatches were found due to errors in map-matching process. The tMM algorithm failed to identify the correct road segment particularly in complex road configurations (such as Y junctions, roundabouts and parallel roads). Examples of mismatching cases at a roundabout and at a Y junction are shown in Figures 3 and 4 respectively.

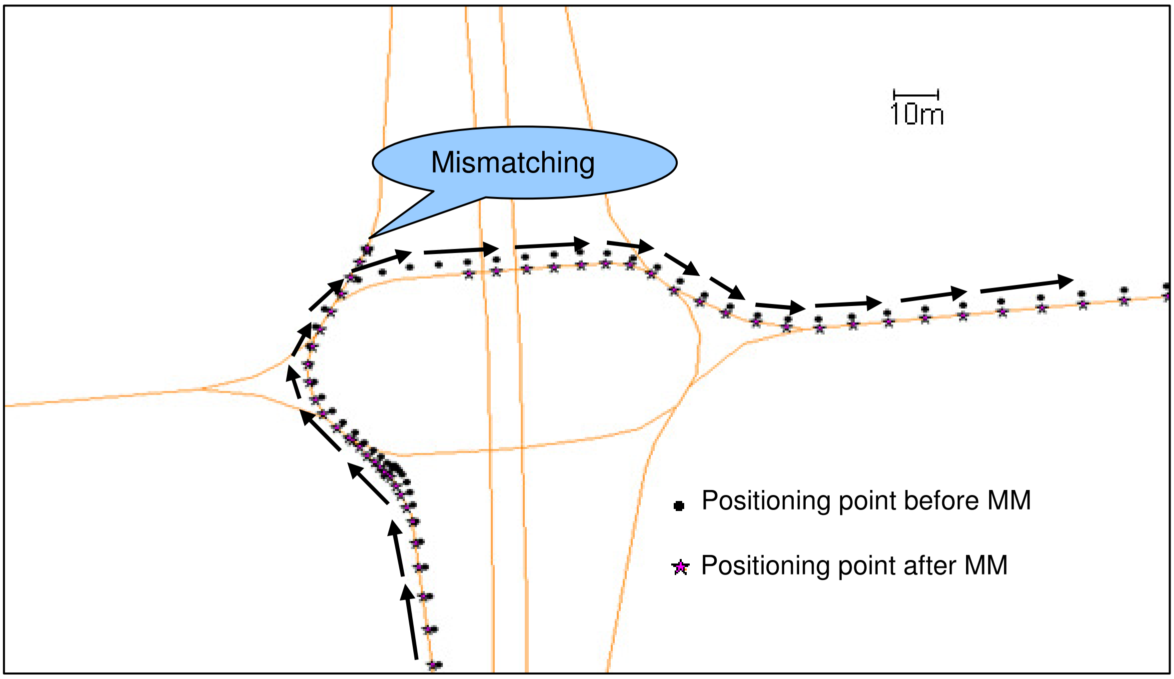

Figure 3 - Mismatching at roundabout

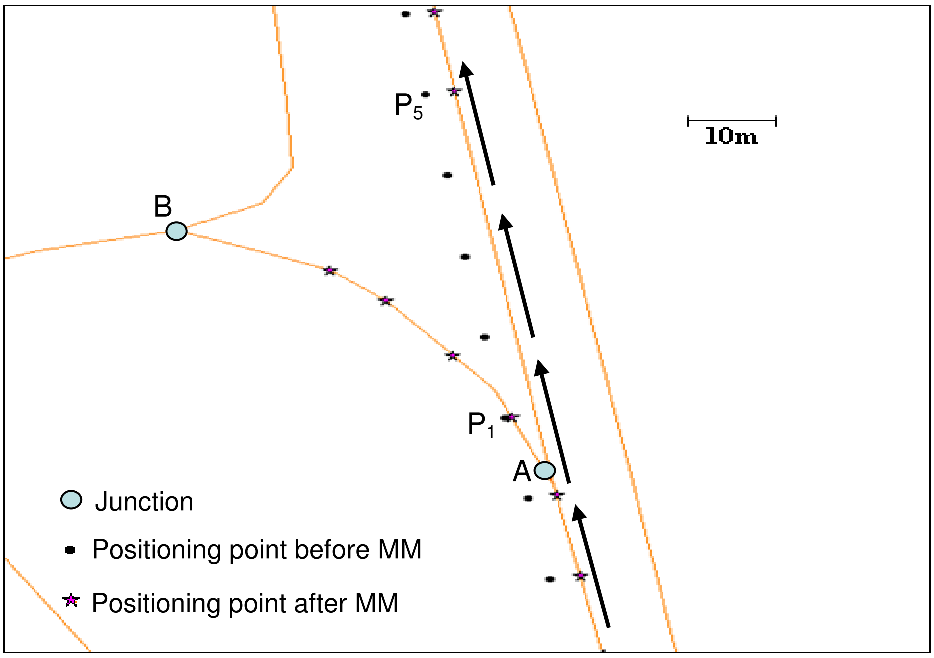

Figure 4 - Mismatching at Y junctions

Figure 4, at junction *A*, for positioning fix P_1 the algorithm identified the wrong road segment (i.e., link *A*-*B*). However, to avoid continuous mismatching, the algorithm measures the distance between the raw positioning point and the map-matched positioning point. If the distance is more than the allowable limit (a threshold value), then the algorithm reinitiates the map-matching process. In this case, for positioning point P_5, the algorithm reinitiates the process and chooses the true road segment. Two other thresholds (a distance threshold and a heading threshold) were also used to check whether a vehicle is near a junction or not. As mentioned before, in the tMM algorithm, the correct road segment selection at a junction is based on the total weight score (TWS) which is the sum of four weights: heading, proximity, link connectivity and turn restriction. Further, the relative importance of these weights varies with the operational environments. Therefore, any mistake in the identification of the operational environment may lead to an error in the total weight score. This may subsequently lead to wrong road link identification. Moreover, the threshold values used in the algorithm may influence the correct road segment identification.

After careful observation of each mismatching case due to errors in map-matching process, the following three strategies were identified to enhance the tMM algorithm:

1. re-examining the optimal weight scores using a Genetic Algorithm (GA) optimisation technique

2. using a lookup table to identify the weight scores corresponding to the operational environment (e.g. urban, suburban and rural)

3. re-estimating the thresholds used in the algorithm

