DOI: 10.1111/tgis.12958

RESEARCH ARTICLE

Transactions in GIS WILEY

FLCSS: A fuzzy-based longest common subsequence method for uncertainty management in trajectory similarity measures

Faraz Boroumand[1] | Ali Asghar Alesheikh[1] | Mohammad Sharif[2] | Mahdi Farnaghi[3]

[1]Department of Geospatial Information Systems, K. N. Toosi University of Technology, Tehran, Iran

[2]Department of Geography, Faculty of Humanities, University of Hormozgan, Bandar Abbas, Iran

[3]Department of Geo-Information Processing (GIP), Faculty of Geo-Information Science and Earth Observation (ITC), University of Twente, Enschede, The Netherlands

Correspondence
Mohammad Sharif, Department of Geography, Faculty of Humanities, University of Hormozgan, Bandar Abbas, Iran.
Email: m.sharif@hormozgan.ac.ir

Abstract

The large quantity of movement data collected from various sources can be inherently uncertain and heterogeneous. In the movement data analysis and mining spectrum, computing the similarity of trajectories while considering the uncertainty and heterogeneity has been less addressed. Generally, two factors of sampling and positioning error cause uncertainty in trajectory databases. Therefore, in this research, a method based on the longest common subsequence (LCSS), named FLCSS, is proposed that uses fuzzy theory and the bead model to consider the uncertainty of trajectories originated from positioning and sampling errors. The performance of FLCSS is evaluated by implementations on real and synthetic datasets, and compared with six important and commonly used similarity measurement methods, namely, LCSS, edit distance on real sequence (EDR), dynamic time warping (DTW), edit distance with real penalty (ERP), Hausdorff distance (HD), and Fréchet distance (FD). The results show that FLCSS has a better performance compared to other methods, in terms of sensitivity to point displacement, noise, and different sampling rates. Furthermore, the high correlation between FLCSS and LCSS ($\rho = 0.91$) confirms the robustness of the proposed method in considering uncertainty in the trajectory databases.

1 | INTRODUCTION

The history of each moving point object can be recorded in the form of a sequence of locations for each step, called a trajectory (Miller & Han, 2009). Several approaches have been introduced to extract information and discover geographic knowledge from large sizes and highly complex trajectory datasets (Mennis & Guo, 2009). One of the most fundamental issues in geographic knowledge discovery from trajectory datasets is similarity measurement (Xie et al., 2017). This procedure identifies the commonalities in the behaviors of moving objects and extracts the similarity degrees among them. The results of similarity measurement can be employed in movement prediction, pattern recognition, trajectory classification, and trajectory clustering of moving entities (Alizadeh et al., 2021a). Due to the complex nature of the movement, various methods for measuring trajectory similarity have been proposed to date (Dodge et al., 2012).

Owing to the progress in navigation and tracking technologies, large numbers of trajectory datasets are being collected in various domains. In many cases, the Global Positioning System (GPS) is used to track moving objects, such as humans (Goudarzi et al., 2022), cars (Lehmann et al., 2019), airplanes (Sharif & Alesheikh, 2017), bicycles (Sharif et al., 2019), and animals (Buchin et al., 2014). Other means of collecting trajectory data include Radio Frequency Identification (RFID), Automatic Identification System (AIS) for tracking vessels (Alizadeh et al., 2021b), satellite images for tracking hurricanes (Dodge et al., 2012), video tracking to record pedestrians movement (Zaki & Sayed, 2018), and Location-Based Social Networks (LBSNs) to record users' activities in a sequence of check-ins (Karami & Malek, 2021). Consequently, movement data are heterogeneous and uncertain because they are collected by various devices and from sources with different accuracies in different domains (Kaffash-Charandabi et al., 2019; Sharif & Alesheikh, 2018).

Generally, uncertainty refers to any aspect of the problem that cannot be confidently commented on. Uncertainty, therefore, includes independent concepts such as imprecision, incompleteness, and ambiguity (Motro, 1993). In the study of moving objects, uncertainty is commonly related to imprecision in the acquisition and presentation of movement in the form of trajectories (Pfoser & Jensen, 1999). In this sense, uncertainty in collecting movement data leads to uncertainty in similarity measurement of trajectories and other possible subsequent processes.

According to Pfoser and Jensen (1999), the factors that cause uncertainty in trajectories can be divided into two main categories. (1) *Positioning error*: determining the position of a moving object is always accompanied by error due to the limitations of positioning systems. For example, the location provided by GPS, as the most widely used positioning system, is associated with many error sources, such as multi-path error, atmospheric effect, and hardware delay to name a few. All these factors lead to a difference between the observed object location and its actual location, which ranges from several centimeters to meters based on the type of positioning system. Therefore, these factors are classified into a single category entitled positioning error. Furthermore, because of the presence of obstacles or weak signals in some areas, positioning errors may vary over time. This problem increases uncertainty in methods such as longest common subsequence (LCSS) that use a fixed threshold. (2) *Sampling error*: movement is a continuous phenomenon that is modeled discretely in the form of a trajectory. Therefore, the way the object moves between the corresponding sampling points is not acquired. To represent a trajectory, the gap between sampling points is filled by linear interpolation (Guo et al., 2019). Accordingly, lower sampling rate results in larger gap areas. Changing the sampling rate changes the shape of the trajectory, which increases the uncertainty in representing a trajectory, and thus in the measurement of similarity between trajectories. In trajectory similarity measurement, both aspects have been less considered, while uncertainty is inevitable in movement studies (Laube & Purves, 2011).

A robust solution to manage uncertainty is fuzzy logic. It provides the ability to represent human knowledge about uncertain situations in a unified manner (Yager, 2002). Meanwhile, it does not require complex mathematical models and exact values of input parameters (Zadeh, 2011). Accordingly, the main contribution of this research is to present a method based on fuzzy logic to manage both positioning and sampling errors in measuring

trajectories similarity. To consider the positioning error, the matching degree of points is determined using a fuzzy membership function. To consider the sampling error, the bead model is applied on the trajectories.

The remainder of this article is organized as follows: In Section 2, basic concepts and prerequisites are explained. Moreover, a review of some existing works related to the subject of this article is provided. In Section 3, the proposed similarity measure is introduced. Section 4 presents the evaluation experiments, results, and discussion. Section 5 concludes the article and outlines potential future works.

2 | PRELIMINARIES AND REVIEW OF LITERATURE

2.1 | Uncertainty management in trajectory databases

There are multiple models that consider uncertainty in trajectory databases, including bead, cylindrical, grid, network-based, and probability-based models. In the bead model, also known as a space–time prism, a trajectory is represented as a chain of ellipses (Kuijpers & Othman, 2010). The dimensions of each ellipse are determined according to different features of the object movement, such as speed, acceleration, and direction of the moving object (Guo et al., 2019). The reason for using this type of geometry is that the movement of the object between each pair of consecutive points of the trajectory is restricted by its maximum speed. As a result, the maximum distance of the object to the two consecutive points of the trajectory cannot exceed a limit, which is reminiscent of an ellipse in two dimensions and a bead in three dimensions (Hornsby & Egenhofer, 2002). Indeed, each ellipse or bead expresses all possible positions that the moving object may have passed through (Kuijpers et al., 2017). The disadvantage of this model is that it only considers the uncertainty caused by a low sampling rate. The dimensions of the ellipses are also affected by the error of observation and are uncertain. Shi et al. (2021) proposed models based on error ellipses that take into account the uncertainty caused by both sampling and measurement errors. To quantify the size of uncertainty ellipses, the Minkowski distance metric was applied. However, the models do not apply to trajectories with large temporal granularity. In the cylindrical model, trajectories are represented as a cylindrical volume and the radius is defined using a threshold by the user (Trajcevski et al., 2004). The advantage of the cylindrical model over the elliptical model is that it is suitable for the uncertainty caused by the positioning error (Frentzos et al., 2009). However, the primary defect of this model is that trajectories are usually collected by different people with different tools and in different conditions. Therefore, the accuracy and sampling rate of each is different from the other, while the uncertainty threshold in this model has a constant value (Jeung et al., 2014). In the grid model, space is divided into a set of cells, and a trajectory is considered as a sequence of cells through which it passes (Pelekis et al., 2009). Similar to the cylindrical model, the main problems in this model are the constant size of the cells and finding the optimum cell size. In the network-based models, the main idea is that the movement of objects is limited to the network space. Therefore, the position of the moving object can be determined any time if it moves at a constant speed between the two sample points (Chen et al., 2015; Zheng et al., 2011). The limitation of these models is that they are only applicable to objects moving in the network space, such as cars and trains. The Evolving density trajectory (Jeung et al., 2014), as a Probability-based model, uses Gaussian distribution to represent an uncertain trajectory, where the mean value and standard deviation are, respectively, considered as the real position and the degree of uncertainty. The fact that each point needs different parameters for the Gaussian distribution—which is not trivial to determine—complicates the Evolving density model.

