Indexing Uncertain Spatio–Temporal Data

Tobias Emrich, Hans–Peter Kriegel, Matthias Renz, Andreas Züfle (LMU)
Nikos Mamoulis (HKU)
Agenda

- Spatio–Temporal Data
- Uncertain Spatio–Temporal Data
- Modeling Uncertain Spatio–Temporal Data
- Indexing Uncertain Spatio–Temporal Data
- Probabilistic Pruning
- Experiments
What is (certain) Spatio–Temporal Data?

- Trajectory data from
  - vehicles
  - mobile users
  - animals
  - ...

- Can be represented treating time as an additional spatial dimension
What is (certain) Spatio–Temporal Data?

- A spatio–temporal database stores triples (oid, time, loc)
- In the best case, this allows to look up the location of an object at any time
What is (certain) Spatio–Temporal Data?

- A spatio–temporal database stores triples (oid, time, loc)
- In the best case, this allows to look up the location of an object at any time
- Allows to answer queries such as "Return objects that intersects some spatial window within some time interval".
What is Uncertain Spatio–Temporal Data?

- In most applications, this data is not complete
  - Delays between GPS signals
  - RFID sensors located only in certain locations
  - Wireless sensor nodes sending infrequently to preserve power
  - Geo–application check–ins

![Diagram showing location space and time space with uncertain points marked with ? and Q highlighted]
What is Uncertain Spatio–Temporal Data?

- Existing works
  - Bound the set of possible (location, time) pairs of an object between observations by using spatio–temporal approximations (diamonds)
  - e.g. by modeling knowledge about maximum speed
What is Uncertain Spatio-Temporal Data?

- Existing works
  - Bound the set of possible (location,time) pairs of an object between observations by using spatio-temporal approximations (diamonds)
  - e.g. by modeling knowledge about maximum speed
  - Allows to make statements like „its possible that o intersects some query window Q“
  - But how likely is this event? „What is the probability of the object traveling through Q?“
Modeling Uncertain Spatio-Temporal Data

- Using a more powerful model [1]
- The position of an object o at some time t is a random variable
- The trajectory of o follows a stochastic process, i.e. a family of random variables o(t)
Modeling Uncertain Spatio-Temporal Data

- Using a more powerful model [1]
- The position of an object $o$ at some time $t$ is a random variable
- The trajectory of $o$ follows a stochastic process, i.e. a family of random variables $o(t)$
- Given a predicate $\varphi$, the event that $o$ satisfies $\varphi$ is a random event.
Markov Chain Model for UST data

- Assumes discrete state space $S$ and discrete time space $T$
- Given the position of an object $o$ at time $t=i$, the position at $t=i+1$ is conditionally independent of $t=i-1$
- Transition probabilities stored in a (sparse) $|S| \times |S|$ matrix $M(o,t)$, called transition matrix
- $M(o,t)[i,j]$ is the probability that object $o$ will transition to state $j$ at time $t+1$, given $o$ is located at state $i$ at time $t$
Markov Chain Model for UST data

- Incorporation of additional observations possible using Bayesian inference
- Each possible path can be associated with a probability
- Several probabilistic window queries possible
- Use sparse matrix operations for efficient implementation
- Details can be found in [1]
Indexing Uncertain Spatio–Temporal Data

- Large number of objects and observations in a database
- Checking each pair of successive observations requires too much computation
- How to prune as many objects and observations as possible during query evaluation?

“Which objects intersect $Q$ with a probability of at least $\tau$?”
Indexing Uncertain Spatio–Temporal Data

- Large number of objects and observations in a database
- Checking each pair of successive observations requires too much computation
- How to prune as many objects and observations as possible during query evaluation?
- Temporal Pruning?