In addition to these three strategies, it was also recognised that integrating a map-matching algorithm with a routing algorithm, which suggests a preferred route based on the shortest path or the lowest travel cost, may lead a better outcome in terms of correct link identification. However, a routing algorithm needs information on an origin (O) and a destination (D) of a trip. Since our objective is to enhance a generic MM algorithm which does not assume any O-D information, enhancing the tMM algorithm by a routing algorithm is not considered in this study.

Optimisation of the weight scores using a Genetic Algorithm (GA)

Previously, a gradient search method was used to determine the optimal values of weight scores used in the map-matching process (see Velaga et al., 2009). In the gradient search minimisation problem there is a possibility that the optimisation stops at a local rather than a global minimum (Michael et al., 2007). In order to ascertain whether the optimisation has reached a global minimum, Konar (2005) suggested to employ a more refined method such as a Genetic Algorithm (GA). Therefore, a GA-based optimisation algorithm is used to determine the relative importance of different wrights.

In this optimisation problem, the MM error (i.e., the percentage of wrong link identification) is assumed to depend on the weight coefficients for heading, proximity, connectivity and turn restriction. Here, the weight coefficients represent the relative importance of each weight. The objective function in the optimisation problem is formed by identifying the relationship between the MM error and the weight coefficients at each junction. To identify this relationship for each operational environment, a regression analysis is carried out by considering the MM error as a dependent variable and the four weight coefficients, their squares, inverse and interaction terms as independent variables. The functional relationship can be written as follows (see Velaga et al., 2009 for details):

$$\ln(MM_{error}) = \alpha + [\beta_{h1}H_w + ... + \beta_{t1}T_w] + [\beta_{h2}H_w^2 + .. + \beta_{t2}T_w^2] + \left[\frac{\beta_{h3}}{H_w} + .. + \frac{\beta_{t3}}{T_w}\right] + [\beta_{hd}(H_w D_w) + ... + \beta_{ct}(C_w T_w)] + \varepsilon_i \tag{1}$$

where

MM_{error} is the map-matching error and
α is an intercept term.
$\beta_{h1}, \beta_{h2}, ..., \beta_{t1}, \beta_{t2}, \beta_{t3}$ are the regression coefficients for heading, proximity, connectivity and turn restriction weights.
ε_i is the error term.
H_w, D_w, C_w, and T_w are the weight coefficients for heading, proximity, link connectivity and turn restriction respectively.