2.2 | Trajectory similarity measures

A trajectory is a kind of time series. The methods for measuring the similarity of time series such as dynamic time warping (DTW; Berndt & Clifford, 1994), edit distance with real penalty (ERP; Chen & Ng, 2004), edit distance on real

sequence (EDR; Chen et al., 2005), and longest common subsequence (LCSS; Vlachos et al., 2002) have been used to measure the similarity between trajectories. Fréchet distance (FD; Eiter & Mannila, 1994) and Hausdorff distance (HD; Alt & Guibas, 2000) are among common geometric similarity measures that are extended for trajectories.

It is evident that none of these methods is completely superior to the other ones. All the methods mentioned above can be used for trajectories of different lengths and support local time shifting. Although DTW, ERP, Fréchet, and Hausdorff distances are said to be noise sensitive (Alt & Guibas, 2000; Kurbalija et al., 2014), LCSS and EDR perform better due to the use of a threshold to determine whether two points match or not (Toohey & Duckham, 2015). On the other hand, setting the appropriate threshold is difficult and application dependent. In addition, LCSS does not consider the distance between two matching points (Chen et al., 2005). Most importantly, none of these methods take into account the uncertainty of the trajectories. More details of the above-mentioned methods are presented in the Appendix.

2.3 | Uncertainty management in trajectory similarity measures

Methods have been developed to overcome the inherent uncertainty in the trajectory datasets and consequently in trajectory similarity results. Wang et al. (2015) have introduced the expected Euclidean distance (EE distance) to estimate the similarity between two uncertain trajectories. EE distance is based on the Euclidean distance and Evolving density model. Implementing the Euclidean distance is simple and has low computational complexity. However, it can only measure the similarity of trajectories of the same length and does not support local time shifting. Naderivesal et al. (2019) proposed the trajectory interval distance estimation (TIDE) method for measuring the similarity of trajectories using maximum estimated velocity. TIDE describes the similarity of two trajectories using an interval instead of a number. However, it only addresses the uncertainty caused by the sampling error. In other research, Furtado et al. (2018) introduced the uncertain movement similarity (UMS) method that uses a modified elliptical model. Unlike the original model based on maximum velocity to determine the size of ellipses, UMS proposes an approximate upper bound distance to determine the size of ellipses. Finally, the similarity is calculated based on three criteria of alikeness, sharpness, and continuity of two elliptical trajectories. Similarly, the uncertain trajectory similarity measure (UTMS) method uses a modified elliptical model called the Amended Ellipse model. In addition to sampling error, UTMS considers positioning error, but it relies on a constant value for positional error (Guo et al., 2019). In another attempt, Sharif et al. (2019) proposed a hierarchal fuzzy inference system, named CaFIRST, to measure the similarity between trajectories enriched by contextual information. This method is able to include both quantitative and qualitative contextual information. However, it suffers from considering uncertainties due to the sampling error.

From all the aforementioned methods, different approaches have been adopted to model uncertainty in trajectory similarity measurement. However, they are unable to simultaneously consider the uncertainty caused by both positioning and sampling errors. In this article, we have alleviated the drawbacks of the previous methods using the fuzzy approach.

3 | FUZZY-BASED LONGEST COMMON SUBSEQUENCE METHOD

Our methodology consists of three main stages as shown in Figure 1: (1) Data preparation; (2) the proposed fuzzy-based longest common subsequence method (FLCSS); and (3) experimental evaluation of FLCSS on real and synthetic datasets.

In the following, the calculation of trajectories similarity using LCSS is initially explained. Then, in two steps, the drawbacks of LCSS in coping with uncertainty and the procedure of FLCSS for considering the uncertainties caused by both positioning and sampling errors are explained in detail.

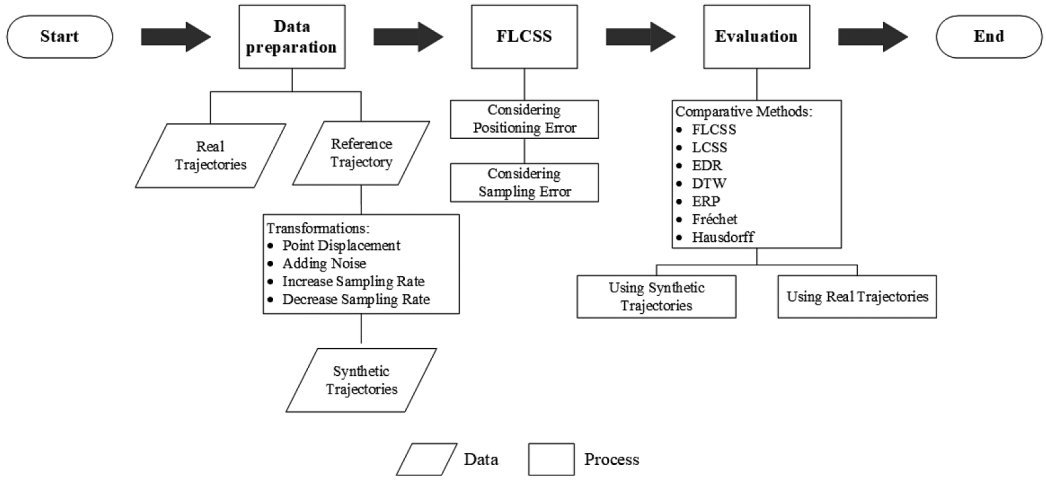

FIGURE 1 Research workflow

The reason why the LCSS is chosen for development is that, first, it can be easily implemented using dynamic programming. Second, some points can stay unmatched, which makes the LCSS more accurate and much more robust to outliers compared with methods like DTW. However, this feature increases the uncertainty caused by sampling error. Therefore, seemingly after some modifications in the LCSS method, it can overcome the competing methods. The LCSS between two trajectories A and B with lengths m and n and with the threshold ε is calculated by Equation (1), where $\text{Head}(A) = \{a_1, a_2, \ldots, a_{m-1}\}$ and $|a_m - b_n|$ is the Euclidean distance between a_m of trajectory A and b_n of trajectory B (Vlachos et al., 2002).

$$LCSS_\varepsilon(A,B) = \begin{cases} 0 & m=0 \text{ or } n=0 \\ 1+LCSS_\varepsilon(Head(A), Head(B)) & |a_m - b_n| \le \varepsilon \\ \max\begin{cases} LCSS_\varepsilon(Head(A), B) \\ LCSS_\varepsilon(A, Head(B)) \end{cases} & |a_m - b_n| > \varepsilon \end{cases} \quad (1)$$

3.1 | Step 1: Considering positioning error

Positioning error can lead to vagueness when comparing the points of two trajectories using the LCSS method. The problem is determining whether the two points are matched or not. Figure 2 shows the situations of four points $b_1, b_2, b_3, b_4 \in B$ compared to $a_1 \in A$. In LCSS (Figure 2a) that uses the threshold, two points can have three situations with respect to each other. (1) The distance from a_1 to b_4 is very high, and it is clear that these two points do not match; therefore, they have no effect on the similarity size of two trajectories. (2) b_3 is ambiguous with respect to a_1 because the distance between them is approximately equal to the threshold, and considering the positioning error, it is not possible to say whether the two points a_1 and b_3 match or not. (3) The distance between both b_1 and b_2 to a_1 is less than the threshold and the situation of both of them is considered the same with respect to a_1. Although the distance between a_1 and b_1 is less than the distance between a_1 and b_2, both points have the same effect on the similarity of the two trajectories.

In the FLCSS (Figure 2b), instead of using the threshold, the matching degree of two points is defined using a fuzzy membership function. The advantage of using the fuzzy membership function over the threshold is that, first, the points can match to some extent; therefore, the status of all points in relation to each other can be defined based on the application. Second, due to the continuous nature of the fuzzy membership function, the

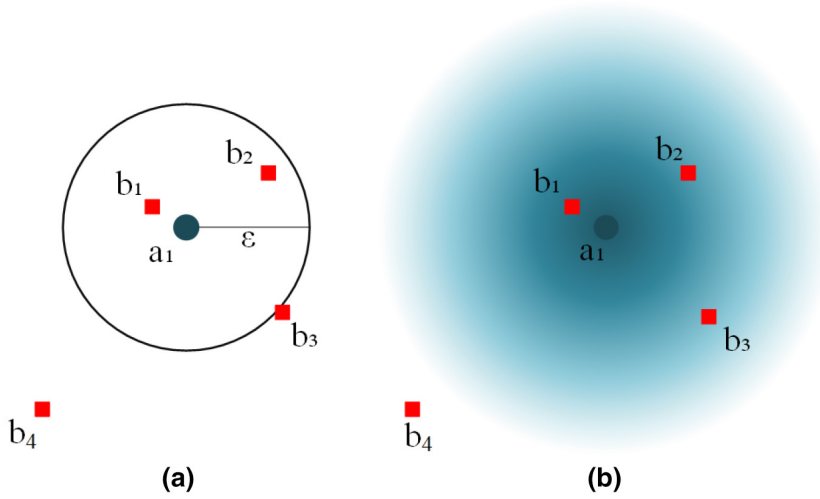

FIGURE 2 The probable position of the sampling points of two trajectories A and B relative to each other: (a) using a threshold; and (b) using a fuzzy membership function

positioning error will have little effect on the matching degree of the points and the overall similarity of the two trajectories. FLCSS is calculated by Equation (2) where the threshold with the fuzzy membership function ($\mu[|a_m - b_n|]$) is replaced.

