Clustering and Representing Movements in an Uncertain World

Nikos Pelekis, Ioannis Kopanakis, Evangelos E. Kotsifakos, Elias Frentzos, Yannis Theodoridis

Laboratory of Information Systems

Department of Informatics

University of Piraeus

Hellas

Technical Report Series

UNIPI-ISL-TR-2009-01

February 2009

Clustering and Representing Movements in an Uncertain World

Nikos Pelekis[1], Ioannis Kopanakis[2], Evangelos E. Kotsifakos[1], Elias Frentzos[1], Yannis Theodoridis[1]

[1]*Dept. of Informatics, Univ. of Piraeus, Greece*
{npelekis, ek, efrentzo, ytheod}@unipi.gr

[2]*Tech. Educational Institute of Crete, Greece i.kopanakis@emark.teicrete.gr*

ABSTRACT

Knowledge discovery in Trajectory Databases (TD) is an emerging field which has recently gained great interest; on the other hand the inherent presence of uncertainty in TD during the mining process has not been taken yet into account. Current approaches group trajectories together by accounting only the degree of their similarity, ignoring at the same time the degree in which they are unrelated and more importantly the hesitancy introduced by the uncertainty. In this paper we study the effect of uncertainty on clustering TD, as well as the representation of the clusters and introduce a three-step approach to deal with this challenge. Firstly, we propose an intuitionistic point vector representation of trajectories that encompasses the underlying uncertainty and we introduce an effective distance metric to cope with uncertainty. At the second step, we devise *CenTra*, a novel algorithm which tackles the problem of discovering the *Centroid Trajectory* of a group of movements taking into advantage the local similarity between portions of trajectories. At the final step, we devise *CenTR-I-FCM* a variant of the Fuzzy C-Means (FCM) clustering algorithm, appropriately modified in the context of Intuitionistic Fuzzy Set (IFS) theory, which embodies *CenTra* as its update procedure when producing the cluster centroids at each iteration. The experimental evaluation with real world trajectory datasets demonstrates the efficiency and effectiveness of our approach.

1. INTRODUCTION

With the integration of wireless communications and positioning technologies, the concept of TD has become increasingly important posing great challenges to the data mining community [13]. On the other hand, since TD consist of objects changing and recording their position over time, such as moving humans or vehicles, uncertainty appears within a TD in various ways; uncertainty due to sampling and/or measurement errors [23], uncertainty in querying and answering [24], or due to other pre-processing tasks like preserving anonymity in TD [1]. Though always present into TD, to the best of our knowledge there is no work in the database literature that studies the effect of the above various types of uncertainty in knowledge discovery tasks from TD.

Clustering of trajectories into separate collections, involves partitioning of a TD into clusters (groups), so that each cluster contains proximate trajectories according to some distance measure. Previous research has mostly focused on clustering of point data that trajectories do not conform to. This means that well-known clustering algorithms (e.g., k-means [20], BIRCH [30], DBSCAN [10], STING [26]) can not be directly applied. As such the idea of measuring the similarity between two trajectories is an attractive solution that has been utilized as the mean to cluster trajectories. Many approaches have been introduced in the literature that try to quantify the (dis)-similarity between trajectories, dealing with basic trajectory features, [25], [28], [5], [6], [22]. However, neither of the above mentioned clustering algorithms nor the previously cited approaches for similarity search deal with the inherent uncertainty in TD.

On the other hand, clustering approaches based on fuzzy logic [29], such as FCM [3], provide some kind of support in uncertainty handling by allowing each data point to belong to different clusters by a certain degree. Considering that input vector values are subject to uncertainty due to imprecise measurements, noise or sampling errors, the distances that determine the membership of a point to a cluster will also be subject to uncertainty. Therefore the possibility of erroneous membership assignments in the clustering process is evident. Current fuzzy clustering approaches do not utilize any information about uncertainty at the elementary level of the data points, which for the case of trajectories are the spatial locations of the objects recorded in temporal order.

In this paper we introduce a three step approach to deal with such kind of information. We initially adopt a symbolic representation and model trajectories as sequences of regions (i.e. wherefrom a moving object passes) accompanied with intuitionistic fuzzy values, i.e. elements of an intuitionistic fuzzy set. Intuitionistic fuzzy sets [2] are generalized fuzzy sets [29] that can be useful in coping with the hesitancy originating from imprecise information. The elements of an intuitionistic fuzzy set are characterized by two values representing their belongingness and non-belongingness to this set, respectively. In the case of TD where this set is the region that a trajectory possibly crosses, the above values represent the probabilities of presence and non-presence in the area. In order to exploit this information, we define a novel distance metric especially designed to operate on such intuitionistic fuzzy vectors, having as goal to incorporate it in some variant of the FCM algorithm that will effectively cluster trajectories under uncertainty.

The success of any FCM-variant algorithm depends on the way that each cluster's centroids are driven towards the correct direction in each algorithm's iteration. However, in the TD setting where trajectories are complex objects of different lengths, varying sampling rates, different speeds, possible outliers and different scaling factors, even the most efficient similarity function would fail in different applications. We argue that we can succeed better clustering results if instead of using *global* similarity functions between whole trajectories we exploit *local* similarity properties between portions of the trajectories. Based on this idea,

at the second step of our approach, we propose *CenTra*, a novel density- as well as similarity-based algorithm to tackle the problem of discovering the *Centroid* of a group of trajectories. Finally, at the third step of our approach we utilize *CenTra* in the centroid update procedure of a new trajectory clustering algorithm, called *CenTR-I-FCM*, which uses a global uncertainty-supporting similarity function, to group trajectories at a higher level, and iteratively refine the results using local similarity between sub-trajectories. This algorithm has the efficiency advantages of partitioning clustering algorithms in comparison to the higher processing cost of density-based algorithms, whereas produces non-spherical clusters due to the inclusion of *CenTra,* that recognises representative movements of any shape.

Summarizing our discussion, the major contributions of this paper are the following:

- We propose an intuitionistic fuzzy vector representation of trajectories that enables the clustering of trajectories by known (fuzzy or not) clustering algorithms.
- We define a global distance metric on the previous trajectory representation which outperforms its competitors proposed in the literature.
- We tackle the problem of identifying the centroid of a bunch of trajectories using density and local similarity properties.
- We propose a novel modification of the FCM algorithm for clustering complex trajectory datasets based on the above distance measure and the idea of the centroid trajectory.
- We conduct a comprehensive set of experiments over a real trajectory dataset, in order to evaluate our approach.

The rest of this paper is structured as follows: Section 2 discusses related work in the involved domains. In Section 3, we introduce the intuitionistic vector representation of trajectories. The proposed similarity measure is defined in Section 4. In Section 5 we describe the *CenTra* and the *CenTR-I-FCM* algorithms, while the results of our experimental evaluation with real-world data are apposed in Section 6. Finally, the conclusions of this study along with ideas for future work are summarized in Section 7.

2. RELATED WORK

In this section we review existing works in the domains related with the current work.

Representing Uncertainty in TD - Probably, the most recognized notion of uncertainty in TD is the uncertainty of the trajectory representation, which means that the location of a moving object stored in a TD will not represent its real location due to a variety of reasons. Although this kind of uncertainty may be inherited by GPS erroneous measurements, its major source in TD is the interpolation method (usually linear) used to capture the complete movement of the moving object and estimate the object's location at timestamps in-between sampled positions. In [23], the authors define the notion of sampling error at a sampled position P_1 at a timestamp t_1 and they study the error behavior across the time axis. They prove the intuitively expected result that by increasing the sampling rate, the sample positions better approximate movement, and the error introduced by sampling is decreased. In [24] a model for uncertain trajectories is proposed that associates an uncertainty threshold ε to the whole trajectory, while a set of uncertainty operators were introduced so as to incorporate uncertainty into user queries. This approach results in trajectories with uncertainty modeled as cylindrical volumes in 3D space. Therefore, each trajectory point (x,y,t) is associated with an ε-uncertainty area which is actually a horizontal disk with radius ε centered at (x,y,t). In order to reduce the complexity of handling this kind of spherical neighborhoods, in [13] square uncertainty areas were introduced.

