Mining Trajectory Databases via a Suite of Distance Operators

Nikos Pelekis[1], Ioannis Kopanakis[2], Irene Ntoutsi[1], Gerasimos Marketos[1], and Yannis Theodoridis[1,3]

[1] Dept. of Informatics, Univ. of Piraeus, Greece

[2] Tech. Educational Institute of Crete, Greece

[1,3] Computer Technology Institute, Patras, Greece

ICDE workshop on Spatio-Temporal Data Mining (STDM07), Istanbul, Turkey, 20 April 2007

Outline

- Mining & Similarity Search in Trajectory Databases
  - Problem statement
  - Related Work
  - Motivation
- A framework of semantically different distance operators
  - (Time-relaxed) Spatial Trajectory Similarity
  - (Time-aware) Spatiotemporal Trajectory Similarity
  - Speed-pattern based Similarity
  - Directional Similarity
- Experimental study

Problem Statement

- Advanced LBS would involve moving object trajectories
  - Common queries: **range** and **nearest-neighbor** (what-is-around, find-the-nearest etc. services)
- KDD - extracting knowledge (e.g. classification & clustering tasks) from trajectory databases
  - the notion of some kind of **distance** function
- Formally:

> Let D be a database of trajectories T_i and Q be a (reference) trajectory consisting of a set of 3D Line Segments.
>
> The *Most-Similar-Trajectory* (MST) S in D with respect to Q is the one that minimizes a distance measure $Dist(Q, T_i)$.

Related Work

- Most approaches inspired by the **time series analysis** domain [AFS99], [KJF97], [CF99].
- Other approaches deal with basic trajectory features [VGD02], [VGK02], [VKG02], [LS05], [CN04], [COO05]
  - different sampling rates, different speeds
  - possible outliers
  - different scaling factors, different trajectory lengths, local time shift.
- Common characteristic of previous works
  - interested in the movement **shape** of the trajectories, usually considered as 2D time series.
  - measure the similarity by just considering the sequences of the sampled positions.
  - temporal dimension is **ignored**, leaving the time recordings out of the KDD process.

Motivation

- Real world: trajectories are represented by finite sequences of time-referenced locations.
- Such sequences may result from various approaches [AAP+07]
  - **time-based** (e.g. every 30 seconds),
  - **change-based** (e.g. when the location of an entity deviates from the previous one by a given threshold),
  - **location-based** (e.g. when a moving object is close to a sensor),
  - **event-based recording** (e.g. when a user requests for localization)
- derived parameters of motion are introduced
  - speed, acceleration, direction, etc.
- A different perspective is required …

Contribution

- We introduce a framework consisting of powerful **distance operators**
  - semantically different properties of trajectories, such as locality, temporality, directionality, rate of change, are taken into consideration.

- ***(time-aware) spatiotemporal similarity***: Find clusters of objects that follow similar *routes* (i.e., projections of trajectories on 2D plane) during the same time interval (e.g. co-location and co-existence from 3pm to 6 pm)
- ***(time-relaxed) spatial similarity***: Find clusters of moving objects taking only their *route* into consideration (i.e., irrespective of time, direction and sampling rate).