$$FLCSS_{\mu}(A,B) = \begin{cases} 0 & m=0 \text{ or } n=0 \\ \max \begin{cases} \mu(|a_m - b_n|) + FLCSS_{\mu}(Head(A), Head(B)) \\ FLCSS_{\mu}(Head(A), B) \\ FLCSS_{\mu}(A, Head(B)) \end{cases} & \text{otherwise} \end{cases} \quad (2)$$

3.2 | Step 2: Considering sampling error

For each sampling point in the LCSS, a maximum of one corresponding point is considered. Therefore, due to the difference in sampling rates between two trajectories, some points may have no corresponding points (Vlachos et al., 2002). From now on, we refer to such points as non-sampled points (NSPs). This feature makes LCSS less sensitive to noise compared with DTW where all points must be matched. On the other hand, this feature increases its sensitivity to the sampling rate. To overcome this problem, the proposed method is inspired by the bead model.

The question is "how much each NSP affects the similarity of the two trajectories?" According to UTMS method, if the point is placed inside the ellipse, that point is a matching point and its degree of matching is equal to one (Guo et al., 2019). According to the first step, due to the positioning error, determining whether the desired point is inside the ellipse or not is accompanied by uncertainty. In addition, the ellipse dimensions are determined based on the movement parameters of the object, such as the maximum velocity, which is less accurate. Therefore, a fuzzy membership function can be used to calculate the degree of matching of the NSPs (see Figure 3). Equation (3) shows how to calculate the matching degree of a NSP, where b_i is considered as NSP.

$$Match(A, b_i) = \mu\left(\left|b_i - a_j\right| + \left|b_i - a_{j+1}\right| - \left|a_{j+1} - a_j\right|\right) \quad (3)$$

Finally, the overall similarity of two trajectories is calculated using Equation 4. In this respect, FLCSS adds only one unit to the similarity for two points that match. However, for each NSP that matches, a unit is added to the similarity.

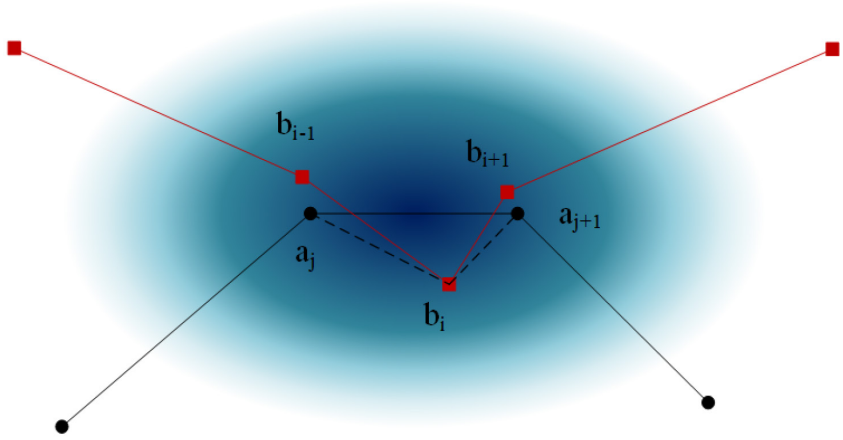

FIGURE 3 Example of calculating the degree of matching of a NSP

Therefore, FLCSS is multiplied by two, and finally the overall similarity is normalized to the sum of lengths of two trajectories to be independent of the lengths.

$$FLCSS(A,B) = \left(2^{*}\, FLCSS_{\mu}(A,B) + \sum_{i}^{NSP\in B} Match\left(A,\, NSP_i\right) + \sum_{j}^{NSP\in A} Match\left(B,\, NSP_j\right)\right)/(m+n) \tag{4}$$

The performance of FLCSS is compared with six important and commonly used similarity measurement methods including LCSS, EDR, DTW, ERP, FD, and HD. It should be noted that the results of different methods except FLCSS, FD, and HD are normalized using larger trajectory length to be independent of size. In addition, to make a uniform comparison, the values obtained from DTW, ERP, FD, and HD are converted into similarities using the equation $S = 1 - e^{-1/D}$ (Guo et al., 2019), and the results of EDR are reported based on the equation $S = 1 - D$ (Vlachos et al., 2002). In these two equations, D is the calculated distance using the desired method and S is the similarity value.

To evaluate the validity of the FLCSS results, the Pearson correlation coefficient is calculated between sets of similarity results obtained from different methods (Equation 5). ρ is the Pearson correlation coefficient, x_i and y_i are the calculated similarities, and $\bar{x}$ and $\bar{y}$ are the means of similarity sets from two methods.

$$\rho = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \tag{5}$$

4 | RESULTS AND DISCUSSION

To evaluate the proposed method and investigate its effectiveness in measuring the similarity of trajectories, two series of experiments are implemented on synthetic and real datasets. The first series of experiments on the synthetic data evaluate the FLCSS's sensitivity to noise, displacement, and different sampling rate values. The second series of the experiments are conducted using real data that compare the correctness and efficiency of FLCSS with other similarity measurement methods. All methods are implemented in Java (JDK 15.0.1) and the experiments were carried out on an Intel(R) Core(TM) i7-4710HQ CPU @ 2.50 GHz with 16 GB RAM and Windows 10 operating system.

4.1 | Dataset

The Beijing GPS trajectory dataset collected by the Geolife Microsoft Project (Zheng et al., 2009) is a rich movement dataset (Figure 4). It contains 18,670 trajectories recorded over 5 years and includes different life routines and entertainment activities of 182 users (e.g., walking, cycling, and driving). The heterogeneity in the trajectories makes it a suitable source for our experiments. In all, 743 trajectories composed of 1,111,175 sampling points are selected for evaluation. In addition, the sampling rate for selected trajectories is between 1 and 5 s. Pre-processing steps such as outlier detection and map matching reduce the amount of error and uncertainty. Therefore, they are avoided in the performance evaluation of the proposed and competing methods.

Similar to the approach proposed by Su et al. (2020), a real trajectory is selected from the Geolife dataset as the reference trajectory to create synthetic trajectories. The reference trajectory should be long enough and with a relatively high sampling rate. The synthetic trajectories are then generated by changing the reference trajectory using four transformation functions introduced as follows. Finally, the similarities of the synthetic trajectories to the reference trajectory are calculated. In this scenario, the changes applied to the reference trajectory can be controlled using the parameters of the transformation functions. Therefore, the sensitivity of the methods to the changes can be measured in a controlled way (Furtado et al., 2016). These four functions are:

1. Displacement: In this transformation, several reference sampling points are randomly selected and each, as determined by the distance (d) parameter, moves in a random direction. The number of points moved in this transformation is determined using the change rate (r) parameter.
2. Noise addition: In this transformation, new points, the number of which is determined using the change rate, are added to the reference trajectory. The gap between these points and reference trajectory ones is determined using the distance parameter.
3. Sampling rate increase: In this transformation, using the change rate parameter, the required number of points is determined and added to the reference trajectory. Unlike the noise addition, new points are placed between the reference trajectory points in this transformation.
4. Sampling rate decrease: This transformation is similar to the sampling rate increase, except that the selected points are removed.