Clustering TD - Clustering is one of the most popular data mining tasks widely used in numerous applications. The vast majority of the proposed clustering algorithms, such as *k*-means [20], BIRCH [30], DBSCAN [10], and STING [26] are tailored to work with point data, making thus their application to TD not a straightforward task. During the last decade several approaches have been proposed in the literature so as to enable the previous algorithms to operate on trajectories. Most of these approaches are inspired by the time series analysis domain and propose trajectory similarity measures as the vehicle to group trajectories; they are interested in the movement shape of trajectories, which are usually considered as 2D or 3D time series data [25], [28], [5], [6]. None of the previous approaches uses the underlying uncertainty. Clustering approaches based on fuzzy logic [29], such as Fuzzy C-Means (FCM) [3] and its variants are competitive to conventional clustering algorithms, especially for real-world applications. The advantage of these approaches is that they do not consider sharp boundaries between clusters, allowing each vector to belong to different clusters by a certain degree. The degree of membership of a vector to a cluster is considered as a function of its distance from the cluster centroid or from other representative vectors of the cluster. However, the adoption of this technique in TD is not directly feasible mainly due to the complex nature of trajectories.

In the past, Gaffney et al. [12], [4] have proposed probabilistic algorithms for clustering short trajectories using a regression mixture model. Subsequently, unsupervised learning is carried out by using EM algorithm to determine the cluster memberships in the model. In this approach, the issue of uncertainty is not taken into account, while representation of cluster centroids is out of the scope of these papers. What is more, in our approach we make no assumption about the size of the trajectories or whether they conform to some regression model, since we are interested in complex, real-world objects following arbitrary movement patterns.

Recently, Lee et al. [17] proposed TRACLUS, a partition-and-group framework for clustering trajectories which enables the discovery of common sub-trajectories, based on a trajectory partitioning algorithm that uses the minimum description length principle. TRACLUS clusters trajectories as line segments (sub-trajectories) independently of whether the whole trajectories belong to different or the same clusters; for this reason a variant of DBSCAN for line segments is proposed [17]; finally, the notion of the *representative trajectory* of a cluster is provided. The fundamental difference of TRACLUS with our approach is that we cluster trajectories as a whole. Furthermore, contrary to our approach, the temporal information is not considered in [17], while the proposed algorithm for identifying the representative trajectory of a cluster primarily supports straight movement patterns and cannot identify complex (e.g. circular) motions, which are common case in real world applications. On top of

these differences, the work of Lee et al. [17] does not deal in any way with the uncertainty in the motion of the trajectories.

Intuitionistic Fuzzy Sets and Similarity - Regarding the theoretical foundations of fuzzy and intuitionistic fuzzy sets, these are described in [29], [2]. Here we briefly outline the basic notions used in this paper.

Definition 1. Let a set E be fixed. A fuzzy set on E is an object $\tilde{A}$ of the form

$$\tilde{A} = \left\{ \left\langle x, \mu_{\tilde{A}}(x) \right\rangle \middle| x \in E \right\}$$

where $\mu_{\tilde{A}} : E \to [0,1]$ defines the degree of membership of the element $x \in E$ to the set $\tilde{A} \subset E$. For every element $x \in E$, $0 \le \mu_{\tilde{A}}(x) \le 1$.■

Definition 2. An intuitionistic fuzzy set A is an object of the form

$$A = \left\{ \left\langle x, \mu_A(x), \gamma_A(x) \right\rangle \middle| x \in E \right\}$$

where $\mu_A : E \to [0,1]$ and $\gamma_A : E \to [0,1]$ define the degree of membership and non-membership, respectively, of the element $x \in E$ to the set $A \subset E$. For every element $x \in E$ it holds that $0 \le \mu_A(x) \le 1$ $0 \le \gamma_A(x) \le 1$ and $0 \le \mu_A(x) + \gamma_A(x) \le 1$. For every $x \in E$, if $\gamma_A(x) = 1 - \mu_A(x)$, A represents a fuzzy set. The function

$$\pi_A(x) = 1 - \mu_A(x) - \gamma_A(x)$$

represents the degree of *hesitancy* of the element $x \in E$ to the set $A \subset E$.■

The plethora and importance of the potential applications of intuitionistic fuzzy sets have drawn the attention of many researchers that have proposed various kinds of similarity measures between intuitionistic fuzzy sets. Example applications include identification of functional dependency relationships between concepts in data mining systems, approximate reasoning, pattern recognition and others. A variety of similarity measures between intuitionistic fuzzy sets have been proposed: S_C [7], [8], S_H [14], S_L [11], S_O by [19], S_{DC} [9], S_{HB} [21], S_e^p, S_s^p and S_h^p [31], and S_{HY}^1, S_{HY}^2 and S_{HY}^3 [15]. A comprehensive survey by Li et al. [18] provide a detailed comparison of these measures, pointing out the weaknesses of each one.

3. INTUITIONISTIC FUZZY VECTOR REPRESENTATION OF TRAJECTORIES

Representing trajectory data stored in TD by means of intuitionistic fuzzy sets is a challenging issue. Formally, let $D = \{T_1, T_2, \ldots, T_N\}$ be a TD consisting of a set of trajectories. Assuming linear interpolation between consecutive time-stamped positions, trajectory $T_i = <(x_{i,1}, y_{i,1}, t_{i,1}), \ldots, (x_{i,n_i}, y_{i,n_i}, t_{i,n_i})>$ consists of a sequence of n_i line segments in 3D space, where each segment represents its continuous "development" between positions sampled at time $t_{i,j}$ and $t_{i,j+1}$.

A basic requirement, for applying existing clustering algorithms for point vector data into TD, is to transform trajectories in a space where every T_i is represented as p-dimensional data point. We therefore propose an approximation technique and define the dimensionality of trajectories by dividing the lifespan of each trajectory in p sub-intervals (e.g. 1 minute periods). Regarding the spatial dimension, we assume a regular grid of equal rectangular cells with user-defined size (e.g. 100×100 m^2); in each cell an identifier is also attached. Given this setting, and inspired by the Piecewise Aggregate Approximation (PAA) technique introduced in [16], we propose a method that accepts a trajectory of length n_i as input and produces an approximated trajectory of reduced size p. The algorithm partitions the initial trajectory into p equi-sized temporal periods and substitutes the 3D line segments of the trajectory traversed during this period with the set of the cells crossing the grid and which form a region. More formally:

Definition 3. Given a regular grid G with cells $c_{k,l}$ ($1 \le k \le m$ and $1 \le l \le n$), a trajectory T_i as a sequence of n_i line segments with lifespan $ls_i = t_{i,n_i} - t_{i,1}$ and a target dimension $p << n_i$, approximate trajectory $\overline{T}_i = < r_{i,1}, \ldots, r_{i,p} >$ is the one produced when all trajectory triplets inside each temporal period $p_j = \left[\frac{ls_i \cdot (j-1)}{p}, \frac{ls_i \cdot j}{p} \right]$, $1 \le j \le p$ are replaced with a region $r_{i,j}$ consisting of the set of cells $c_{k,l}$ that T_i crosses during p_j■

The advantage of this technique is that it allows us to view all trajectories in D as vectors of the same user-defined dimensionality. Note that depending on the choice of the spatial and temporal granularity a trajectory may introduce *gaps* (i.e. regions with empty set of cells due to that there is no motion during the particular period of time). Figure 1(a) illustrates a simple linear trajectory and the set of cells that it crosses.

Next, inspired by the approach of Giannotti et al. [13], we model the *Uncertain Trajectory* (*UnTra*) of $\overline{T}_i$ as the one where its regions $r_{i,j}$ have been formed by extending them with those cells around it, such that the ε-buffer [13] of the initial trajectory T_i touches them. Formally:

Definition 4. Given an approximate trajectory $\overline{T}_i = < r_{i,1}, \ldots, r_{i,p} >$ and an uncertainty threshold *epsilon* ε, the $UnTra(\overline{T}_i) = < ur_{i,1}, \ldots, ur_{i,p} >$ is obtained by replacing each region $r_{i,j}$ with an uncertain region $ur_{i,j}$ consisting of the set of cells $c_{k,l}$ that the ε-buffer of T_i crosses during p_j■

Figure 1(b) illustrates the *UnTra* counterpart of Figure 1(a) with ε=1, while Figure 1(c) shows the various uncertain regions (with different colours) when p=5. Without loss of generality in the sequel we assume that all trajectories in D have the same uncertainty threshold ε.