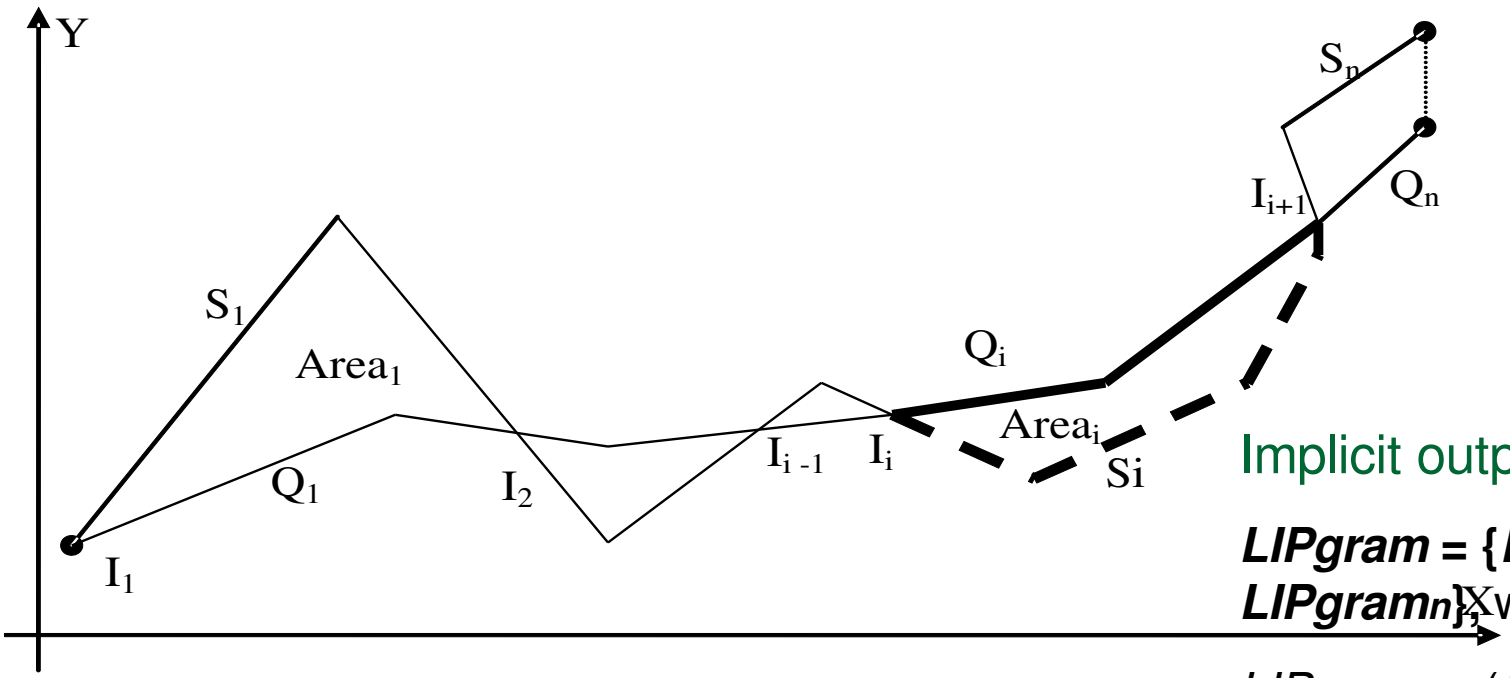
and variations

- ***speed-pattern based spatial similarity***: Find clusters of objects that follow similar routes and, additionally, move with a similar speed pattern, and
- ***directional similarity***: Find clusters of objects that follow a given direction pattern (e.g. NE during the first half of the route and subsequently W).

(Time-relaxed) Spatial Trajectory Similarity

- **Operator:** Locality In-between Polylines distance (LIP)

$$LIP(Q,S) = \sum_{\forall\, polygon_i} Area_i \cdot w_i \quad where\ w_i = \frac{Length_Q(I_i, I_{i+1}) + Length_S(I_i, I_{i+1})}{Length_Q + Length_S}$$

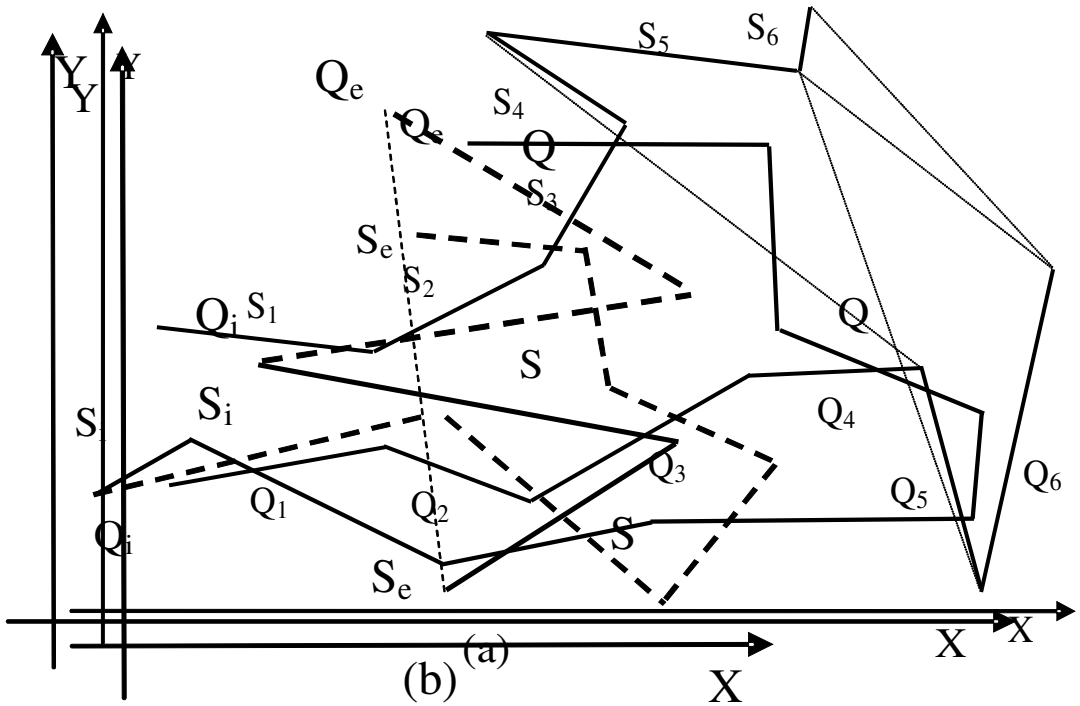


Implicit output of LIP:

***LIPgram* = {*LIPgram*1, …, …, *LIPgramn*}**, where

*LIPgram*i = (*I*i, *I*i+1, *LIP*i)

Special cases for LIP



LIP criterion: the segment implied between the ending points of the currently investigated segments crosses none of the previous segments of Q and S

GenLIP algorithm

```
Algorithm GenLIP(Q polyline, S polyline, p int)
1.  WHILE q < Q.LAST AND s < S.LAST
2.    IF intersect(Q_q, S_s) THEN
3.      Mark Q_q,S_s as 'good' & add them to Q',S'
4.    ELSIF NOT Bad(Q', S', Q_q, S_s) THEN
5.      Mark Q_q,S_s as 'good' & add them to Q',S'
6.    ELSE
7.      Q_q, S_s are marked as 'bad'
8.      FOR k=1 to p
9.        Give p chances to repair LIP criterion
10.     NEXT
11.     IF repairing attempt succeeded THEN
12.       GOTO line 1 with policy-dependant q,s
13.     ELSE
14.       Recover from attempt and GOTO line 17
15.     END IF
16.  NEXT
17.  result = result + LIP(Q', S')
18.  Q = Q - Q'
19.  S = S - S'
20.  RETURN result + GenLIP(Q, S, p)
```

O(*M*og*N*) time complexity

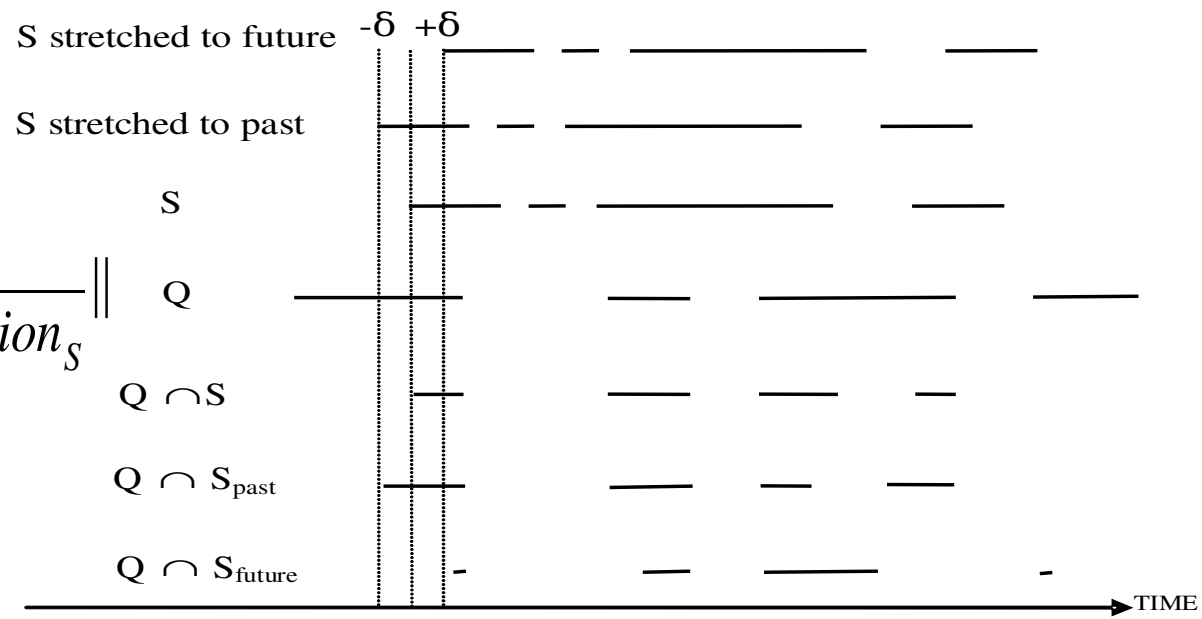
(Time-aware) Spatiotemporal Trajectory Similarity

- **Operator:** Spatiotemporal LIP distance (STLIP)

$$STLIP(Q,S,k,\delta) = \sum_{\forall\, polygon_i} STLIP_i$$

$$STLIP_i = LIP_i \cdot \left(1 + k \cdot TLIP_i\right), \text{ where } k \geq 0$$

$$TLIP_i = \left\| 1 - 2\frac{MDI_i(\delta)}{Duration_Q + Duration_S} \right\|$$



Speed-pattern based Similarity

- **Operator:** Speed-Pattern LIP distance (SPLIP).