To show the methods' sensitivities to any change applied to the reference trajectory, we have defined two criteria α and β in Equations (6) and (7). The first criterion indicate sensitivity to the rate of changes and the second

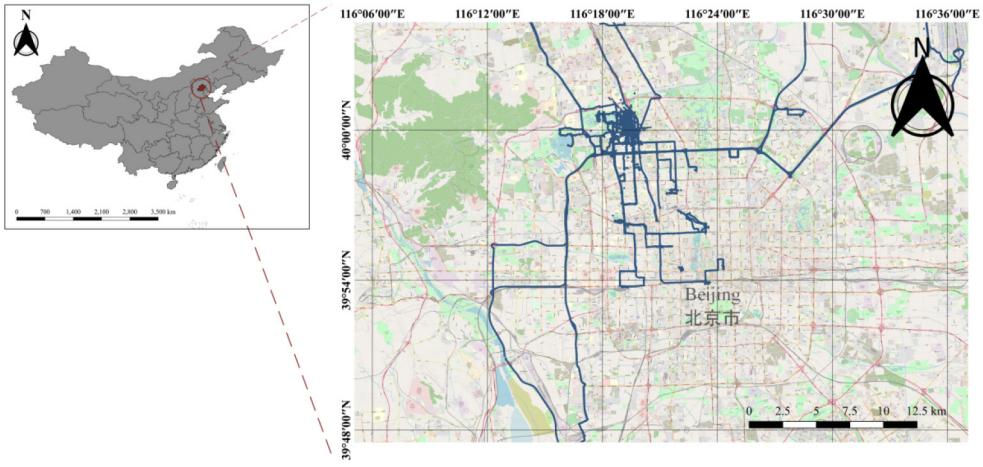

FIGURE 4 Study area: Selected Geolife trajectories in Beijing metropolitan area

one indicates sensitivity to the distance of changes. Larger α and β values indicate higher sensitivity. In these equations, S_1 is the similarity between the reference trajectory and a transformed trajectory, which is created according to a set of parameters. S_2 is the similarity between the reference trajectory and another transformed trajectory, which is created according to another set of parameters. r_1 and r_2 are, respectively, the rates of changes in the first and second transformed trajectories. Similarly, d_1 and d_2 are the distance of changes in those trajectories. For example, if we add 20% noise to the reference trajectory with a constant distance, and the similarity is reduced by 30% in the first method and 10% in the second method, the α values for the first and second methods are 1.5 and 0.5, respectively. This result indicates that the first method is more sensitive to noise.

$$\alpha = \frac{(S_2 - S_1)/S_1}{(r_2 - r_1)/r_1} \tag{6}$$

$$\beta = \frac{(S_2 - S_1)/S_1}{(d_2 - d_1)/d_1} \tag{7}$$

4.2 | Implementation

4.2.1 | Experiment 1: Synthetic dataset

This experiment aims at the evaluation of the FLCSS and competing methods in terms of sensitivity to point displacement, noise addition, and changing the sampling rate. In this experiment, synthetic trajectories are created using the above-mentioned transformation functions, and their similarities to the reference trajectory are calculated. The reference trajectory is 412 points long, and the sampling rate is 5 s (Figure 5). For each transformation and series of parameters, 100 new trajectories are created. The parameters used for these transformations are listed in Table 1.

A challenging issue in the FLCSS is determining the relationship between the distance of points and their matching degree, which depends on the dataset and application. However, it can be mentioned that the smaller the distance between the points, the higher the matching degree, and vice versa. Accordingly, different types of membership functions can be used. FLCSS was tested with three commonly used fuzzy membership functions including trapezoidal, parabolic, and reversed parabolic. The results show that there is no particular priority in choosing the type of membership function. Accordingly, the trapezoidal membership function (Figure 6) is selected, which is simple and fast to calculate. Parameters a and b were set to 5 and 100 m, respectively.

Although selecting thresholds for the LCSS and EDR are application dependent, a radius of 100 m or less is a common choice for these methods (Furtado et al., 2018). After experimenting with several values, the average distance between the points of the trajectories was selected as the threshold, which is approximately 50 m.

To assess the sensitivity to displacement, the reference trajectory points are shifted at different distances and rates, and the similarities between the reference trajectory and the transformed trajectories are calculated. Figures 7 and 8 show the results of similarity with different distances and rates, respectively. In addition, α and β values were calculated for every two consecutive sets of parameters (e.g., $r_1 = 0.1$ and $r_2 = 0.2$), and their averages for each method are reported in Table 2.

As illustrated in Figure 7, in the LCSS and EDR that use a threshold, the similarity does not change as long as the distance is less than the threshold. Only when the distance is larger than the threshold, the similarity dropped off a bit and then it remains constant. For this reason, LCSS and EDR have the least β (0.25 and 0.27, respectively). In addition, in the DTW, ERP, FD, and HD, the similarity decreases with a large slope as the distance increases and it makes them have a larger value for β. Furthermore, even when the displacement distance is small (e.g., 10 m), the

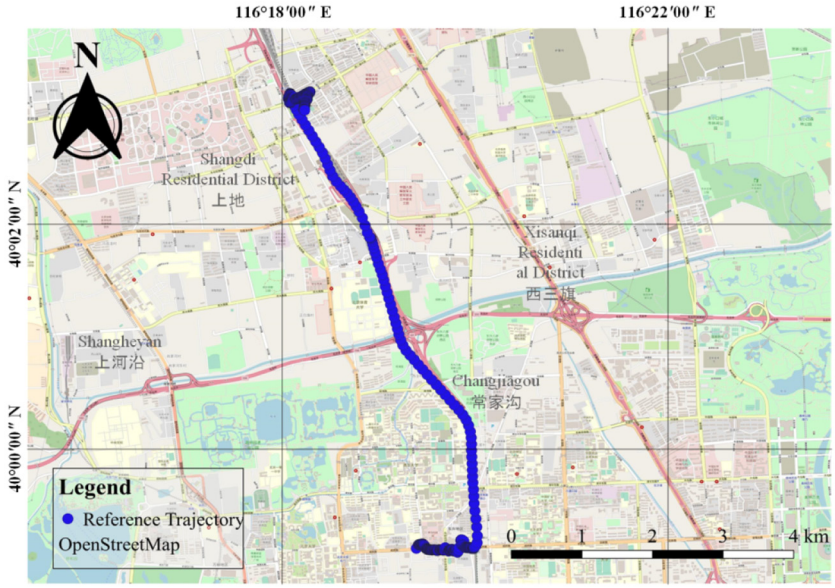

FIGURE 5 Reference trajectory

TABLE 1 List of transformation types and values of parameters used in experiment 1

Transformation type	Distance (m)	Rate
Displacement	{10, 20, 30, ..., 110}	0.5
Displacement	60	{0.1, 0.2, ..., 0.9}
Noise addition	{10, 20, 30, ..., 110}	0.5
Noise addition	60	{0.1, 0.2, ..., 0.9}
Sampling rate increase	–	{0.1, 0.2, ..., 0.9}
Sampling rate decrease	–	{0.1, 0.2, ..., 0.9}

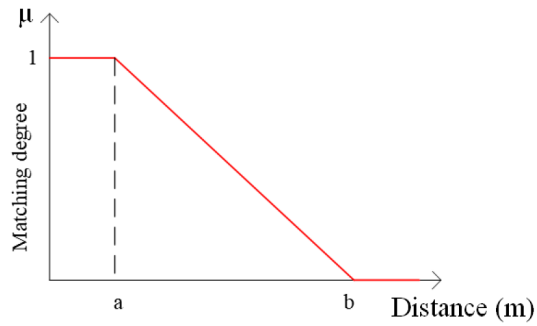

FIGURE 6 Fuzzy membership function used in the FLCSS

similarity is also too small that shows high sensitivity to point displacement. Unlike other methods, in the FLCSS, first, the similarity decreases in proportion to the increase in distance with a medium slope ($\beta = 0.60$), and then, because the displaced points are converted to NSP and shifted in random directions, the similarity increases to some extent. Moreover, it is shown in Figure 8 that in HD and FD, by increasing the rate of displacement, the similarity remains constant ($\alpha = 0$). However, both have the lowest values of similarity. In other methods, the similarity

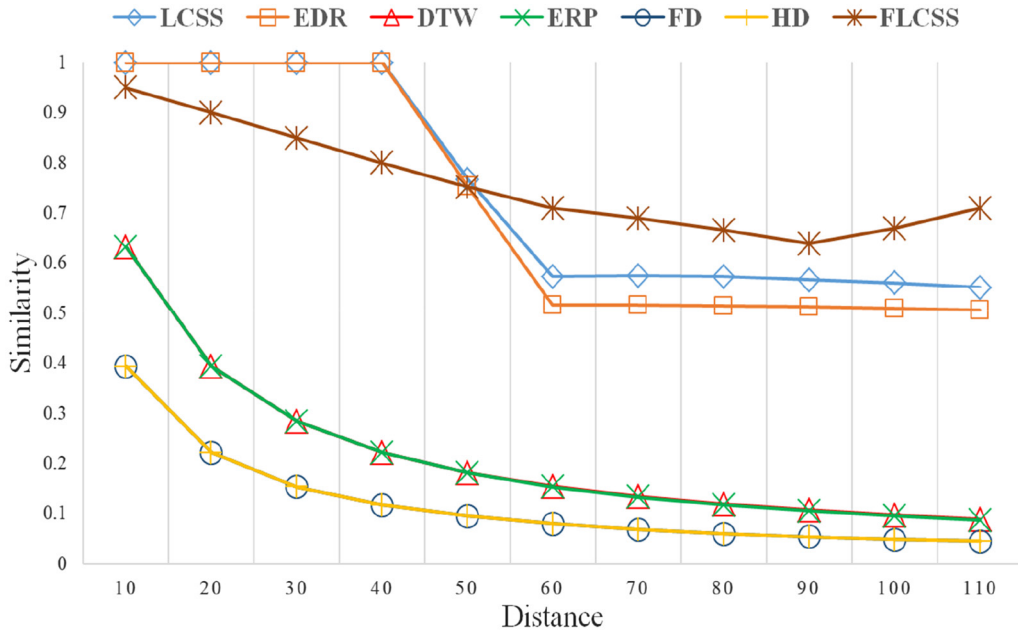

FIGURE 7 The similarity of trajectories resulting from the displacement of points with different distances