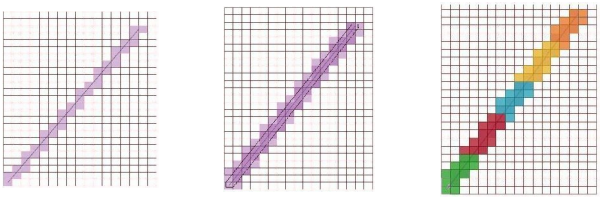


Figure 1 (a) Crossed cells by trajectory (b) UnTra with ε=1 (c) UnTra with p=5

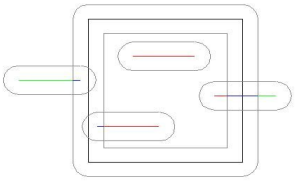


Figure 2 Membership, non-membership and hesitancy in the continuous space

Based on the above representation, in this paper we propose an intuitionistic fuzzy vector representation of a trajectory. The idea is to model each region $ur_{i,j}$ of an *UnTra* as an intuitionistic fuzzy set $A \subset E$ of the regions universe E that belongs to A by a degree $\mu_A(ur_{i,j})$ and does not belong to A by a degree $\gamma_A(ur_{i,j})$. Let us for the moment assume that we work in the continuous space. As we do not have uncertainty in the temporal dimension, each $ur_{i,j}$ is subject to uncertainty only to the spatial dimension. Figure 2 depicts one of the rectangular cells of the spatial grid and two auxiliary buffers (grey cells), one interior and one exterior in distance ε from the cell. There are also the projections of four segments along with their corresponding buffers (also in ε distance from the interpolated segment). The red portion of these segments implies the part of the segment that is *inside* the cell with 100% probability. The green portion implies the part of the segment that is *outside* the cell with 100% probability, while the blue portions are the parts of the segments that we do not know whether they are in or out of the cell. So, the length of the red portion divided to the total length corresponds to the membership of the segment to the cell. Similarly, the green fraction corresponds to non-membership while the blue to hesitancy. Technically, the red portion is the result of the intersection of the interior buffer of the rectangle of the cell with the segment. The green portion is the topological difference of the segment from the exterior buffer of the cell. Note that the previous discussion stands also for any shape other than a rectangular cell, or any movement pattern of a trajectory. Returning to our discretized world, as we assume that after the initial preprocessing we work with trajectory $\overline{T}_i$ that is in a way the set of cells that for sure are crossed by the interpolated trajectory T_i; we can approximate the previous probabilities by counting the number of cells of $r_{i,j}$ and $ur_{i,j}$. Formally:

Definition 5. Given an $UnTra(\overline{T}_i)$, its Intuitionistic counterpart, $I-UnTra(\overline{T}_i)$ is defined as a p-dimensional vector of triplets $I-UnTra(\overline{T}_i) = <(ur_{i,1}, \mu_A(ur_{i,1}), \gamma_A(ur_{i,1})), ..., (ur_{i,p}, \mu_A(ur_{i,p}), \gamma_A(ur_{i,p}))>$ where each triplet ($ur_{i,j}$, $\mu_A(ur_{i,j})$, $\gamma_A(ur_{i,j})$) associates each uncertain region $ur_{i,j}$ with the membership (non- membership) to the fuzzy set A that the trajectory has (not) traversed with 100% probability this region $\mu_A(ur_{i,j})$, ($\gamma_A(ur_{i,j})$), that in their turn are defined by the following equations:

$$\mu_A(ur_{i,j}) = |r_{i,j}| / |UnTra(\overline{T}_i)|, \quad (1)$$

$$\gamma_A(ur_{i,j}) = \left(|UnTra(\overline{T}_i)| - |ur_{i,j}|\right) / |UnTra(\overline{T}_i)| \quad (2)$$

and $|...|$ notates the number of cells of $UnTra(\overline{T}_i)$, $r_{i,j}$ and $ur_{i,j}$.■

Similarly, the hesitancy $\pi_A(ur_{i,j})$, namely, the degree that we are not sure whether the trajectory has passed or not from $ur_{i,j}$, is given by the following equation:

$$\pi_A(ur_{i_j}) = \left(|UnTra(\overline{T}_i)| - |r_{i_j}|\right) / |UnTra(\overline{T}_i)| \quad (3)$$

Note that it is a straightforward task to prove the intuitionistic property that $\pi_A(ur_{i_j}) = 1 - \mu_A(ur_{i_j}) - \gamma_A(ur_{i_j})$.

4. A DISTANCE METRIC FOR I-UnTra

In this section we propose a novel distance (dis-similarity) metric that can be applied between *I-UnTra*. The key observation is that such a metric can be decomposed in two parts, one measuring the distance between the sequences of regions of the two trajectories (D_{UnTra}), and the other measuring the distance between intuitionistic fuzzy sets, based only on the corresponding membership and non-membership values (D_{IFS}). Then having these two different metrics we can combine them into a single one. For example the two distances D_{UnTra} and D_{IFS} may be weighted with parameters W_{UnTra} and W_{IFS} such that $W_{UnTra} + W_{IFS}=1$. In this connection the total distance D_{total} between two *I-UnTra* A and B can be expressed as follows:

$$D_{total}(A,B) = |A-B|_{IFS}^{UnTra} = W_{UnTra} \cdot D_{UnTra}(A,B) + W_{IFS} \cdot D_{IFS}(A,B) \quad (4)$$

It is evident that given that D_{UnTra} and D_{IFS} satisfy the metric space properties the distance D_{total} is also a metric.

4.1 A Distance Metric for Sequences of Regions

In this section, we propose an appropriate modification of the Edit distance with Real Penalty (ERP) [5] as the mean to measure the distance D_{UnTra} between two *UnTra*. Among the several proposals in the literature, we chose to modify ERP, as the Euclidean distance has been proved to have poor performance at the presence of noise and local time shift, the LCSS [25], DTW [28] and EDR [6] do not satisfy the metric space properties. Below we give the definition of the distance between two regions (i.e. vectors of cells) that is the building element of the D_{UnTra} definition.

Definition 6. Given two uncertain regions ur_i and ur_j their distance $|ur_i - ur_j|_d$ is defined in two different versions using two different distances d={*min*, *ext*} between their corresponding *Minimum Bounding Rectangles* (*mbr*):

$$|ur_i - ur_j|_{min} = \min \left\| mbr(ur_i) - mbr(ur_j) \right\|_2, \quad (5)$$

namely the minimum Euclidean distance of the MBRs, and,

$$|ur_i - ur_j|_{ext} = 1 - \frac{1}{2}\left(\frac{ext_x(mbr(ur_i)) + ext_x(mbr(ur_j))}{2 \cdot ext_x(mbr(ur_i \cup ur_j))} + \frac{ext_y(mbr(ur_i)) + ext_y(mbr(ur_j))}{2 \cdot ext_y(mbr(ur_i \cup ur_j))} \right), \quad (6)$$

where e.g. $ext_x(mbr(ur_i))$ is the extent of the *mbr* of ur_i along the x axis.■

It is self-evident that $|ur_i - ur_j|_{ext}$ always results into [0,1]. Intuitively, it takes into account not only the Euclidean distance between regions, but also their extents, while it produces non-zero results in the case of overlapping regions; in the latter case, $|ur_i - ur_j|_{min}$ yields zero. Therefore, one may choose $|ur_i - ur_j|_{ext}$ instead of $|ur_i - ur_j|_{min}$ when refinement into the details of the ur_i, ur_j is desired. Finally, in order for the $|ur_i - ur_j|_{min}$ to range also in [0,1] it should be divided by the maximum possible distance of two regions, that is the distance between the two diagonal cells (i.e. the bottom left and the upper right) of the grid. Now the distance between *UnTras* (D_{UnTra}) is:

Definition 7. Given a regular grid G with cells $c_{k,l}$ ($1 \le k \le m$ and $1 \le l \le n$) and two *UnTra*, $UnTra(\overline{T}_i)$ and $UnTra(\overline{T}_j)$, the D_{UnTra} between them is defined as:

$$D_{UnTra}\left(UnTra(\overline{T}_i), UnTra(\overline{T}_j)\right) = \min \left\{ \begin{array}{l} D_{UnTra}\left(Rst\left(UnTra(\overline{T}_i)\right), Rst\left(UnTra(\overline{T}_j)\right)\right) + |ur_{i,1} - ur_{j,1}|_d, \\ D_{UnTra}\left(Rst\left(UnTra(\overline{T}_i)\right), UnTra(\overline{T}_j)\right) + |ur_{i,1} - gap|_d, \\ D_{UnTra}\left(UnTra(\overline{T}_i), Rst\left(UnTra(\overline{T}_j)\right)\right) + |ur_{j,1} - gap|_d \end{array} \right\} \quad (7)$$

where *Rst* denotes the remaining regions of the *UnTra*. Similarly with [5] where the *gap* (i.e. the deletion operation in terms of string edit distance) element is defined to be 0, i.e. the first value of the time scale for the time series, we define gap as the region containing the first cell of our grid (i.e. cell $c_{1,1}$). ■

Note that as all *UnTra* have the same dimensionality, *gap* regions may be introduced not due to difference in lengths rather than the non definition of the motion of an individual during this particular period. Next we present Lemma 1, necessary to prove Theorem 1 providing the metric space property of D_{UnTra}.