$$SPSTLIP(Q, S, k, l, \delta) = \sum_{\forall\, polygon_i} SPSTLIP_i$$

$$SPSTLIP_i = LIP_i \cdot \left(1 + k \cdot TLIP_i\right) \cdot \left(1 + l \cdot SPLIP_i\right)$$

$$SPLIP_i = \frac{\left\| LQ_{Qp_i} - LS_{Qp_i} \right\|}{LQ_{Qp_i}}$$

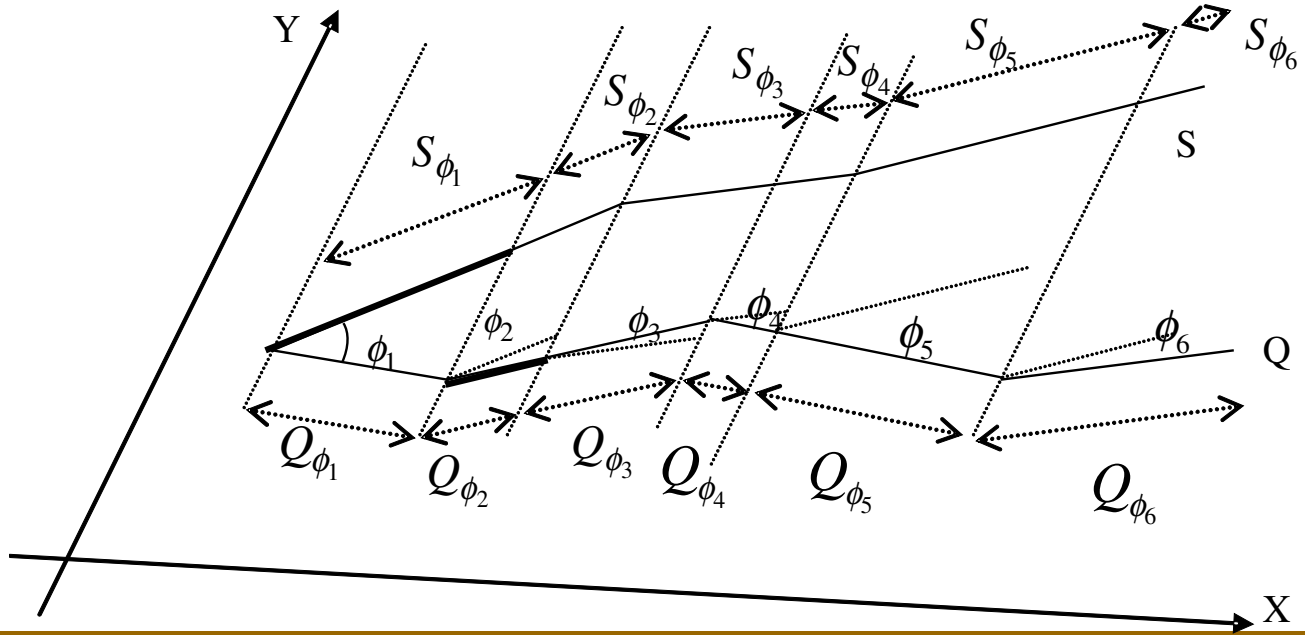
(Time-relaxed) Directional Similarity

- **Operator:** Directional Distance (DDIST)