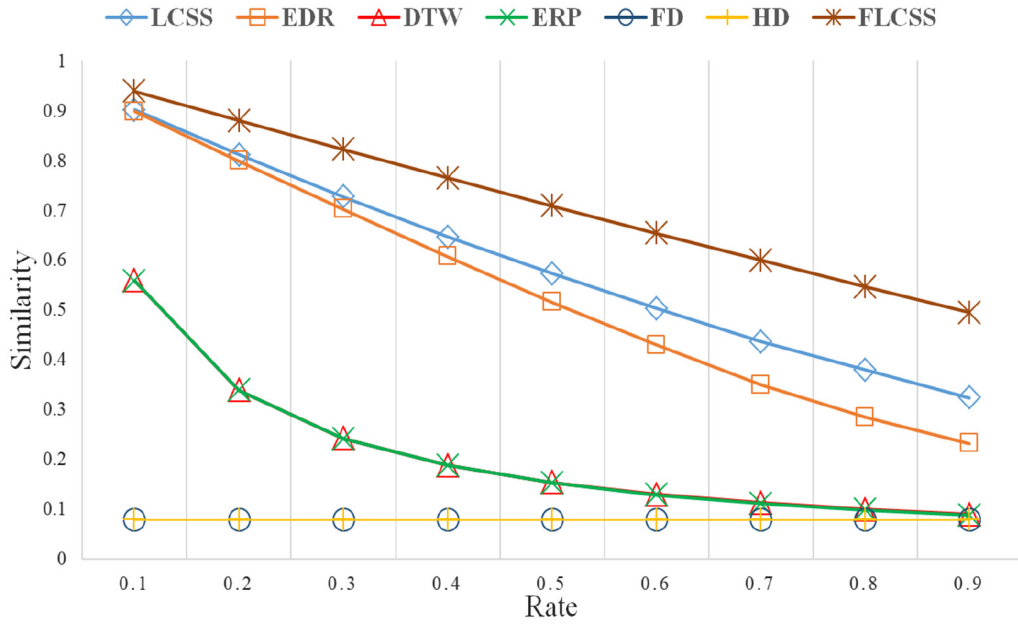

FIGURE 8 The similarity of trajectories resulting from the displacement of points with different rates

decreases by increasing the rate of change, among which FLCSS has the lowest slope ($\alpha = 0.37$), followed by the LCSS, EDR, DTW, and ERP, respectively.

In the HD and FD shape-based functions, the displacement of only one point can cause a decrease in similarity. That is why in these methods the rate of displacement has no effect on similarity and instead, the distance of displacement causes a sharp drop in similarity. This indicates the high sensitivity of these methods to the uncertainty

TABLE 2 Average values of α and β for the displacement of points

	LCSS	EDR	DTW	ERP	FD	HD	FLCSS
α	0.57	0.76	0.70	0.70	0.00	0.00	0.37
β	0.25	0.27	0.71	0.73	0.76	0.76	0.60

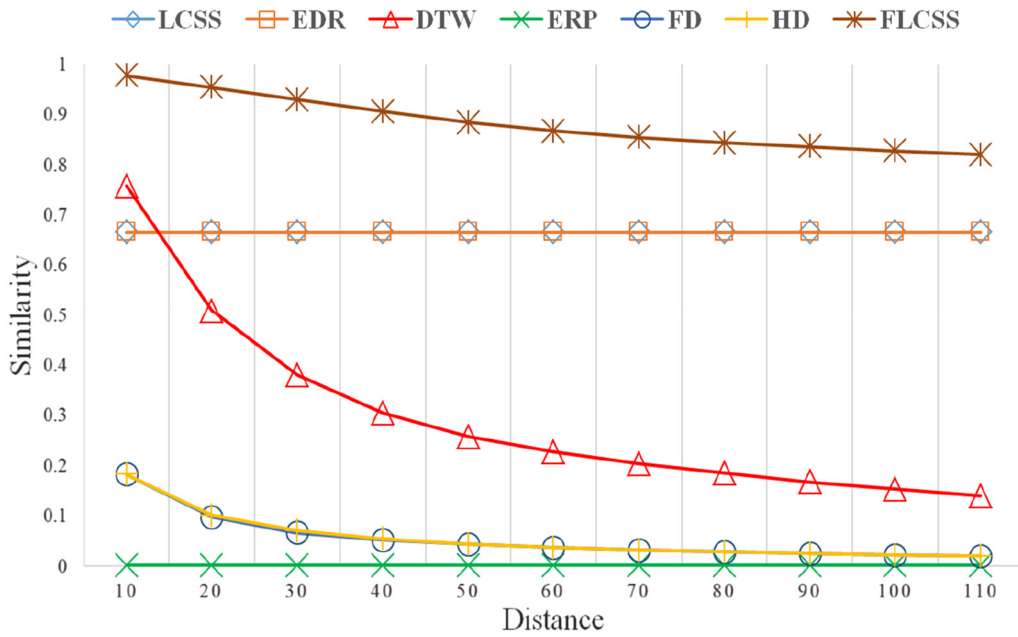

FIGURE 9 The similarity of trajectories resulting from adding noise with different distances

and positioning error. The DTW and ERP are also very sensitive to rate and distance of displacement. Therefore, it can be concluded that they are sensitive to the positioning error. The LCSS and EDR perform well only when the positioning error is smaller than the threshold, which indicates the importance and difficulty of selecting an appropriate threshold. In contrast to other methods, FLCSS shows reasonable behavior in rate and distance increment, which is confirmed by the α and β values. It proves the effectiveness of FLCSS when considering uncertainty caused by the positioning error.

In this step, noise is added to the reference trajectory at different rates and distances according to the values in Table 1. The results of similarity measurement between the reference trajectory and the transformed trajectories with different distances and rates are shown in Figures 9 and 10, respectively. In addition, the average values of α and β are reported in Table 3.

From the figures, it can be seen that with different distances and rates of adding noise, the similarities calculated by ERP are approximately equal to 0. The reason is that a cost is considered for NSPs in this method which is calculated based on the distance of these points to a fixed point. The effect of noise on DTW is also significant. Given that $\alpha < \beta$, it can be concluded that sensitivity to increasing distance is more than sensitivity to increasing rate of adding noise. FD and HD are susceptible to noise too. Similar to DTW, the effect of increasing distance is greater than increasing rate. The results of LCSS and EDR are almost the same and they have better performance than previous methods. In these two methods, for each point of the first trajectory, a maximum of one corresponding point in the second trajectory is considered. Therefore, increasing the distance of noise-contaminated points does not affect the similarity of the two trajectories ($\beta = 0$). However, an increase in the number of points

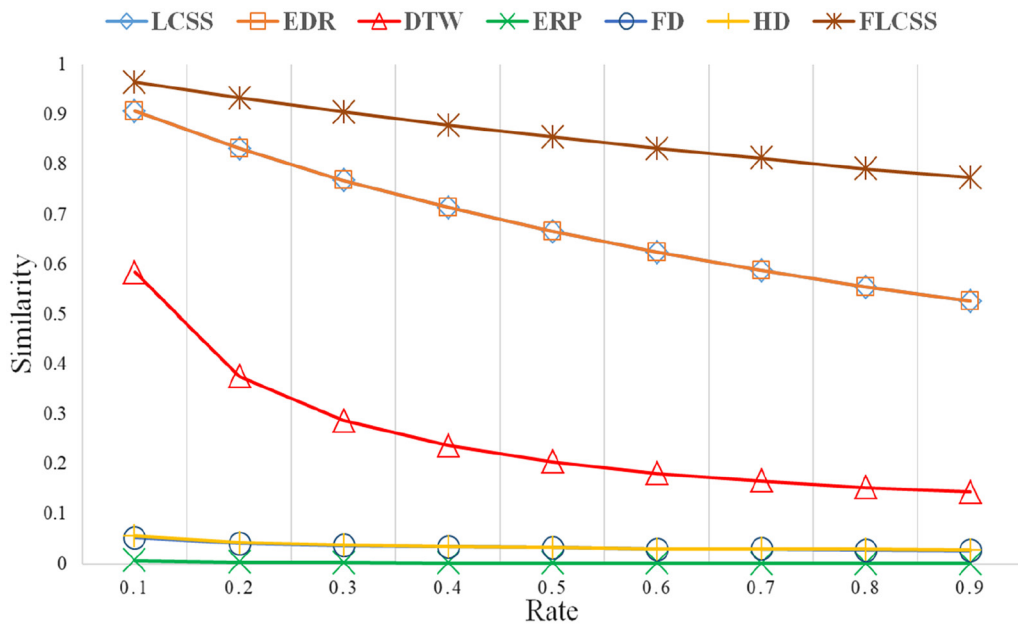

FIGURE 10 The similarity of trajectories resulting from adding noise with different rates

TABLE 3 Average values of α and β for adding noise

	LCSS	EDR	DTW	ERP	FD	HD	FLCSS
α	0.27	0.27	0.50	0.53	0.24	0.25	0.12
β	0.00	0.00	0.63	0.00	0.75	0.75	0.08

reduces the similarity. In contrast to the LCSS and EDR, in the FLCSS, the points that are added to the reference trajectory can match to some degree. Therefore, the similarity decreases with a slight slope as the distance and rate of noise increase ($\alpha = 0.12$ and $\beta = 0.08$).