Lemma 1 For any three regions ur_q, ur_i, ur_j, any of which may be a gap region, it is $\left|ur_q - ur_j\right|_d \le \left|ur_q - ur_i\right|_d + \left|ur_i - ur_j\right|_d$.

Theorem 1 Let *A*, *B*, *C* be three *UnTras*, then $D_{UnTra}(A,C) \le D_{UnTra}(A,B) + D_{UnTra}(B,C)$ also stands.

Proof: As Lemma 1 stands true (it is straightforward to prove that $\left|ur_i - ur_j\right|_{MIN}$ and $\left|ur_i - ur_j\right|_{EXT}$ satisfy the metric space properties), according to Waterman et al. [27], D_{UnTra} also satisfies the triangle inequality.■

We have to note here that any meaningful metric distance between regions can be plugged in to the D_{UnTra} without loss of any of its properties. For example one may use even a more refined function than $\left|ur_i - ur_j\right|_{ext}$ by looking not only their MBRs, rather than the actual cells that compose them. In this case, an intuitive choice would be the ERP distance between these sequences of cells similarly with the definition of D_{UnTra}, with the difference that the elementary distance between cells would be the Euclidean distance of their centroids.

4.2 A Distance Metric for Intuitionistic Fuzzy Sets

Here we devise a method to measure the similarity between intuitionistic fuzzy sets, based on the membership and non-membership values of their elements. Given an intuitionistic fuzzy set *A* we define three fuzzy sets, namely M_A, Γ_A, $\Pi_A \in F(E)$ where $F(E)$ is the set of all fuzzy subsets of an element $x \in E$. The membership, non-membership and hesitancy of these sets is defined as $M_A=\{\mu_A(x)\}$, $\Gamma_A=\{\gamma_A(x)\}$, $\Pi_A=\{\pi_A(x)\}$ $\forall x \in E$. In this connection, *A* can be described by the triplet (M_A, Γ_A, Π_A).

Definition 8. Considering two intuitionistic fuzzy sets $A=(M_A, \Gamma_A)$ $B=(M_B, \Gamma_B)$, where M_A, M_B, Γ_A, $\Gamma_B \in F(E)$, and *E* as a finite universe $E=\{x_1,x_2,..,x_n\}$, we define the similarity measure *Z* between the intuitionistic fuzzy sets *A* and *B* by the following equation:

$$Z(A,B) = \tfrac{1}{3}\left(z(M_A,M_B) + z(\Gamma_A,\Gamma_B) + z(\Pi_A,\Pi_B)\right) \quad (8)$$

where

$$z(A',B') = \begin{cases} \dfrac{\sum_{i=1}^{n}\min\left(A'(x_i),B'(x_i)\right)}{\sum_{i=1}^{n}\max\left(A'(x_i),B'(x_i)\right)}, & A' \cup B' \neq \varnothing \\ 1, & A' \cup B' = \varnothing \end{cases} \quad (9)$$

with $A', B' \in F(E)$. ■

In order to accept *Z* as a metric Lemma 2 that follows provides that *z* satisfies the triangular inequality, as it is straightforward to prove that the other metric properties are satisfied by *z*.

Lemma 2. For all $A', B', C' \in F(E)$, where $F(E)$ is the set of all fuzzy subsets of an element $x \in E$ and considering *E* as a finite universe $E=\{x_1,x_2,..,x_n\}$, if $A' \subseteq B' \subseteq C'$ then $z(A',C') \le z(A',B')$ and $z(A',C') \le z(B',C')$.

Proof of lemma 2. By $A' \subseteq B' \subseteq C'$ we have that $A'(x_i) \le B'(x_i) \le C'(x_i)$ and

$$z(A',C') = \frac{\sum_{i=1}^{n}\min(A'(x_i),C'(x_i))}{\sum_{i=1}^{n}\max(A'(x_i),C'(x_i))} = \frac{\sum_{i=1}^{n}A'(x_i)}{\sum_{i=1}^{n}C'(x_i)},$$

$$z(A',B') = \frac{\sum_{i=1}^{n}\min(A'(x_i),B'(x_i))}{\sum_{i=1}^{n}\max(A'(x_i),B'(x_i))} = \frac{\sum_{i=1}^{n}A'(x_i)}{\sum_{i=1}^{n}B'(x_i)},$$

$$z(B',C') = \frac{\sum_{i=1}^{n}\min(B'(x_i),C'(x_i))}{\sum_{i=1}^{n}\max(B'(x_i),C'(x_i))} = \frac{\sum_{i=1}^{n}A'(x_i)}{\sum_{i=1}^{n}B'(x_i)}.$$

Thus, $\frac{\sum_{i=1}^{n}A'(x_i)}{\sum_{i=1}^{n}C'(x_i)} \le \frac{\sum_{i=1}^{n}A'(x_i)}{\sum_{i=1}^{n}B'(x_i)}$, $\frac{\sum_{i=1}^{n}A'(x_i)}{\sum_{i=1}^{n}C'(x_i)} \le \frac{\sum_{i=1}^{n}B'(x_i)}{\sum_{i=1}^{n}C'(x_i)}$

hence, $z(A',C') \le z(A',B')$ and $z(A',C') \le z(B',C')$

Since $A,B,C \in IFSs(E)$ and $A \subseteq B \subseteq C$ we have

$\mu_A(x) \le \mu_B(x) \le \mu_C(x)$ and $\gamma_A(x) \ge \gamma_B(x) \ge \gamma_C(x)$ $\forall x_i \in E, i = 1,2,...,n$

therefore, $z(M_A,M_B)$, $z(\Gamma_A,\Gamma_B)$ and $z(\Pi_A,\Pi_B)$ satisfy all metric properties and so *Z* also satisfies these properties. Thus, *Z* is a similarity metric. ■

The above definitions can be demonstrated by the following simple numeric example: Assuming three sets $A,B,C \in IFSs(E)$ with $A=\{x, 0.4, 0.2\}$, $B=\{x, 0.5, 0.3\}$, $C=\{x, 0.5, 0.2\}$ we want to find whether *B* or *C* is more similar to *A*. Using the equations of Definition 6 we compute the similarity of *B* and *C* to set *A*. $Z(A,B) = (0.4/0.5 + 0.2/0.3 + 0.2/0.4)/3 = 0.65$, and $Z(A,C) = (0.4/0.5 + 0.2/0.2 + 0.3/0.4)/3 = 0.85$, concluding that *C* seems to be more similar to *A* than B.

The proposed intuitionistic similarity measure uses the aggregation of the minimum and maximum membership, non-membership, and hesitancy values. It is simple to calculate, sensitive to small value variations and deals well with all the counter-intuitive cases in which other measures fail. The majority of the similarity measures reviewed in Section 2, fail to result to a valid intuitionistic value for specific cases; some of them result to 0 or 1 suggesting that the compared sets are either totally irrelevant or identical, while it is obvious that this is false, while others result in a high similarity value for obviously different sets. Table 1 presents all the counter-intuitive cases defined in [18], along with the measure calculation for those cases.

Table 1 – Qualitative evaluation between proposed and other similarity measures with counter-intuitive cases

No	*Measure*	*Counter-intuitive cases*	*Measure Values*	*Proposed measure value*
I.	S_C, S_{DC}	$A=\{(x,0,0,1)\}$, $B=\{(x,0.5,0.5,0)\}$	$S_C(A,B)=S_{DC}(A,B)=1$	$Z=0$
II.	S_H, S_{HB}, S_e^p	$A=\{(x,0.3,0.3,0.4)\}$, $B=\{(x,0.4,0.4,0.2)\}$, $C=\{(x,0.3,0.4,0.3)\}$, $D=\{(x,0.4,0.3,0.3)\}$	$S_H(A,B)=S_{HB}(A,B)=S_e^p(A,B)=0.9$ $S_H(C,D)=S_{HB}(C,D)=S_e^p(C,D)=0.9$	$Z(A,B)=0.66$ $Z(C,D)=0.83$
III.	S_H, S_{HB}, S_e^p	$A=\{(x,1,0,0)\}$, $B=\{(x,0,0,1)\}$, $C=\{(x,0.5,0.5,0)\}$	$S_H(A,B)=S_{HB}(A,B)=S_e^p(A,B)=0.5$ $S_H(B,C)=S_{HB}(B,C)=S_e^p(B,C)=0.5$	$Z_1(A,B)=$ $Z_1(B,C)=0$
IV.	S_L and S_S^p	$A=\{(x,0.4,0.2,0.4)\}$, $B=\{(x,0.5,0.3,0.2)\}$, $C=\{(x,0.5,0.2,0.3)\}$	$S_L(A,B)=S_S^p(A,B)=0.95$ $S_L(A,C)=S_S^p(C,D)=0.95$	$Z_1(A,B)=0.65$ $Z_1(A,C)=0.85$
V.	$S_{HY}^1, S_{HY}^2, S_{HY}^3$	$A=\{(x,1,0,0)\}$, $B=\{(x,0,0,1)\}$	$S_{HY}^1(A,B)=S_{HY}^2(A,B)=S_{HY}^3(A,B)=0$	$Z_1(A,B)=0$
VI.	$S_{HY}^1, S_{HY}^2, S_{HY}^3$	$A=\{(x,0.3,0.3,0.4)\}$, $B=\{(x,0.4,0.4,0.2)\}$, $C=\{(x,0.3,0.4,0.3)\}$, $D=\{(x,0.4,0.3,0.3)\}$	$S_{HY}^1(A,B)=S_{HY}^1(C,D)=0.9$ $S_{HY}^2(A,B)=S_{HY}^2(C,D)=0.85$ $S_{HY}^3(A,B)=S_{HY}^3(C,D)=0.82$	$Z_1(A,B)=0.66$ $Z_1(C,D)=0.85$