$$DDIST(Q,S)=\sum_{\forall \varphi_i} DDIST_{\varphi_i}$$

$$DDIST_{\varphi_i}=\frac{\varphi_i}{\Pi} ¿ w_i$$

$$w_i=\frac{length(Q_{\varphi_i})+length(S_{\varphi_i})}{length(Q)+length(S)}$$



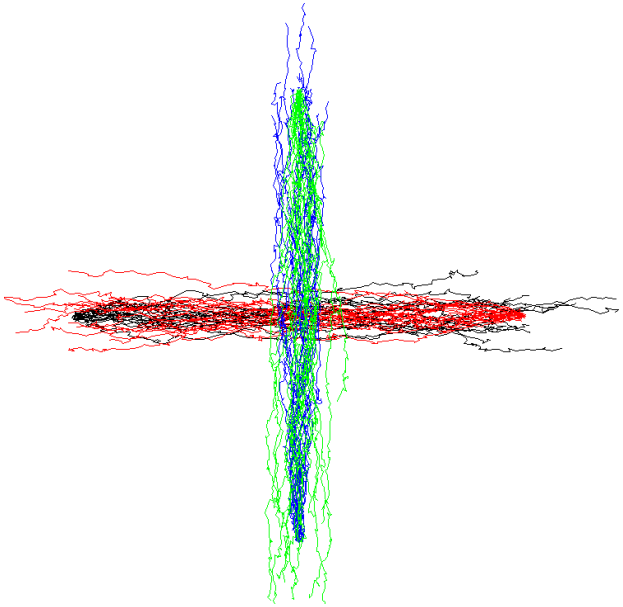
(Time-aware) Directional Similarity

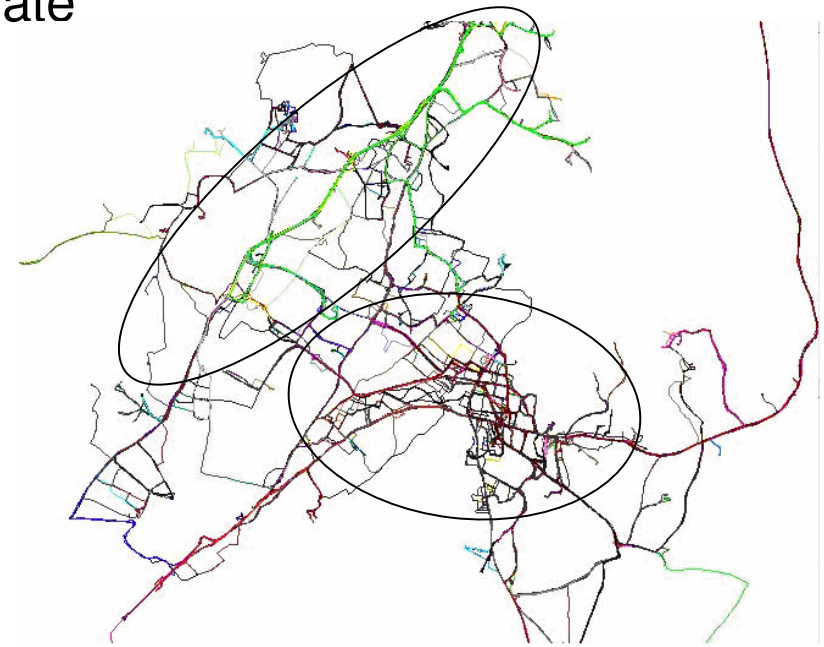
- **Operator:** Temporal Directional distance (TDDIST)

$$TDDIST(Q,S) = \frac{\sum_{\forall Q_i} DDIST_{\varphi_i}(Q_i, S_{Q_i})}{¿i}$$

Experimental Study – Datasets

- Real data - fleet of trucks (276) available in [The]
  - Manual extraction of 2 clusters (i.e. E→N→W→S & N→E patterns)
- Synthetic datasets - generated by the GSTD data generator [TSN99]
  - Manual incorporation of Gausian noise
  - Manual increase of their sampling rate