In the regression analysis statistically insignificant parameters were eliminated using a step-by-step backward elimination process. The final functional form with all statistically significant parameters is considered as the objective function in the optimisation problem.

In the previous optimisation test the sample size for the rural operational environment was low, containing only 40 junctions. Here this is increased to 186. A new objective function (i.e., the relationship between the map-matching error and the weight coefficients) for the rural operation environment is identified. A detailed description of the derivation of the objective function is provided in Velaga et al., 2009. The objective function for the rural operation environment is identified as:

$$\ln(MM_{error}) = 0.029H_w + 0.02D_w + 0.035C_w + 0.028T_w - 0.00079H_w D_w + 0.00037D_w T_w \tag{2}$$

The associated constraints are:

$$\begin{aligned} H_w + D_w + C_w + T_w &= 100 \\ 1 \le (H_w, D_w, C_w, T_w) &\le 100 \end{aligned} \tag{3}$$

This is a minimisation problem in which minimising the map-matching error is the objective.

The adjusted R^2 value of the above model (equation 2) was found to be 0.98. For the other two operational environments (i.e., urban and suburban), optimisation functions are the same as the functions provided in Velaga et al. (2009).

To re-estimate the optimal weight scores, the Matlab GA toolbox is used (Russel and Norvig, 2002; Callan, 2003; Karray and DeSilva, 2004; MathWorks, 2008). In the GA optimisation process, a population size of 20, which were uniformly distributed with lower range of 1 and higher range of 100 were used. Stopping criteria is selected as 5,000 generations. After approximately 1,500 generations, the function value (fitness value) shows that the function achieves the global optimal values. The optimal values of heading, proximity, connectivity and turn restriction weight scores for the three operational environments, using the gradient search method and a Genetic Algorithm, are illustrated in Table 3.

Table 3 - Optimal weight scores using gradient search method and GA

Weight coefficient	Gradient search method			Genetic Algorithm		
	Urban	Suburban	Rural	Urban	Suburban	Rural
H_w	39.99	46.24	44.48	37.15	46.42	42.37
D_w	8.13	44.99	53.52	8.06	43.76	55.63
C_w	36.4	4.46	1	35.85	4.29	1
T_w	15.48	4.31	1	18.94	5.53	1

It is noticeable from Table 3 that the weight scores from the two optimisation techniques are very similar. But, there is a slight difference in H_w values in urban and rural areas and D_w values in suburban and rural, C_w value in urban area and T_w value in urban and suburban areas. However, the optimal weight scores obtained using the GA are more reliable as those are solved as a global optimisation problem. Therefore, the second set of optimal values (from GA) is included in the enhanced tMM algorithm.

Use of a lookup table to identify the operational environment

Table 3 shows that the relative importance of the weights varies with the operational environment. The algorithm should identify the operational environment in which the vehicle is travelling, and should select the corresponding weights from the weight matrix shown in Table 3. The identification of the operational environment (whether the vehicle is in an urban, suburban or rural area) can be based on the complexity of road network, land-use data, building height data, etc. For instance, the road network in an urban area is denser (i.e., more junctions and roads per unit area) than that of in a suburban or a rural area. Since, land-use data and building height data are not easily available, the identification of operational environment can be determined by a threshold which is a function of the total length of the road network and the number of junctions per unit area:

$$T_{OE} = f(L, N) \tag{4}$$

where,

T_{OE} is the threshold for operational environment,

L is the total length of road network (in km) within a given area and
N is the number of junctions in that area.