In case (I) of Table 1 $S_C(A,B)$ and $S_{DC}(A,B)$ imply that A and B are totally similar. The proposed measure in the contrary, suggests that A and B are totally different. In cases (II) and (IV) other measures result in a rather big similarity value while our measure is not that optimistic. In case (II), $Z(A,B)=0.66$ as the hesitancy in set A is a rather large quantity, while $Z(C,D)=0.83$ as hesitancy is constant at 0.3. Moreover, in case (IV) it is obvious that set A is more similar to C than to B (A and C have the same non-membership value), something that other measures do not take into account. In (III), while A, B and C are totally different, all other measures give a similarity value of 0.5. In (VI) they do not recognize that C is more similar to D than A is to B, due to the same hesitancy value of C and D, and in (VII) they do not recognize that A is more similar to C than to B, due to the same non-membership value of A and C.

Table 1 indicates the intuitiveness of the proposed measure; it satisfies all the properties of a similarity metric and does not fail in cases where other measures do. Furthermore, it is to calculate it without exponents or other time consuming functions. Finally, as we need to need to measure distance D_{IFS} between two *I-UnTra* can be expressed as:

$$D_{IFS}(A,B)=1-Z(A,B) \tag{10}$$

5. A NOVEL TRAJECTORY CLUSTERING ALGORITHM

The majority of the proposed clustering methods so far assume that each vector belongs only to one cluster, a reasonable assumption when vectors reside in dense and well-separated clusters. However, in real-world applications where complex input data may form overlapping clusters, the degree of membership of a vector x_k to the i-th cluster u_{ik} is a value in the interval [0,1]. Based on this observation, Bezdek, et al. [3] introduced the FCM algorithm which uses a weighted exponent on the fuzzy memberships. FCM iteratively discovers cluster centroids that minimize a criterion function, which measures the quality of a fuzzy partition. A fuzzy partition is denoted by a $(c\times N)$-dimensional matrix U of reals $u_{ik}\in[0,1]$, $\forall 1\le i\le c$ and $1\le k\le N$, where c and N is the number of clusters and the cardinality of the data vectors, correspondingly. The following constraint is imposed upon u_{ik}:

$$\sum_{i=1}^{c}u_{ik}=1,\ 0<\sum_{k=1}^{N}u_{ik}<N \tag{11}$$

Given this, the FCM objective function has the form:

$$J_m(U,V)=\sum_{i=1}^{c}\sum_{k=1}^{N}(u_{ik})^m d_{ik}^2 \tag{12}$$

where V is a $(p\times c)$-dimensional matrix storing the c centroids, p is the dimensionality of the data, d_{ik} is an A-norm measuring the distance between data vector x_k and cluster centroid v_i, and $m\in[1,\infty)$ is a weighting exponent. The parameter m controls the fuzziness of the clusters. When m approximates 1, FCM performs a hard partitioning as the k-means algorithm does, while as m converges to infinity the partitioning is as fuzzy as possible. There is no analytical methodology for the optimal choice of m. By iteratively updating the cluster centroids and the membership degrees for each feature vectors, FCM iteratively moves the cluster centroids to the "right" location within the data set.

Regarding the centroid calculation, Lee et al.[17] present a first approach to solve this problem in the context of TD, providing the notion of *representative trajectory*. Assuming that movement patterns are more or less straight lines, introduce an averaging technique between segments that works well when trajectories are dense and follow such a linear regression model. However, in this paper we claim that real-world applications involve trajectories that follow circular movement patterns or present large agility. Moreover, trajectories that follow similar routes for only a portion of their lifespan and then they are diverging would result in non representative motions patterns that can not be described by conventional averaging techniques. In order to overpass these obstacles and support real-world requirements, we argue that a better representation can be succeeded if we utilize local criteria (contrary to global via generic distance functions) to decide whether a sub-trajectory is part of the movement pattern. For this reason next we provide a methodology that enables this calculation exploiting local trajectory matches.

5.1 The Centroid Trajectory algorithm

We base our proposal for the centroid trajectory (*CenTra*) estimation on the definition of *I-UnTra*. Our methodology not only overpasses the previously mentioned obstacles, but also, it may be used to represent the *thickness* of the centroid, so as to model the amount of trajectories that contribute to its formation. Towards this goal, we firstly adopt some local similarity function to identify common sub-trajectories (concurrent existence in space-time), and secondly we follow a *region growing* approach so as to represent this local cluster. The idea is to form *CenTra* similar to an *UnTra*, requiring at the same time to satisfy some similarity and density constraints. Formally:

Definition 9. Given a regular grid G with cells $c_{k,l}$ ($1\le k\le m$ and $1\le l\le n$), each of which has density $G(k, l)$ (i.e. the number of distinct trajectories traversing the cell), a density threshold δ, a similarity threshold σ and a set S of p-dimensional $UnTra(\overline{T}_i)$, we define the *CenTra* of S as an *UnTra* whose regions at each period p_j, $1\le j\le p$, corresponds to a *Local CenTra* (*L_CenTra*), which is an *Augmented Region* (*AR*) of a seed region that has been extended "towards" other regions (i.e. sub-trajectories) if and only if (a) the similarity between $ur_{i,j}$ (under examination) regions and

L_CenTra is $similarity(L\_Centra, ur_{i,j}) \geq \sigma$, and (b) adopted regions $AR_{i,j}$ have average density $avg\_density(AR_{i,j}) \geq \delta$.■

```
Algorithm CenTra(set of I-UnTra S, Grid G, Real δ, Real
σ, set of k Most Similar Trajectories k-MST)
01.  CenTra=∅ ;
02.  forall temporal periods p_j
03.    L_CenTra = Init_Local_Centra(p_j);
04.    repeat
05.      forall regions ur_{i,j} in k-MST
06.        AR_{i,j} = L_CenTra extended with ur_{i,j};
07.        AR ={ur_{i,j}|similarity(L_CenTra,ur_{i,j})≥σ
                    and avg_density(AR_{i,j})≥δ;
08.      if AR ≠ ∅
09.        ur_{i,j}=argmax_{regeAR}(similarity(L_CenTra,AR_reg),
                              avg_density(AR_reg));
10.        L_CenTra=AR_{i,j};
11.    until AR ≠ ∅;
12.    CenTra=CenTra ∪ L_CenTra;
13.  return CenTra;
```

Figure 3: CenTra Algorithm

Figure 3 illustrates the developed *CenTra* algorithm. The basic background idea is to perform some kind of time-focused local clustering using a region growing technique under similarity and density constraints. The algorithm for each time period (line 2), determines an initial seed region, (via the *Init_Local_CenTra* (line 3)) and searches for the maximum region that is composed of all sub-trajectories that are similar over σ and dense over δ. The seed region is determined as the one with the minimum average distance from all other candidate regions. Subsequently, the growing process begins (line 4) and the algorithm tries to find the next region to extend (lines 5-6) among the k Most Similar Trajectories (k-MST), as someone would expect to find the *best region* in one of these k regions. Note that searching for the k-MST introduces only a small overhead in the algorithm's execution since the corresponding results are kept in a priority queue that has been fed during the initialization of the seed region (line 3). Then the algorithm searches among the candidates regions, i.e., those that satisfy the similarity and density constraints (line 7), in order to find the best, i.e., the one that has the maximum similarity, or secondly, the one that maximizes the average density after growing (lines 9-10). The whole process is continuing until no more growing can be applied (line 11) and then this local centroid is appended to the *CenTra* (line 12).