To simulate different sampling rates, a number of points are added to the reference trajectory or some points are removed through sampling rate increase and decrease transformations. The number of points is determined according to the rate of changes mentioned in Table 1. The results are shown in Figures 11 and 12, respectively. Since these two functions are only dependent on the rate, only α value is calculated as reported in Table 4.

Increasing the sampling rate is similar to adding noise, except that the new points are located between the reference trajectory points and not in random directions. Therefore, the shape of the transformed trajectory does not change and FD and HD are expected to perform well. However, it is shown in Figure 11 that the similarity is very low (about 0.1) and it remains almost constant. This is because discrete versions of FD and HD are used for trajectories, and the greatest distance between the nearest points is considered as the similarity of the two trajectories. The performance of the other competing methods is similar to theirs against noise addition. Conversely, FLCSS obtains a similarity equal to 1 for all rates which demonstrates the robustness of the proposed method against the sampling error.

In the sampling rate increase transformation, the number of NSPs increases due to the increase in mismatched points. In contrast, in the sampling rate decrease transformation, the number of NSPs increases due to deleting the matching points. Accordingly, in this transformation, the decrease in similarities calculated by LCSS and EDR occurs with a greater slope as the rate of change increases ($\alpha = 1.29$). In addition, according

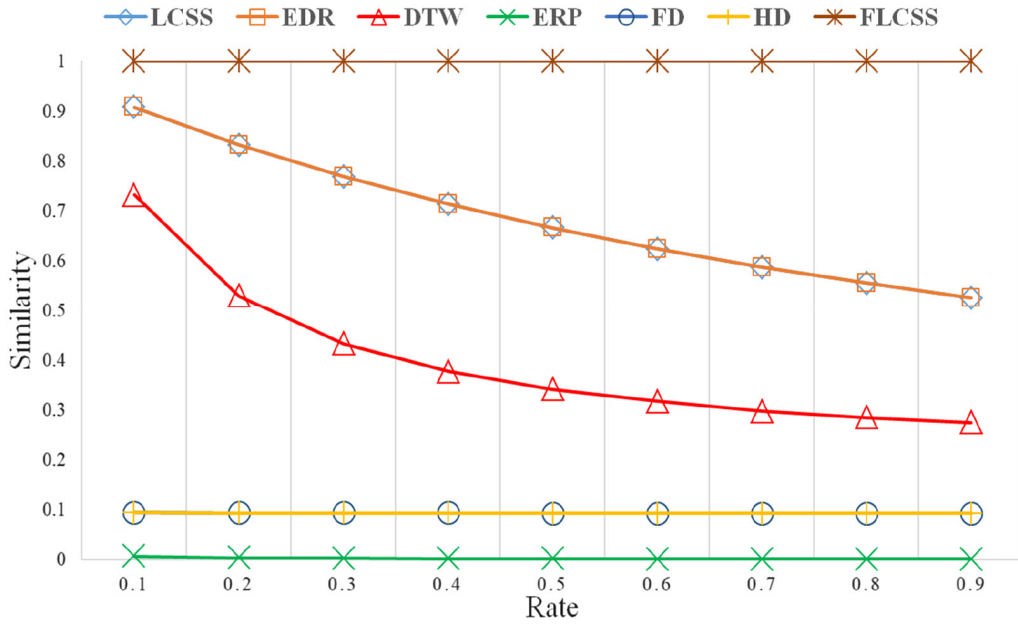

FIGURE 11 Similarity of trajectories resulting from increasing sampling rate

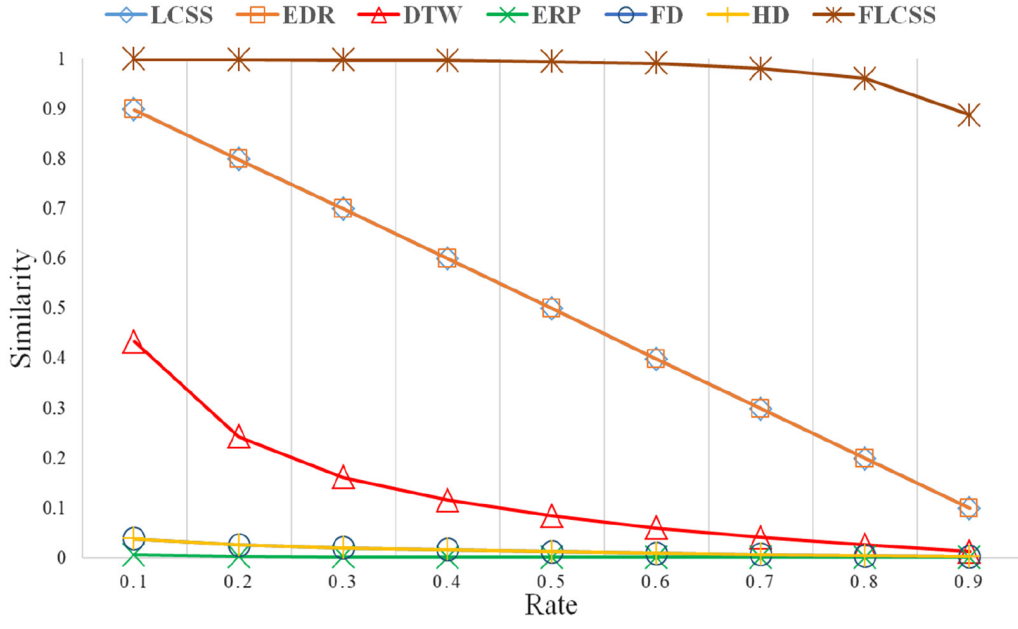

FIGURE 12 Similarity of trajectories resulting from decreasing sampling rate

TABLE 4 Average values of α for increasing and decreasing sampling rate

	LCSS	EDR	DTW	ERP	FD	HD	FLCSS
Sampling rate increase	0.27	0.27	0.34	0.53	0.01	0.01	0
Sampling rate decrease	1.29	1.29	1.63	0.77	1.35	1.35	0.11

to higher values of α, something similar but more intense than what occurred in adding noise and increasing sampling rate occurs for DTW and ERP. Furthermore, in this transformation, which greatly changes the shape of the reference trajectory, FD and HD show poor performance. In contrast, FLCSS shows high robustness against increasing the rate of change, and the similarity decreases only when the rate of change grows to about 0.7.

4.2.2 | Experiment 2: Real dataset

This experiment aims to verify the validity and computational efficiency of FLCSS. For this purpose, the relationship between similarity values calculated by different methods is investigated using the Pearson correlation coefficient. In this experiment, the Geolife dataset is used, and the fuzzy membership function for the FLCSS and the thresholds for the LCSS and EDR are set similar to Experiment 1.

Table 5 shows the correlation coefficients between different methods. The high correlation between FLCSS and LCSS (0.91) confirms the validity and correctness of the results of FLCSS. In addition, a slightly lower correlation is seen between FLCSS and EDR. However, there are low correlations between FLCSS and DTW, ERP, FD, and HD. This is because FLCSS, LCSS, and EDR are limited by the number of points in the two trajectories. Although DTW, ERP, FD, and HD were normalized, they are unbounded.

To evaluate the efficiency of FLCSS, the similarities between a randomly selected trajectory and all other trajectories are calculated using FLCSS and LCSS, and their execution times are recorded. The results show that on average, FLCSS is four times slower than LCSS. However, similar to LCSS, as the length of trajectories increases, the execution time of FLCSS also increases up to 1 s. The relationship between execution time and the length of trajectories was clear. However, another parameter affects the execution time in FLCSS. By comparing the steps of LCSS and FLCSS, it can be noticed that this parameter is related to the number of NSPs. Because NSPs must be identified first, then the matching degree must be recalculated for each NSP. It indicates that the execution time of FLCSS is also related to the uncertainty and sampling error of the trajectories.

Considering the results, it can be noticed that the robustness of trajectory similarity methods to sampling error is related to how they consider NSPs. For example, in the ERP, which has the weakest performance, a fixed point is used to manage the NSPs, and the distances between this point and the points added to the reference trajectory are very long. In the DTW, the smallest distance to each NSP is used in calculating the similarity, which can be a relatively large and reduce the similarity. NSPs that are added in random directions and distances to the reference trajectory cause it to deform. Consequently, for the shape-based FD and HD, a sharp drop in similarity occurs even when the distance or rate of changes is small. Furthermore, LCSS and EDR perform relatively weak against sampling error due to the use of the threshold for determining the matching points and neglecting the NSPs.