An empirical analysis was conducted, using a national level GIS road network, to identify the T_{OE} which can be used to detect the operational environment on which a vehicle is travelling. Firstly, using the UK Google Earth map, a sample of rural areas was identified in the network. Random points are selected within the road network of rural area, and a circle of radius 200m is drawn around each of the points. After performing a sensitivity analysis a circular area of radius 200m was found to be successfully established the threshold for the identification of operational environment. The total length of all road segments (L) and the number of junctions (N) within that circular area are calculated. This procedure is repeated for 300 different random points in the network. This procedure provides a set of L and N for that particular operational environment. A factor, which is the ratio of N and L, is identified and its mean (μ) and standard deviation (σ) are calculated. The same procedure is repeated for the urban and suburban environments. The means and standard deviations of the factor for urban, suburban and rural operational environments are shown in Figure 5. Here, μ_U, μ_S and μ_R are the mean values and σ_U, σ_S and σ_R are the standard deviations of the factor for urban, suburban and rural operational environments respectively.

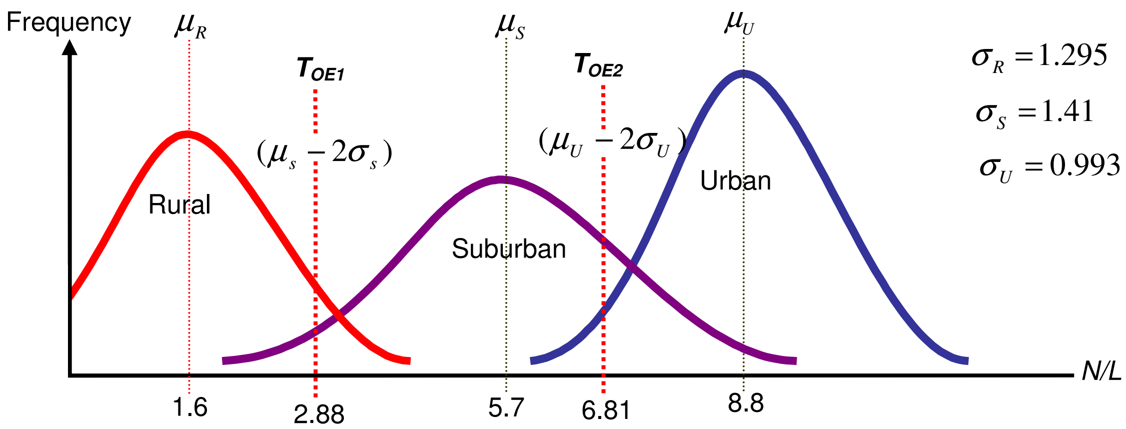

Figure 5 - Thresholds for operational environment identification

As stated before, the identification of an operational environment is critical to a MM algorithm. Figure 5 shows that the values of the means and standard deviations of the N by L ratios for different operational environments are over-lapped meaning that it is not easy to derive threshold values for the identification of an operational environment. In the case where a vehicle travels along a mixed urban setting (i.e. partly urban and partly suburban), the algorithm should recognise that the vehicle is in an urban area so that more stringent weight coefficients are selected for the map-matching process. This is also true for the case of mixed suburban area (i.e. partly suburban and partly urban) in which the weight coefficients for the suburban area should be employed.

Assuming the ratio of *N* and *L* follows a normal distribution and based on the above argument, two threshold values (T_{OE1} and T_{OE2}) are identified as:

$$T_{OE1} = \mu_S - 2\sigma_S \tag{5}$$

$$T_{OE2} = \mu_U - 2\sigma_U \tag{6}$$

if $T_{OE} \le T_{OE1}$ then it is assumed that the operational environment is *rural*
if $T_{OE1} < T_{OE} < T_{OE2}$ then it is assumed that the operational environment is *suburban*
if $T_{OE} \ge T_{OE2}$ then it is assumed that the operational environment is *urban*

Where, T_{OE} is the calculated threshold in the map matching process.

The mean of the factor for urban, suburban and rural operational environments are identified as 8.8, 5.7 and 1.6 respectively; and the corresponding (σ) values are 0.993, 1.41 and 1.29 respectively. The T_{OE1} and T_{OE2} were found to be 2.88 (i.e., 5.7-2*1.41) and 6.81 (i.e., 8.8-2*0.99). In the map-matching process if the calculated threshold (T_{OE}) using the same area (i.e., 200m radius circle), is less than T_{OE1} (i.e., 2.88) a vehicle is in a rural area; if it is more than T_{OE2} (i.e., 6.81) vehicle is in an urban area, if it is between these two values then the vehicle is in a suburban area.