5.2 The CenTR-I-FCM algorithm for I-UnTra

Continuing our discussion regarding the adoption of FCM in the context of TD it must be pointed that its direct employment would result to an inefficient scheme, as the initial trajectories should be interpolated to any time instance of all other trajectories as to be transformed to data points, a fact that would prohibitively increase the dimensionality of the problem. More importantly, using an A-norm as the mean to measure the distance between trajectories one expect to encounter all the well-known problems present when measuring the similarity in time series data, as the presence of outliers, different speeds, local shifts, different baselines and scales. Furthermore, FCM tries to partition the dataset by just looking at the vector values ignoring the fact that these vectors may be accompanied by qualitative information (i.e. the uncertainty) which may be given per dimension.

Contrary to these shortcomings our aim is to take advantage of our intuitionistic trajectory representation *I-UnTra*, i.e., the p-dimensional vectors of triplets $(ur_{i,j}, \mu_A(ur_{i,j}), \gamma_A(ur_{i,j}))$. While it is evident that the FCM algorithm can not utilize intrinsically such qualitative information, in this paper, we propose a different perspective by substituting the distance function with the distance metric D_{total} introduced in Section 4. Using the proposed distance function the fuzzy c-means objective function takes the form:

$$J_m^{TR-I-FCM}(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{N}(u_{ik})^m \left|x_k - v_i\right|_{IFS}^{UnTra} \tag{13}$$

Theorem 2. Given V that is a $(p \times c)$-dimensional matrix storing the c centroids trajectories *I-UnTra*, p is the dimensionality of the trajectories, $\left|x_k - v_i\right|_{IFS}^{UnTra}$ is the distance between trajectory x_k and cluster centroid v_i, and $m \in [1,\infty)$ is a weighting exponent; then if m and c are fixed parameters and $I_k, \tilde{I}_k$ are sets defined as:

$$\forall\ 1 \leq k \leq N,\ \begin{cases} I_k = \left\{ i \mid 1 \leq i \leq c;\ \left|x_k - v_i\right|_{IFS}^{UnTra} = 0 \right\}, \\ \tilde{I}_k = \{1,2,...,c\} \setminus I_k, \end{cases}$$

then $J_m^{TR-I-FCM}(U,V)$ may be minimized only if:

$$\underset{\substack{1\leq i\leq c \\ i\leq k\leq N}}{\forall} u_{ik} = \begin{cases} \left(\left|x_k - v_i\right|_{IFS}^{UnTra}\right)^{\frac{1}{1-m}} \Big/ \sum_{j=1}^{c}\left(\left|x_k - v_j\right|_{IFS}^{UnTra}\right)^{\frac{1}{1-m}}, & I_k = \varnothing, \\ \begin{cases} 0, & i \notin I_k \\ \sum_{i\in I_k} u_{ik} = 1, & i \in I_k \end{cases}, \quad I_k \neq \varnothing, \end{cases} \tag{14}$$

and

$$\underset{1\leq i\leq c}{\forall} v_i = \sum_{k=1}^{N}(u_{ik})^m x_k \Big/ \sum_{k=1}^{N}(u_{ik})^m. \tag{15}$$

Proof of theorem 2. The minimization of Eq. (13) can be achieved term by term:

$$J_m^{TR-I-FCM}(U,V) = \sum_{k=1}^{N}\varphi_k(U) \tag{16}$$

where

$$\underset{1\leq k\leq N}{\forall} \quad \varphi_k(U) = \sum_{i=1}^{c}(u_{ik})^m \left|x_k - v_i\right|_{IFS}^{UnTra} \tag{17}$$

The Lagrangian of (17) with constraints from (11) is:

$$\underset{1\leq k\leq N}{\forall} \quad \Phi_k(U,\lambda) = \sum_{i=1}^{c}(u_{ik})^m \left|x_k - v_i\right|_{IFS}^{UnTra} - \lambda\left(\sum_{i=1}^{c}u_{ik} - 1\right) \tag{18}$$

where λ is the Lagrange multiplier. Setting the partial derivatives of $\Phi_k(U,\lambda)$ to zero we obtain:

$$\underset{1\leq k\leq N}{\forall} \quad \frac{\partial\Phi_k(U,\lambda)}{\partial\lambda} = \sum_{i=1}^{c}u_{ik} - 1 = 0 \tag{19}$$

and

$$\underset{\substack{1\leq z\leq c \\ 1\leq k\leq N}}{\forall} \quad \frac{\partial\Phi_k(U,\lambda)}{\partial u_{zk}} = m(u_{zk})^{m-1}\left|x_k - v_i\right|_{IFS}^{UnTra} - \lambda = 0 \tag{20}$$

Solving (20) for u_{zk} we get:

$$u_{zk} = \left(\frac{\lambda}{m}\right)^{\frac{1}{m-1}} \left(\left|x_k - v_z\right|_{IFS}^{UnTra}\right)^{\frac{1}{1-m}} \quad (21)$$

From (19) and (21) we obtain:

$$\left(\frac{\lambda}{m}\right)^{\frac{1}{m-1}} = \frac{1}{\sum_{j=1}^{c}\left(\left|x_k - v_j\right|_{IFS}^{UnTra}\right)^{\frac{1}{1-m}}} \quad (22)$$

The combination of (21) and (22) yields:

$$\underset{\substack{1\le z\le c\\ 1\le k\le N}}{\forall} \quad u_{zk} = \frac{\left(\left|x_k - v_z\right|_{IFS}^{UnTra}\right)^{\frac{1}{1-m}}}{\sum_{j=1}^{c}\left(\left|x_k - v_j\right|_{IFS}^{UnTra}\right)^{\frac{1}{1-m}}} \quad (23)$$

Equation (15) follows with the same way as in [3]. ■

Note that u_{ik} corresponds to the membership of the *k*-th *I-UnTra* to the *i*-th cluster and it is different from the internal intuitionistic fuzzy memberships of each *I-UnTra*. Moreover, after centroids' computation (15) and before the next iteration, where the memberships u_{ik} to the new clusters are updated, we calculate the memberships and non-memberships of the new (virtual) centroid trajectories. At each iteration and for every centroid we extract the membership degree μ_{i_j} of centroid v_i as the average of the memberships of all *I-UnTra* that belong to cluster *i*. Similarly, for the non-membership degrees γ_{i_j}. More formally, if C_i is a set defined as

$$\underset{1\le i\le c}{\forall} \quad C_i = \left\{ k \mid 1\le k\le N; \left|x_k - v_i\right|_{IFS}^{UnTra} < \left|x_k - v_r\right|_{IFS}^{UnTra}, \forall 1\le r\le c \wedge r\ne i \right\}$$

we obtain:

$$\underset{1\le j\le p}{\forall} \quad \mu_{i_j} = \sum_{\forall k\in C_i} \mu_{k_j} \Big/ \left|C_i\right|, \quad \nu_{i_j} = \sum_{\forall k\in C_i} \gamma_{k_j} \Big/ \left|C_i\right| \quad (24)$$

```
Algorithm CenTR-I-FCM (set of I-UnTra S, Real ε, Int c)
01. V(0) = c random I-UnTra; j=1;
02. repeat
03.   Calculate membership matrix U(j); // use (14)
04.   Update the centroids' matrix V(j) using CenTra;
05.   Compute membership and non-membership degrees of
      V(j);  // use (16)
06. While ||U(j+1)-U(j)||F>ε; j=j+1;
```

Figure 4 CenTR-I-FCM algorithm for clustering I-UnTra

Using the update procedure of Eq. (15) in the TD setting we would share the same problems with FCM and k-means. As we are especially interested in the representation of real movement patterns, as the centroid of each cluster may be, the idea is to use the density-based CenTra algorithm instead of this weighted averaging technique. We argue that the adoption of *CenTra* as the update centroid methodology of the result of Theorem 2, will result to more meaningful trajectory clustering. The idea is that the algorithm implied by Theorem 2 iteratively tries to diminish the intra-cluster variance using some global, approximate distance metric, and *CenTra* comes at each iteration to push (i.e. grow) the centroid (only the sub-trajectories and not the whole trajectory) towards *interesting* places, where interestingness in our case means high density and similarity. The incorporation of *CenTra* into FCM (named *Centroid TRajectory Intuitionistic FCM* (CenTR-I-FCM)) is a straightforward task and only takes place at line 4 of the algorithm in Figure 4 with the invocation of *CenTra*.

6. EXPERIMENTAL EVALUATION

In this section, we present the datasets and then the experimental results evaluating our approach. The experiments were run on a PC with Intel Core Duo at 2.53 GHz, 4 GB RAM and 240 GB hard disk. We implemented the proposed algorithms using C++.