TABLE 5 The correlation coefficient (ρ) between different methods

	FLCSS	LCSS	EDR	DTW	ERP	FD	HD
FLCSS	–	–	–	–	–	–	–
LCSS	0.91	–	–	–	–	–	–
EDR	0.83	0.93	–	–	–	–	–
DTW	0.33	0.37	0.43	–	–	–	–
ERP	0.17	0.20	0.24	0.78	–	–	–
FD	0.13	0.16	0.20	0.67	0.81	–	–
HD	0.17	0.19	0.23	0.76	0.86	0.97	–

Eventually, FLCSS is the most robust against sampling error among studied methods, which is mainly because of the use of the bead model to consider NSPs in similarity measurement.

As mentioned in the experiments, FLCSS is four times slower than LCSS due to more complex calculations to take into account the uncertainty in trajectory datasets. Thereby, FLCSS may not be suitable for applications that require real-time computations, which is the most important drawback of this method. In addition, unlike a parameter-free method, such as DTW, FLCSS requires the relationship between matching degree and distance to be defined through a fuzzy membership function which is application dependent and sometimes hard to define.

5 | CONCLUSION AND FUTURE WORK

In this article, a fuzzy version of LCSS, named FLCSS, was developed to measure the similarity between trajectories of uncertain values. In this method, fuzzy theory and the bead model were used to consider the uncertainty of movement data caused by positioning and sampling errors. Two experiments were implemented on synthetic and real trajectories to evaluate FLCSS. The first experiment examined FLCSS from sensitivity to the displacement of sampling points, adding noise, and increasing and decreasing the sampling rate aspects. The second experiment was carried out to prove the validity and correctness of FLCSS. The results clearly show that FLCSS has a better performance than LCSS, EDR, DTW, FD, and HD.

There are a variety of paths for future research. FLCSS can be developed for trajectories that are enriched and annotated by semantic and context information and consider the underlying uncertainty in these datasets. In addition, the results of similarity searches can be employed in movement prediction, pattern recognition, trajectory classification, and trajectory clustering of moving entities while taking the uncertainty in the dataset into account.

CONFLICT OF INTEREST

No potential conflict of interest was reported by any of the authors.

DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

ORCID

Ali Asghar Alesheikh https://orcid.org/0000-0001-9537-9401
Mohammad Sharif https://orcid.org/0000-0002-1241-007X
Mahdi Farnaghi https://orcid.org/0000-0002-3071-5563

REFERENCES

Alizadeh, D., Alesheikh, A. A., & Sharif, M. (2021a). Prediction of vessels locations and maritime traffic using similarity measurement of trajectory. *Annals of GIS*, *27*, 151–162. https://doi.org/10.1080/19475683.2020.1840434

Alizadeh, D., Alesheikh, A. A., & Sharif, M. (2021b). Vessel trajectory prediction using historical automatic identification system data. *Journal of Navigation*, *74*, 156–174. https://doi.org/10.1017/S0373463320000442

Alt, H., & Guibas, L. J. (2000). Discrete geometric shapes: Matching, interpolation, and approximation. In J. R. Sack & J. Urrutia (Eds.), *Handbook of computational geometry* (pp. 3.1–3.34). North-Holland.

Berndt, D. J., & Clifford, J. (1994). Using dynamic time warping to find patterns in time series. In *Proceedings of the 1994 KDD Workshop*, Seattle, WA (pp. 359–370). AAAI.

Buchin, M., Dodge, S., & Speckmann, B. (2014). Similarity of trajectories taking into account geographic context. *Journal of Spatial Information Science*, *2014*, 101–124. https://doi.org/10.5311/JOSIS.2014.9.179

Chen, L., & Ng, R. (2004). On the marriage of lp-norms and edit distance. In *Proceedings of the 30th International Conference on Very Large Data Bases*, Toronto, ON, Canada (pp. 792-803). VLDB Endowment.

Chen, L., Özsu, M. T., & Oria, V. (2005). Robust and fast similarity search for moving object trajectories. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, Baltimore, MD (pp. 491-502). ACM.

Chen, L., Tang, Y., Lv, M., & Chen, G. (2015). Partition-based range query for uncertain trajectories in road networks. *GeoInformatica*, *19*, 61–84. https://doi.org/10.1007/s10707-014-0206-6

Dodge, S., Laube, P., & Weibel, R. (2012). Movement similarity assessment using symbolic representation of trajectories. *International Journal of Geographical Information Science*, *26*, 1563–1588. https://doi.org/10.1080/13658816.2011.630003

Eiter, T., & Mannila, H. (1994). *Computing discrete Fréchet distance*. Christian Doppler Laboratory for Expert Systems, TU Vienna.

Frentzos, E., Gratsias, K., & Theodoridis, Y. (2009). On the effect of location uncertainty in spatial querying. *IEEE Transactions on Knowledge and Data Engineering*, *21*, 366–383. https://doi.org/10.1109/TKDE.2008.164

Furtado, A. S., Alvares, L. O. C., Pelekis, N., Theodoridis, Y., & Bogorny, V. (2018). Unveiling movement uncertainty for robust trajectory similarity analysis. *International Journal of Geographical Information Science*, *32*, 140–168. https://doi.org/10.1080/13658816.2017.1372763

Furtado, A. S., Kopanaki, D., Alvares, L. O., & Bogorny, V. (2016). Multidimensional similarity measuring for semantic trajectories. *Transactions in GIS*, *20*, 280–298. https://doi.org/10.1111/tgis.12156

Goudarzi, S., Sharif, M., & Karimipour, F. (2022). A context-aware dimension reduction framework for trajectory and health signal analyses. *Journal of Ambient Intelligence and Humanized Computing*, *13*, 2621–2635. https://doi.org/10.1007/s12652-021-03569-z

Guo, N., Shekhar, S., Xiong, W., Chen, L., & Jing, N. (2019). UTSM: A trajectory similarity measure considering uncertainty based on an amended ellipse model. *ISPRS International Journal of Geo-Information*, *8*, 518. https://doi.org/10.3390/ijgi8110518

Hornsby, K., & Egenhofer, M. J. (2002). Modeling moving objects over multiple granularities. *Annals of Mathematics and Artificial Intelligence*, *36*, 177–194. https://doi.org/10.1023/A:1015812206586

Jeung, H., Lu, H., Sathe, S., & Yiu, M. L. (2014). Managing evolving uncertainty in trajectory databases. *IEEE Transactions on Knowledge and Data Engineering*, *26*, 1692–1705. https://doi.org/10.1109/TKDE.2013.141

Kaffash-Charandabi, N., Alesheikh, A. A., & Sharif, M. (2019). A ubiquitous asthma monitoring framework based on ambient air pollutants and individuals' contexts. *Environmental Science and Pollution Research*, *26*, 7525–7539. https://doi.org/10.1007/s11356-019-04185-3

Karami, F., & Malek, M. R. (2021). Trajectory similarity measurement: An enhanced maximal travel match method. *Transactions in GIS*, *25*(3), 1485–1503. https://doi.org/10.1111/tgis.12733

Kuijpers, B., Miller, H. J., & Othman, W. (2017). Kinetic prisms: incorporating acceleration limits into space-time prisms. *International Journal of Geographical Information Science*, *31*, 2164–2194. https://doi.org/10.1080/13658816.2017.1356462

Kuijpers, B., & Othman, W. (2010). Trajectory databases: Data models, uncertainty and complete query languages. *Journal of Computer and System Sciences*, *76*, 538–560. https://doi.org/10.1016/j.jcss.2009.10.002

Kurbalija, V., Radovanović, M., Geler, Z., & Ivanović, M. (2014). The influence of global constraints on similarity measures for time-series databases. *Knowledge-Based Systems*, *56*, 49–67. https://doi.org/10.1016/j.knosys.2013.10.021

Laube, P., & Purves, R. S. (2011). How fast is a cow? Cross-scale analysis of movement data. *Transactions in GIS*, *15*, 401–418. https://doi.org/10.1111/j.1467-9671.2011.01256.x

Lehmann, A. L., Alvares, L. O., & Bogorny, V. (2019). SMSM: A similarity measure for trajectory stops and moves. *International Journal of Geographical Information Science*, *33*, 1847–1872. https://doi.org/10.1080/13658816.2019.1605074

Mennis, J., & Guo, D. (2009). Spatial data mining and geographic knowledge discovery: An introduction. *Computers, Environment and Urban Systems*, *33*, 403–408. https://doi.org/10.1016/j.compenvurbsys.2009.11.001

Miller, H. J., & Han, J. (2009). *Geographic data mining and knowledge discovery*. CRC Press.

Motro, A. (1993). Sources of uncertainty in information systems. In *Proceedings of the Second Workshop on Uncertainty Management and Information Systems: From Needs to Solutions*, Fairfax, VA (pp. 9-34). Kluwer.