Checking threshold values used in the algorithm

In the topological MM algorithm three different threshold values are used. They are distance threshold (D_t), heading threshold (H_t) and a threshold value for a consistency check (C_t). The former two threshold values are used to identify whether the vehicle is near a junction. The tMM algorithm checks whether a vehicle reaches a junction using two criteria: (1) checking distance from the previously map-matched vehicle position to the downstream junction; in which a distance threshold (D_t) is used. (2) checking the vehicle heading with respect to the previously matched link direction; in which a heading threshold (H_t) is used. The third threshold value is used in a consistency check (i.e., whether the distance between the raw position point and the map-matched position on the link is large). Every time the algorithm checks the distance between the raw positioning point and the map-matched positioning point. If it exceeds a certain limit (i.e., the threshold value) then the algorithm re-initiates the process. Previously, these three thresholds (D_t, H_t and C_t) were identified, using 1800 positioning points, by manually checking whether the algorithm selects 'map-matching at junction' process when the vehicle reaches a junction and whether the algorithm can recognise the continuous mismatches in order to reinitiate the process to identify the correct link. D_t, H_t and C_t thresholds were identified as 20, 5 and 40 respectively. Now, as part of the algorithm correction process, thresholds (D_t, H_t and C_t) were re-estimated using a positioning data set of 2,814 positioning points collected in Central London (part of data set 1 in Table 1). An experiment was conducted with different possible threshold values and its corresponding percentage of correct link identification was measured. The D_t, H_t and C_t

values with minimum error in correct link identification are identified as 23, 5 and 37 respectively.

PERFORMANCE OF THE ENHANCED TMM ALGORITHM

An independent dataset (sample size 5,256 positioning points) collected in and around Nottingham, UK was used to re-evaluate the performance of the enhanced map-matching algorithm. This positioning data is a part of dataset 4 in Table 1. A reference (true) trajectory was obtained from a carrier phase GPS receiver integrated with a high-grade Inertial Navigation System (INS). Accuracy of this equipment (carrier-phase GPS/INS) was found to be better than 5 centimetres over 97.5 percent of the time in all three coordinate components (Aponte et al., 2009). The total length of the test trajectory is 55.9 km.

The improvement in the correct road link identification also affects the horizontal accuracy. The highly accurate positioning data from carrier phase GPS/INS enabled us to check the algorithm's horizontal positioning accuracy. The algorithm's performance for each enhancement strategy, with respect to the original (base) tMM algorithm is shown in Table 4.

Table 4 – Enhanced algorithm performance

Enhancement	% of correct link identification	Horizontal accuracy (m)		Along-track error (m)		Cross track error (m)	
		Average	SD	Average	SD	Average	SD
Base tMM algorithm	96.5	4.33	2.83	2.16	1.74	3.29	2.86
1: New weight scores	96.7	4.31	2.78	2.14	1.69	3.28	2.85
2: Lookup table	97.7	4.20	2.48	2.12	1.53	3.20	2.60
3: Threshold values	96.5	4.33	2.83	2.16	1.74	3.30	2.87
1 and 2	97.8	4.19	2.47	2.11	1.52	3.19	2.59
1 and 3	96.7	4.31	2.79	2.15	1.69	3.29	2.85
2 and 3	97.7	4.20	2.48	2.12	1.53	3.20	2.60
1, 2 and 3	**97.8**	**4.19**	**2.47**	**2.11**	**1.52**	**3.19**	**2.59**

SD- Standard deviation

The original algorithm correctly identifies the road links 96.5% of the time. However, when all three improvements are included in the final algorithm which increases the correct link identified to 97.8%. An improvement of 1.3% in correct road link identification is noticed. The second improvement (i.e., using a lookup table to identify the operational environment) contributes most to the improvement in the algorithms performance. The first enhancement (i.e., re-examining the optimal weight scores using a Genetic Algorithm) slightly improves the algorithm and the third enhancement (re-estimating the thresholds used in the algorithm) did not contribute in improving the algorithm performance. The horizontal accuracy of the enhanced algorithm is identified as 9.1m $(\mu+2\sigma)$ with along track and cross track errors as 5.2 m $(\mu+2\sigma)$ and 8.4m $(\mu+2\sigma)$ respectively.