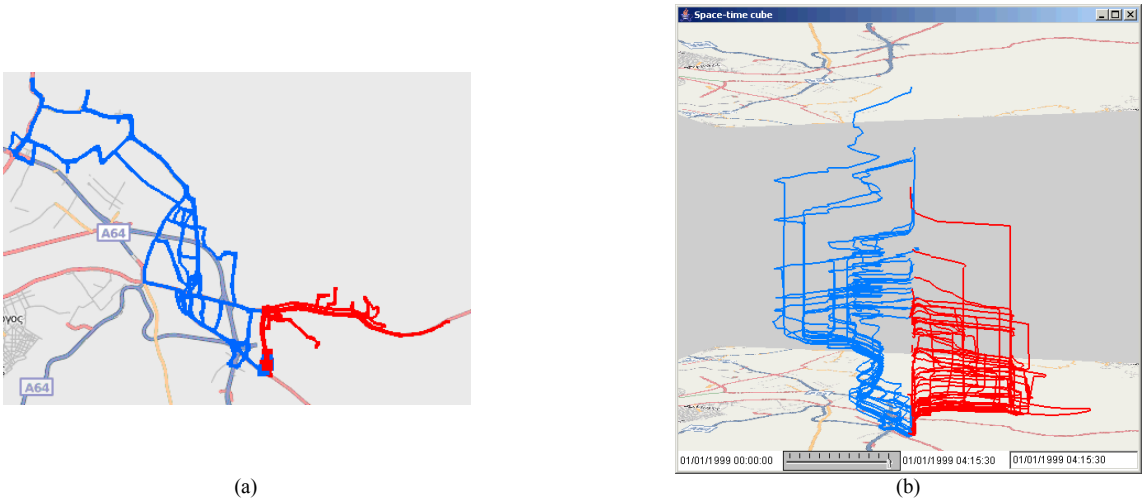
6.1 Datasets

To the best of our knowledge in the TD domain there is no available real dataset grouped in clusters so it could be used as ground truth for benchmarking. Nevertheless, in this paper, we have used a real dataset from which we extracted real clusters. The initial dataset consists of the GPS-tracked positions of 50 trucks transporting concrete in the area of Athens between August and September 2002 (the dataset is publicly available at hhtp://www.rtreeportal.org). There are 112,300 position records consisting of the truck identifiers, dates and times, and geographical coordinates. The temporal spacing is regular and equals 30 seconds. From these raw data, we produced 1100 trajectories by splitting the recordings of a truck in subsets if there was a temporal gap between two consecutive recordings larger than 15 minutes. Subsequently, we used visual analytics tools to identify real clusters, producing thus, four clusters namely, two "opposite" and two "round trips".

More specifically, we discovered two clusters of trajectories where the start and end locations almost coincide, i.e. each truck returned to its original location after performing a round trip; the directions of the trips of the two clusters differ. A two-dimensional snapshot of the two clusters is illustrated in Figure 5(a), while Figure 5(b) illustrates them in the 3D space, where time is represented as the third dimension; the two clusters are formed from the red and blue trajectories. Likewise, we also discovered two clusters of trajectories going to opposite directions, illustrated in a similar way in Figure 6

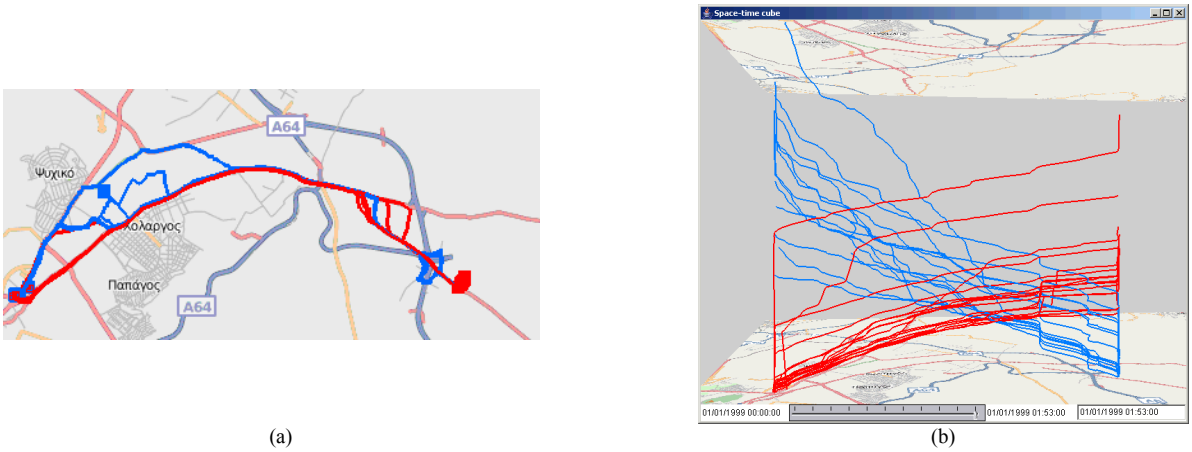
6.2 Experiments

We implemented a variation of the classic FCM algorithm appropriately modified for our needs. In order to be as fair as possible, this algorithm, named TR-FCM, uses our point vector representation of trajectories, along with the minimum distance between MBRs so as to calculate the distance between the cluster's centroid and each candidate trajectory. In our first experiment we employed only the two "opposite" clusters. We then used our CenTR-I-FCM and TR-FCM algorithms varying the grid's *cell size* and ε, and we measured the algorithm's success as the percentage of the correctly classified trajectories. The corresponding results regarding CenTR-I-FCM are illustrated in Figure 8; note that cell Size in Figure 8(a) and (b) is demonstrated as percentage of the size of the total space. Regarding the other experiment's parameters, in Figure 8(a) we fix the value for the density threshold δ to 2% (of the total number of trajectories), while in Figure 8(b), we set ε to 1. In all cases we fix parameters σ to 0.5 and k to the number of trajectories in each cluster. Clearly, as Figure 8 demonstrates, CenTR-I-FCM achieves very good results, with a typical rate above 70%, while it reaches 90% when the cell size is set to its maximum value, regardless of the value of



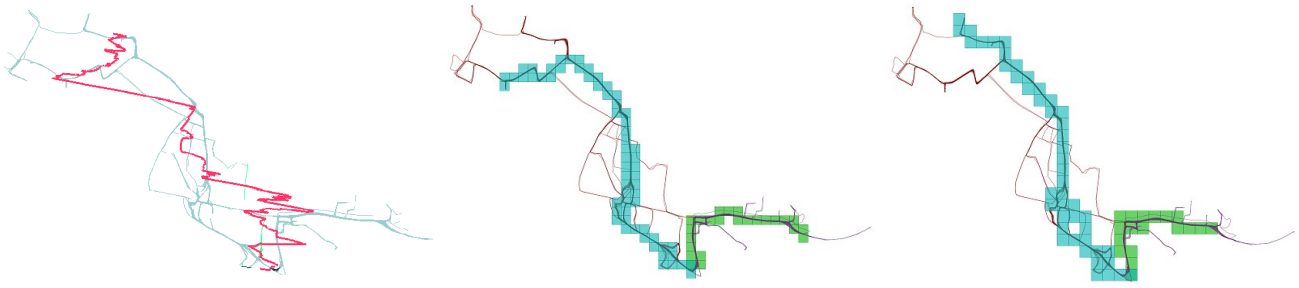
(a) (b)

Figure 5: The two "Round trips" clusters in 2 (a) and 3 dimensions(b)



(a) (b)

Figure 6: The two "Opposite" clusters in 2 (a) and 3 dimensions(b)



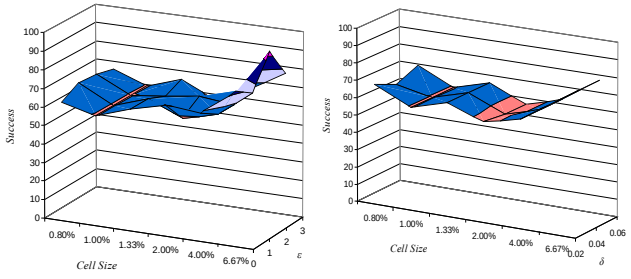
(a) (b) (c)

Figure 7: Representative Trajectories (TRACKLUS) (a) and Centroid Trajectories (CenTra) ((b) and (c)) in 'round trips' dataset

δ and ε, as clustering is performed at a higher granularity level where specific movement details are vanishing. On the other hand, when using the same experimental settings over TR-FCM, it produces rather poor results, with an average success of about 53% regardless of the experimental settings. We also performed the same experiments on the other two clusters (i.e., "round trips"); the respective figures are omitted due to space constraints. Nevertheless, the general observation straightforwardly obtained from this study, is that the CenTR-I-FCM outperforms TR-FCM regardless of the experimental setting, verifying that it produces very good clustering results, with a typical rate above 65%.