Naderivesal, S., Kulik, L., & Bailey, J. (2019). An effective and versatile distance measure for spatiotemporal trajectories. *Data Mining and Knowledge Discovery*, *33*, 577–606. https://doi.org/10.1007/s10618-019-00615-5

Pelekis, N., Kopanakis, I., Kotsifakos, E., Frentzos, E., & Theodoridis, Y. (2009). Clustering trajectories of moving objects in an uncertain world. In *Proceedings of the Ninth IEEE International Conference on Data Mining*, Miami, FL (pp. 417-427). IEEE.

Pfoser, D., & Jensen, C. S. (1999). Capturing the uncertainty of moving-object representations. In R. H. Güting, D. Papadias, & F. Lochovsky (Eds.), *Advances in spatial databases: SSD 1999* (Lecture Notes in Computer Science, Vol. *1651*, pp. 111-131). Springer.

Sharif, M., & Alesheikh, A. A. (2017). Context-awareness in similarity measures and pattern discoveries of trajectories: A context-based dynamic time warping method. *GIScience & Remote Sensing*, *54*, 426–452. https://doi.org/10.1080/15481603.2017.1278644

Sharif, M., & Alesheikh, A. A. (2018). Context-aware movement analytics: implications, taxonomy, and design framework. *WIREs Data Mining and Knowledge Discovery*, *8*, e1233. https://doi.org/10.1002/widm.1233

Sharif, M., Alesheikh, A. A., & Tashayo, B. (2019). CaFIRST: A context-aware hybrid fuzzy inference system for the similarity measure of multivariate trajectories. *Journal of Intelligent & Fuzzy Systems*, *36*, 5383–5395. https://doi.org/10.3233/JIFS-181252

Shi, W., Chen, P., Shen, X., & Liu, J. (2021). An adaptive approach for modelling the movement uncertainty in trajectory data based on the concept of error ellipses. *International Journal of Geographical Information Science*, *35*, 1131–1154. https://doi.org/10.1080/13658816.2020.1828591

Su, H., Liu, S., Zheng, B., Zhou, X., & Zheng, K. (2020). A survey of trajectory distance measures and performance evaluation. *The VLDB Journal*, *29*, 3–32. https://doi.org/10.1007/s00778-019-00574-9

Toohey, K., & Duckham, M. (2015). Trajectory similarity measures. *SIGSPATIAL Special*, *7*(1), 43–50. https://doi.org/10.1145/2782759.2782767

Trajcevski, G., Wolfson, O., Hinrichs, K., & Chamberlain, S. (2004). Managing uncertainty in moving objects databases. *ACM Transactions on Database Systems*, *29*, 463–507. https://doi.org/10.1145/1016028.1016030

Vlachos, M., Kollios, G., & Gunopulos, D. (2002). Discovering similar multidimensional trajectories. In *Proceedings 18th International Conference on Data Engineering*, San Jose, CA (pp. 673-684). IEEE.

Wang, S., Wu, L., Zhou, F., Zheng, C., & Wang, H. (2015). Group pattern mining on moving objects' uncertain trajectories. *International Journal of Computers Communications & Control*, *10*, 428–440. https://doi.org/10.15837/ijccc.2015.3.1667

Xie, D., Li, F., & Phillips, J. M. (2017). Distributed trajectory similarity search. *Proceedings of the VLDB Endowment*, *10*, 1478–1489. https://doi.org/10.14778/3137628.3137655

Yager, R. R. (2002). Uncertainty representation using fuzzy measures. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *32*, 13–20. https://doi.org/10.1109/3477.979955

Zadeh, L. A. (2011). Generalized theory of uncertainty: Principal concepts and ideas. In S. M. S. A. Brandolini & R. Scazzieri (Eds.), *Fundamental uncertainty: Rationality and plausible reasoning* (pp. 104–150). Palgrave Macmillan UK.

Zaki, M. H., & Sayed, T. (2018). Automated analysis of pedestrian group behavior in urban settings. *IEEE Transactions on Intelligent Transportation Systems*, *19*, 1880–1889.

Zheng, K., Trajcevski, G., Zhou, X., & Scheuermann, P. (2011). Probabilistic range queries for uncertain trajectories on road networks. In *Proceedings of the 14th international conference on extending database technology* (pp. 283–294). ACM.

Zheng, Y., Zhang, L., Xie, X., & Ma, W.-Y. (2009). Mining interesting locations and travel sequences from GPS trajectories. In *Proceedings of the 18th international conference on world wide web* (pp. 791–800). ACM.

How to cite this article: Boroumand, F., Alesheikh, A. A., Sharif, M. & Farnaghi, M. (2022). FLCSS: A fuzzy-based longest common subsequence method for uncertainty management in trajectory similarity measures. *Transactions in GIS*, *00*, 1–19. https://doi.org/10.1111/tgis.12958

APPENDIX A

This section provides the definitions of the comparative methods mentioned in Section 2.2.

EDIT DISTANCE ON REAL SEQUENCE (EDR)

For trajectories *A* and *B*, EDR distance is equal to the number of operations required to convert *A* to *B* (Chen et al., 2005). This operation involves deleting, adding, and replacing points. The EDR distance between trajectories *A* and *B* with lengths *m* and i and a threshold of ε is defined using Equation (A1).

$$EDR_{\varepsilon}(A,B)=\begin{cases} m & \text{if } n=0 \\ n & \text{if } m=0 \\ \min\begin{cases} subcost_{\varepsilon}(a_m, b_n)+EDR_{\varepsilon}(Head(A), Head(B)) \\ 1+EDR_{\varepsilon}(Head(A), B) \\ 1+EDR_{\varepsilon}(A, Head(B)) \end{cases} & \text{otherwise} \end{cases} \tag{A1}$$

where $subcost_{\varepsilon}(a_m, b_n)=1$ if the distance between a_m and b_n is greater than threshold, and $subcost_{\varepsilon}(a_m, b_n)=0$ otherwise (Toohey & Duckham, 2015).

DYNAMIC TIME WARPING (DTW)

DTW uses local time shifting to find the shortest distance between two trajectories. It means the trajectories are warped in a nonlinear way to minimize the distance between the trajectories. DTW distance between trajectories *A* and *B* with lengths m and n is given by Equation (A2) (Kurbalija et al., 2014).

$$DTW(A,B)=\begin{cases}0 & \text{if } m=n=0\\ \infty & \text{if } (m=0,\ n>0) \text{ or } (m>0, n=0)\\ |a_m-b_n|+\min\begin{cases}DTW(Head(A),\ Head(B))\\ DTW(Head(A),\ B)\\ DTW(A,\ Head(B))\end{cases} & \text{otherwise}\end{cases} \tag{A2}$$

EDIT DISTANCE WITH REAL PENALTY (ERP)

Equation (A3) represents ERP distance between trajectories *A* and *B* with lengths m and n. g is a constant point and the cost for non-matching points is calculated based on the distance to this point (Chen & Ng, 2004).

$$ERP(A,B)=\begin{cases}\sum_{i=1}^{n}|b_i-g| & \text{if } m=0\\ \sum_{i=1}^{m}|a_i-g| & \text{if } n=0\\ \min\begin{cases}|a_m-b_n|+ERP(Head(A),\ Head(B))\\ |a_m-g|+ERP(Head(A),\ B)\\ |b_n-g|+ERP(A,\ Head(B))\end{cases} & \text{otherwise}\end{cases} \tag{A3}$$

HAUSDORFF DISTANCE (HD)

For trajectories *A* and *B*, HD is equal to the largest Euclidean distance among the distances between each point of one trajectory to the closest point to them on the other trajectory. For trajectories *A* and *B*, the discrete HD is given by Equation (A4) (Alt & Guibas, 2000).

$$H(A,B)=\max(h(A,B),\ h(B,A)) \tag{A4}$$

where $h(A, B)$ is defined by Equation (A5).

$$h(A,B)=\max_{\forall a\in A}\left(\min_{\forall b\in B}(|a-b|)\right) \tag{A5}$$

FRÉCHET DISTANCE (FD)

FD is similar to HD except that FD takes into account the order of the points. The discrete FD between *A* and *B* with lengths m and n is defined by Equation (A6) (Eiter & Mannila, 1994).

$$F(A,B)=\min(|L|) \tag{A6}$$

where *L* is a coupling between the points of *A* and *B*. Assuming that $m>n$, a coupling *L* is a sequence as Equation (A7), such that $\alpha(1)=1$, $\beta(1)=1$, $\alpha(m)=m$ and $\beta(m)=n$.

$$L=\{(A(\alpha(1)),\ B(\beta(1))),\ (A(\alpha(2)),\ B(\beta(2))),\ \ldots,\ (A(\alpha(m)),\ B(\beta(m)))\} \tag{A7}$$

Eventually, the length of *L* is calculated by Equation (A8).

$$|L|=\max_{i=1,\ldots,m}|A(\alpha(i))-B(\beta(i))| \tag{A8}$$