CONCLUSIONS

An improvement process of a weight-based topological map-matching algorithm was presented in this paper. The enhancement process included: mismatching detection, improvement strategies identification, algorithm modification and performance re-evaluation. After map-matching using extensive positioning data sets, all mismatches due to positioning data errors, digital road map and map-matching process were identified. In positioning data sets 1 and 2 (for which the digital maps are good) about 50% of the wrong road link identification was due to the map-matching process. After further examining the mismatches due to the map-matching process, three strategies were identified to enhance the tMM algorithm. They were:

1. re-examining the relative importance of weight scores using a Genetic Algorithm tool,
2. using lookup table for operational environment identification,
3. re-estimating threshold values

The performance of the algorithm was re-evaluated using an independent positioning data. The enhanced algorithm succeeded 97.8% of the time in correct link identification with an horizontal accuracy of 9.1 m $(\mu + 2\sigma)$. Before the enhancement the success rate was 96.5% with 10.0m $(\mu + 2\sigma)$ horizontal accuracy. This suggests that the proposed modifications were rational as they improves the performance by 1.3% in correct link identification and 0.9m horizontal accuracy.

REFERENCE

Aponte, J., X. Meng, T. Moore, C. Hill and M. Burbidge (2009). Assessing Network RTK Wireless Delivery, GPS World, 20(2), 14-27.

Callan, R., (2003). Artificial intelligence, Ashford colour press, Gosport. (ISBN: 0333801369).

Goodwin, C and J. Lau (1993). Vehicle Navigation and Map Quality. Proceedings of the IEEE-IEE Vehicle Navigation & Information Systems Conference, Ottawa, 17-20.

Greenfeld, J.S. (2002). Matching GPS observations to location on a digital map. In: Proceedings of the 81st Annual Meeting of the Transportation Research Board, Washington, D.C.

Kaplan, E. D and C. J. Hegarty (2006). Understanding GPS: Principles and Applications, Artech House Inc, London. (ISBN 1580538940).

Karray, F, O., and C. De Silva (2004), Soft Computing and Intelligent Systems Design: Theory, Tools and Applications, Addison-Wesley, Essex, England. (ISBN:10-0321116178).

Kim, W., G. Jee and J. Lee (2000). Efficient use of digital road map in various positioning for ITS. IEEE Symposium on Position Location and Navigation, San Diego, CA, 170–176.

Konar, A., (2005). Computational intelligence: Principles, Techniques and Applications, Springer Berlin, New York.

Li, X., H. Lin, Y. Zhao (2005). A connectivity based map-matching algorithm. Asian Journal of Geoinformatics, 5(3), 69-76.

MathWorks, 2008, Genetic Algorithm and Direct Search toolbox user's guide. The Mathworks inc., Michael, C.F., O. L. Mangasarian and S. J. Wright (2007). Linear Programming with MATLAB. MPS-SIAM series on optimization, Inc., Philadelphia. (ISBN: 9780898716436)

Quddus, M.A., W. Y. Ochieng, L. Zhao and R. B. Noland, (2003). A general map matching algorithm for transportation telematics applications. GPS Solutions, 7(3), 157–167.

Quddus, M.A., W. Y. Ochieng, and R. B. Noland (2007). Current Map Matching Algorithm for Transport Applications: State-of-the Art and Future Research Direction. Transportation Research Part-C: Emerging Technologies, 15, 312 -328.

Russel, S., and P. Norvig (2002), Artificial Intelligence: A Modern Approach, Prentice Hall, New Jersey.

Taylor, G and G. Blewitt (2006). Intelligent positioning: GIS-GPS Unification. John Wiley publishers, west Sussex, England.

Velaga, N. R., M. A. Quddus and A. L. Bristow (2009). Developing an Enhanced Weight Based Topological Map-Matching Algorithm for Intelligent Transport Systems. Transportation Research C: Emerging Technologies. 17(6), 672-683.

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(1), 91–108.