In order to study the algorithms' behaviour in cases where more than two clusters are present, we performed another experiment using different portions of the trucks dataset containing three (i.e., the two "round trips" clusters, and one of the "opposites" clusters), and four clusters. The results of this experiment are illustrated in Figure 9(a); again, CenTR-I-FCM clearly outperforms its competitor in terms of successive cluster discovering. On the other hand, the performance of both

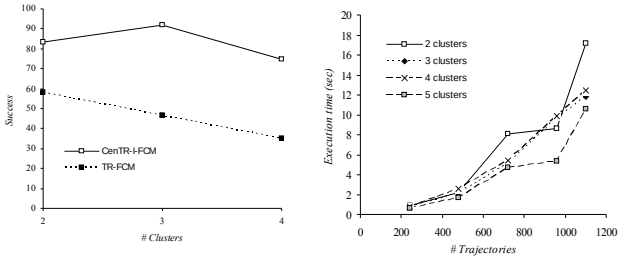
algorithms evidently downgrades as the number of requested clusters increases; however note that the performance of our proposal decreases with a smaller ratio than the respective one of CenTR-I-FCM, remaining in any case above 75%.



(a) (b)

Figure 8: Clustering accuracy scaling the cell size, ε (a) and density threshold, δ (b)

Regarding the performance of the CenTR-I-FCM algorithm, it was evaluated using the whole "trucks" dataset by increasing the trajectory cardinality. The respective results illustrated in Figure 9(b) demonstrate the efficiency of the proposed algorithm for various numbers of clusters requested. It is clear that the algorithm is not depended on the number of clusters, while all curves illustrated in Figure 9(b) imply that the algorithm has super-linear behaviour regarding the dataset cardinality.



(a) (b)

Figure 9: (a) Clustering accuracy scaling the number of clusters (b) TR-I-FCM performance scaling the dataset cardinality

Although starting from different base lines and focusing on different applications, we provide a qualitative evaluation of the results of the CenTra algorithm with the technique of representative trajectory presented in [17]. In Figure 7(a), it is evident that in the TRACKLUS approach the cluster representative (red bold line) does not fit to a real trajectory, rather it lies outside of the real data (light blue cluster), due to its averaging technique. Note that this cluster contains both of the real clusters of the dataset. Obviously, this is the effect for clustering segments instead of whole trajectories. Even considering this, the algorithm does not compass the turn occurring at the bottom of the figure. On the contrary, CenTra (Figure 7(b) & (c), where in (c) we increase the cell size and decrease the density δ), not only resides on the data (due to its density-based approach), but also vanishes the non-interesting movement details (the 'noisy' infrequent parts are not part of the centroid), it catches turns, while it may be thicker in portions that something interesting (i.e. dense-similar subtrajectories) happens.

7. CONCLUSION AND FUTURE WORK

In this paper, we proposed a three-step approach, challenging by clustering and representation issues in TD that are inherently subject to uncertainty. In detail, we proposed an intuitionistic fuzzy vector representation of trajectories, upon which we defined a distance metric that was used to device the CenTR-I-FCM algorithm for clustering trajectories under uncertainty, which uses as its update centroids' technique a novel algorithm that is able discover the centroid trajectory of a bundle of trajectories. The efficiency of our approach has been proved experimentally using a real trajectory dataset.

Clear future work objectives arise from this work. More specifically, we plan to adopt some clever sampling technique for multi-dimensional data as to diminish the effect of initialization in our algorithms. We will also develop an index-based version and perform extensive evaluation using large datasets.

8. REFERENCES

[1] O. Abul, F. Bonchi, M. Nanni, 'Never Walk Alone: Uncertainty for Anonymity in Moving Objects Databases', In *Proc. of ICDE*, 2008.

[2] K.T. Atanassov, 'Intuitionistic Fuzzy Sets: Theory and Applications', *Studies in Fuzziness and Soft Computing*, 35, 1999.

[3] J.C. Bezdek, R. Ehrlich, and W. Full, 'FCM: the Fuzzy c-Means clustering algorithm', *Computers and Geosciences*, 10, 1984.

[4] I. V. Cadez, S. Gaffney, and P. Smyth, 'A general probabilistic framework for clustering individuals and objects', In *Proc. of SIGKDD*, 2000.

[5] L. Chen and R. Ng, 'On the marriage of edit distance and Lp norms', In *Proc. of VLDB*, 2004.

[6] L. Chen, M. Tamer Özsu, and V. Oria, 'Robust and Fast Similarity Search for Moving Object Trajectories', In *Proc. of SIGMOD*, 2005.

[7] S.M. Chen, 'Measures of similarity between vague sets', *Fuzzy Sets and Systems*, 74 (2), 1995.

[8] S.M. Chen 'Similarity measures between vague sets and between elements', *IEEE TSMC*, 27(1), 1997.

[9] L. Dengfeng, C. Chuntian, 'New similarity measure of intuitionistic fuzzy sets and application to pattern recognitions', *Pattern Recognition Letters*, 23, 2002.

[10] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, 'A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise', In *Proc. of KDD,* 1996.

[11] L. Fan, X. Zhangyan, 'Similarity measures between vague sets', *J. Software,* 12(6), 2001 (in Chinese).

[12] S. Gaffney, and P. Smyth, 'Trajectory Clustering with Mixtures of Regression Models', In *Proc. of SIGKDD*, 1999.

[13] F. Giannotti, M. Nanni, D. Pedreschi, F. Pinelli, 'Trajectory Pattern Mining', In *Proc. of SIGKDD*, 2007.

[14] D.H. Hong, C. Kim, 'A note on similarity measures between vague sets and between elements', *Inform. Science*, 115, 1999.

[15] W.-L. Hung, M.-S. Yang, 'Similarity measures of intuitionistic fuzzy sets based on Hausdorff distance', *Pattern Recognition Letters*, 25, 2004.

[16] E. J. Keogh and M. J. Pazzani. 'A simple dimensionality reduction technique for fast similarity search in large time series databases'. In *Proc. of PAKDD*, 2000.

[17] J.-G. Lee, J. Han, and K.-Y. Whang, 'Trajectory clustering: a partition-and-group framework'. In *Proc. of SIGMOD*, 2007.

[18] Y. Li, D.L. Olson, Z. Qin, 'Similarity measures between vague sets: A comparative analysis', *Pattern Recognition Letters*, 28, 2007.

[19] Y. Li, C. Zhongxian, Y. Degin, 'Similarity measures between vague sets and vague entropy. *J.Computer Science*. 29(12), 2002 (in Chinese).

[20] S. Lloyd, 'Least Squares Quantization in PCM', *IEEE Trans. Information Theory*, 28(2), 1982.

[21] H.B. Mitchell, 'On the Dengfeng–Chuntian similarity measure and its application to pattern recognition', *Pattern Recognition Letters*, 24, 2003.

[22] N. Pelekis, I. Kopanakis, I. Ntoutsi, G. Marketos, G. Andrienko and Y. Theodoridis. 'Similarity Search in Trajectory Databases'. In *Proc. of TIME*, 2007.

[23] D. Pfoser, and C. S. Jensen, 'Capturing the Uncertainty of Moving-Object Representations'. In *Proc. of SSD*, 1999.

[24] G. Trajcevski, O. Wolfson, K. Hinrichs, and S. Chamberlain, 'Managing uncertainty in moving objects databases', *ACM TODS*. 29(3), 2004.

[25] M. Vlachos, G. Kollios, and D. Gunopulos, 'Discovering Similar Multidimensional Trajectories', *Proc. of ICDE*, 2002.

[26] W. Wang, J. Yang, and R. R. Muntz, 'STING: A Statistical Information Grid Approach to Spatial Data Mining', In *Proc. of VLDB*, 1997.

[27] M.S. Waterman, T.F. Smith, and W.A. Beyer, 'Some biological sequence metrics', Advances in Mathematics, 20(4), 1976.

[28] B-K Yi, H. Jagadish, and C. Faloutsos, 'Efficient Retrieval of Similar Time Sequences under Time Warping'. In *Proc. of ICDE*, 1998.

[29] L.A. Zadeh, 'Fuzzy sets', *Information Control*, 8, 1965.

[30] T. Zhang, R. Ramakrishnan, and M. Livny, 'BIRCH: An Efficient Data Clustering Method for Very Large Databases', In *Proc. of SIGMOD*, 1996.

[31] L. Zhizhen, S. Pengfei, 'Similarity measures on intuitionistic fuzzy sets', *Pattern Recognition Letters*. 24, 2003.