“Which objects intersect Q with a probability of at least $\tau$?”
Indexing Uncertain Spatio–Temporal Data

- Index possible positions in (location + time) in an R*-Tree
- Apply multistep filter
  - R-Tree Filter

“Which objects intersect Q with a probability of at least $\tau$?”
Indexing Uncertain Spatio–Temporal Data

- Index possible positions in \((\text{location} + \text{time})\) in an R*-Tree
- Apply multistep filter
  - R-Tree Filter
  - Diamond Filter

"Which objects intersect Q with a probability of at least \(\tau\)?"
Indexing Uncertain Spatio–Temporal Data

- Index possible positions in (location + time) in an R*-Tree
- Apply multistep filter
  - R–Tree Filter
  - Diamond Filter
  - True Hit Detection

“Which objects intersect Q with a probability of at least \( \tau \)?”
Indexing Uncertain Spatio–Temporal Data

- Index possible positions in (location + time) in an R*-Tree

- Apply multistep filter
  - R–Tree Filter
  - Diamond Filter
  - True Hit Detection
  - Probabilistic Pruning

“Which objects intersect Q with a probability of at least $\tau$?”
Probabilistic Pruning

- Lower bound the probability of the object \( o \) to move through the gray area only \( P_{LB}(o \text{ in gray}) \)

- Then \( P_{UB}(o \text{ intersects } Q) = 1 - P_{LB}(o \text{ in gray}) \)

- If \( P_{UB}(o \text{ intersects } Q) < \tau \) 
  \( \Rightarrow \) \( o \) can be pruned

- How obtain \( P_{LB}(o \text{ in gray}) \) without computing it?

“Which objects intersect \( Q \) with a probability of at least \( \tau \)?”
Probabilistic Pruning

- Offline computation:
  - Precompute probability for $o$ to stay in sub-diamonds
  - Simple variation of the window query for each sub-diamond
Probabilistic Pruning

- **Offline computation:**
  - Precompute probability for \( o \) to stay in sub-diamonds
  - Simple variation of the window query for each sub-diamond
Probabilistic Pruning

- Offline computation:
  - Precompute probability for \( o \) to stay in sub-diamonds
  - Simple variation of the window query for each sub-diamond
Probabilistic Pruning

- Offline computation:
  - Precompute probability for \( o \) to stay in sub-diamonds
  - Simple variation of the window query for each sub-diamond
Probabilistic Pruning

- Offline computation:
  - Precompute probability for \( o \) to stay in sub-diamonds
  - Simple variation of the window query for each sub-diamond
Probabilistic Pruning

- Offline computation:
  - Precompute probability for o to stay in sub-diamonds
  - Simple variation of the window query for each sub-diamond

![Diagram showing location space and time space with points and a shaded area]
Probabilistic Pruning

- Offline computation:
  - Storing all these probabilities is of course memory inefficient
  - Approximation with a linear function using linear programming
Probabilistic Pruning

- During Query Processing:
  - Construct largest possible sub-diamond
  - Lookup of $P_{LB}(o \text{ in sub-diamond})$
Experiments

- Indexing UST data yields orders of magnitude over a scan based method (with temporal pruning)

- Using diamond and probabilistic filtering (UST-Tree) speeds up the R*-Tree by factor 3–4

- Setting: 1000 objects á 100 observations and 10000 possible locations in 2D
Experiments

- All Filters can be computeted efficiently in contrast to the verification step (~500 ms in this example)

- Effectiveness of probabilistic filter is dependent on the threshold $\tau$
Experiments

- Probabilistic spatio-temporal window queries are usually CPU-bound

- I/O-cost of UST-Tree are higher in the filter step since more information is stored in the leaves (=> higher tree)

- I/O-cost of UST-Tree are lower in the refinement step since less candidates have to be loaded from disk
Summary

- Indexing based on the Markov–Chain Model for UST data
  - Yields several orders of magnitudes over scan–based method
  - Yields 3–4 times speedup over straightforward indexing

- Techniques for spatio–temporal and probabilistic pruning are applicable to other models for UST data

- Techniques are extendable to the multi–dimensional case

- More special cases are treated in the paper
  - Other window queries
  - Query window overlapping several diamonds of the same object
Thank you for listening!
Related Work

Index Entries at Leaf Level:

directory levels:

leaf level:

oid