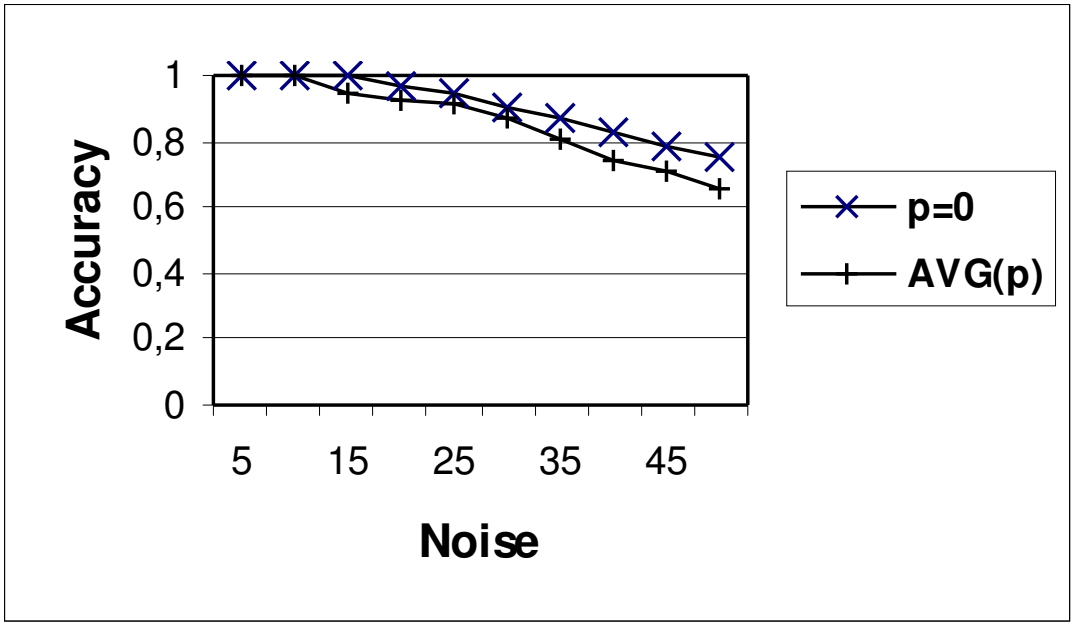




GenLIP Quality

- "Leave-One-Out" classification introduced by Keogh et al. [**KK02**].
- Usage of the datasets having noise *N*i, *S*i, *W*i and *E*i with i = 5, 10, …, 50.
- **Experiment idea**: confuse GenLIP by interleaving routes with noise that introduce larger polygons and more *bad* segments than the initial.
- **Results**: it presents zero misses up to 25% noise. Even adding more noise, the average classification error rate does not exceed 12.5% (i.e. 10 / 80 misses).

- **Inter-cluster quality**: For each route in any of the two clusters we apply (k–1)-NN (k is the number of the routes in each cluster) queries, and we sum all the correct classifications inside the k–1 nearest neighbors.



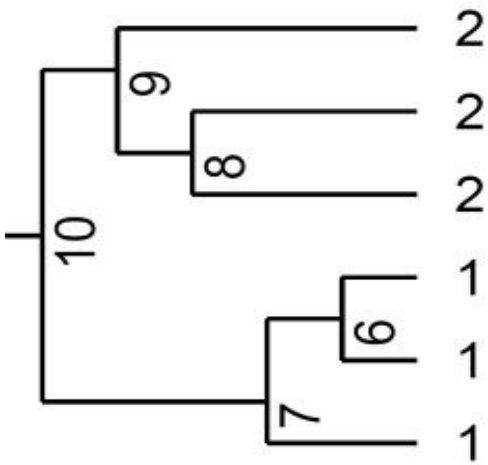
Experiments on Spatiotemporal Similarity 1/3

- Random selection of 10 trucks, which were compressed using the TD-TR algorithm described in [**MB04**] producing similar but not identical artificial trajectories. We applied the TD-TR compression technique with parameter values of *p* in the set {0.02%, 0.05%, 0.1%, 0.15%, 0.2%, 1%, 2%, 5%, 10%} of the length of each trajectory.

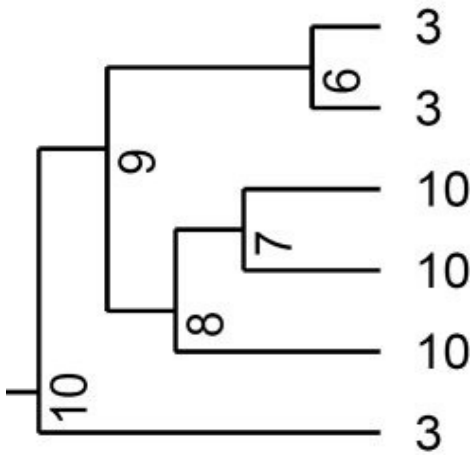
Experiments on Spatiotemporal Similarity 2/3

- We formed 10 datasets of 10 clusters each, one for each trajectory, where one dataset is different from the other only in the number of trajectories per cluster.
- For each dataset, we got all possible pairs of clusters (i.e., 45 cluster pairs) and we partitioned them into two clusters applying agglomerative hierarchical clustering.

Correct clustering

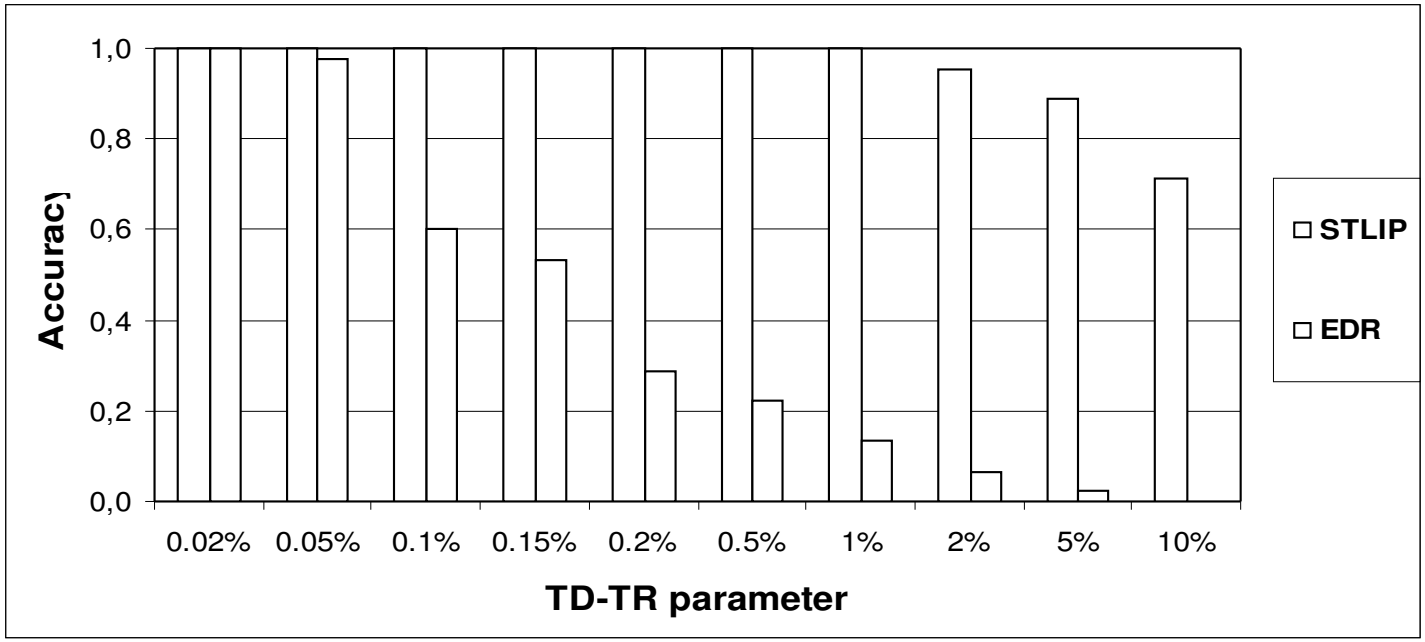


Erroneous clustering



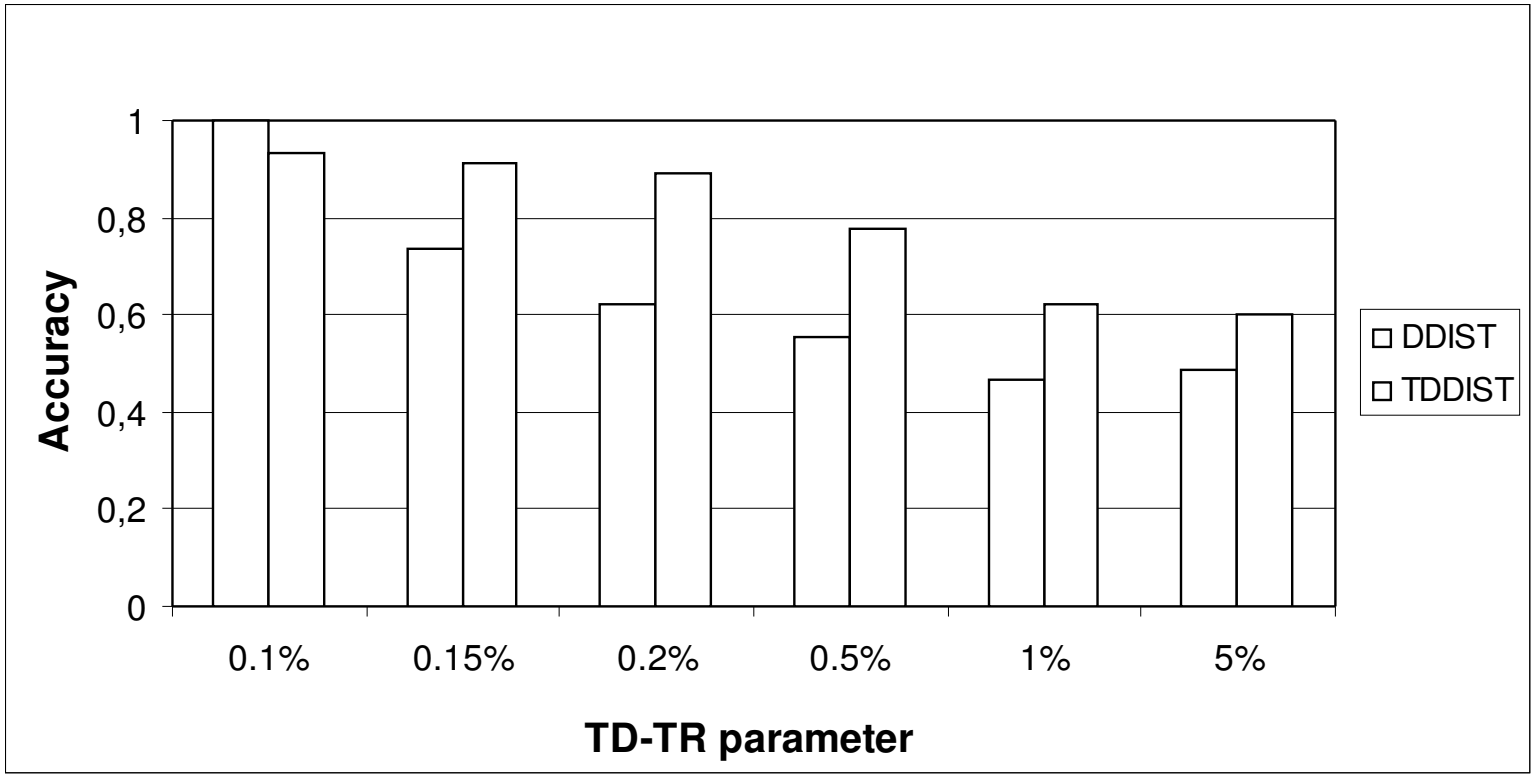
Experiments on Spatiotemporal Similarity 3/3

- Comparison with EDR [COO05], which can identify the NN of the query trajectory and temporarily/initially identify the correct cluster at the lower levels of the dendrogram.
- **However**, at the end it fails in detecting similar trajectories of almost the same length which have been sampled differently.



Experiments on Directional Similarity

- The same experiment as previously for a subset of the produced datasets.



Conclusions

- We proposed novel distance operators, to address different versions of the so-called **trajectory similarity search** problem that could support knowledge discovery in TD.
- To the best of our knowledge, this is the first work that decomposes the problem into different types of similarity queries based on various motion parameters of the trajectories.
- The synthesis of the operators under a unified trajectory management framework provides functionality so far unmatched in the literature.
- The efficiency and robustness of the operators have been proved experimentally by performing clustering and classification tasks to both real and synthetic trajectory datasets.

Future Work

- We plan to devise appropriate indexing structures in order to improve the overall performance of the operators,
- Further qualitative evaluation of the operators.
- Study the quality of *LIPgrams* and utilize these similarity meta-data patterns so as to perform other mining tasks.
- Investigation of extending our techniques to address the problem of similarity search for trajectories restricted in spatial networks.

References 1/2

[AAP+07] N. Andrienko, G. Andrienko, N. Pelekis, and S. Spaccapietra, "Basic Concepts of Movement Data", chapter in F. Giannotti and D. Pedreschi (eds.) *Geography, Mobility and Privacy: A Knowledge Discovery Vision*, Springer, 2007, to appear.

[AFS99] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient Similarity Search in Sequence Databases", Proceedings of *FODO*, 1993.

[CF99] K.P. Chan and A.W-C Fu, "Efficient time series matching by Wavelets", Proceedings of *ICDE*, 1999.

[CN04] L. Chen and R. Ng, "On the marriage of edit distance and Lp norms", Proceedings of *VLDB*, 2004.

[COO05] L. Chen, M. Tamer Özsu, and V. Oria, "Robust and Fast Similarity Search for Moving Object Trajectories", Proceedings of *ACM SIGMOD*, 2005.

[KK02] E. Keogh and S. Kasetty "On the need for time series data mining benchmarks: a survey and empirical demonstration". Proceedings of *SIGKDD*, 2002.

[KJF97] F. Korn, H. Jagadish, and C. Faloutsos, "Efficiently Supporting Ad hoc Queries in Large Datasets of Time Sequences", Proceedings of ACM *SIGMOD*, 1997.

[MB04] N. Meratnia and R.A. de By, "Spatiotemporal Compression Techniques for Moving Point Objects", Proceedings of *EDBT*, 2004.

References 2/2

[The] Y. Theodoridis, “R-tree Portal”, *www.rtreeportal.org* (URL valid on February 12, 2007).

[TSN99] Y. Theodoridis, J. R. O. Silva, and M. A. Nascimento, “On the Generation of Spatio-temporal Datasets”, Proceedings of *SSD*, 1999.

[VGD02] M. Vlachos, D. Gunopulos, and G. Das, “Rotation Invariant Distance Measures for Trajectories”, Proceedings of *SIGKDD*, 2002.

[VGK02] M. Vlachos, D. Gunopulos, and G. Kollios, “Robust Similarity Measures for Mobile Object Trajectories”, Proceedings of *MDDS*, 2002.

[VKG02] M. Vlachos, G. Kollios, and D. Gunopulos, “Discovering Similar Multidimensional Trajectories”, Proceedings of *ICDE*, 2002.

[LS05] B. Lin, and J. Su, “Shapes Based Trajectory Queries for Moving Objects”, Proceedings of *ACM GIS*, 2005.