Predictive Spatio-Temporal Queries: A Comprehensive Survey and Future Directions*

Abdeltawab M. Hendawi Mohamed F. Mokbel
Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455, USA
{hendawi, mokbel}@cs.umn.edu

ABSTRACT

Predictive queries over spatio-temporal data proved to be vital in many location-based services including traffic management, ride sharing, and advertising. In the last few years, one of the most exciting work on spatio-temporal data management is about predictive queries. In this paper, we review the current research trends and present their related applications in the field of predictive spatio-temporal queries processing. Then, we discuss some basic challenges arising from new opportunities and open problems. The goal of this paper is to catch the interesting areas and future work under the umbrella of predictive queries over spatio-temporal data.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Spatial databases and GIS

General Terms

Algorithms, Performance

Keywords

Predictive Spatio-temporal Queries, Location-Based Services, Location Prediction, Trajectories, Moving Objects, Authentication, Privacy, Monitoring and Tracking, Query Optimization

1. INTRODUCTION

With the emerging and popularity of GPS enabled mobile devices and wireless communications, processing and managing spatio-temporal data becomes vital for many location based services and applications [21, 41, 42]. A typical aim of these applications is to answer users' queries such as range queries [12, 18, 63], e.g., "find all restaurants within two miles of my current location", K-nearest-neighbor (KNN) queries, e.g., "find the nearest pharmacy within two miles of my current location", and aggregate queries, e.g., "count the number of cars within one mile of a university campus". As a consequence of the wide spread of these location-aware devices and the importance of location-based applications, a considerable number of research was introduced to handle different problems of spatio-temporal data management, such as query processing and optimization [12, 18, 53, 63], indexing and accessing techniques [2, 4, 31, 52], and privacy preserving for users' exact locations [1, 25, 28].

However, the previous examples give more attention to queries related to the *current* locations of moving objects. Another important set of location-based services focuses on *predictive* spatio-temporal queries [19, 23, 24, 27, 36] in which a user can ask the same previous queries but for *future* time instance rather than *current* time instance. Such *predictive* queries are too substantial in many different applications that include location-based advertising, e.g., "find customers who are predicted to be within five miles of my store in the next 30 minutes", traffic management, e.g., "find areas with predicted traffic jam before it takes place", and car pooling and taxi services, e.g., "find drivers who are mostly expected to pass by my location after 10 minutes".

In this paper, triggered by its importance, we study the challenges and the opportunities in different research areas of the field of predictive spatio-temporal query processing and management. Basically, the meaning of predictive spatio-temporal query is explained in Figure 1. Without loosing the generality, this figure provides an example for *predictive range* query which is a main query type in predictive spatio-temporal queries. O_1*, to* O_4 are the objects in the Euclidean space at the current time t_1. The objects move and change their locations over time in their trips starting from their sources at time t_1 until reaching their destinations at time t_5. A user issues query at current time t_1 asking about the objects predicted to be inside a query rectangular region R_1 after four time units in the future, at t_5. As a query result, both O_2 *and* O_4 are predicted not to show up in the query region, while O_1 *and* O_3 are the objects that probably be inside the query region at t_5. The final answer returned to the user would indicate that O_1 *and* O_3 are the two objects expected to be in R_1 at t_5.

Our contribution in this paper is as follows. First, we survey the current research trends and present their related applications in the field of predictive spatio-temporal queries processing. This includes, (1) query evaluation and optimization, in which the main concern is to find the optimum or at least a good enough strategy for executing the received queries, (2) prediction functions, which refer to the underlying prediction model employed to anticipate the next/final destination or the complete forthcoming trajectory of a given moving object, (3) spatio-temporal indexing techniques, which attempt to find an efficient way to store and retrieve moving objects data, and (4) uncertainty, which deals with imprecise data

*The work of this paper is supported in part by the National Science Foundation under Grants IIS-0811998, IIS-0811935, CNS-0708604, IIS-0952977, and the Egyptian Ministry of Higher Education under Grant GM-887.

ACM SIGSPATIAL MobiGIS'12 November 6, 2012. Redondo Beach, CA, USA

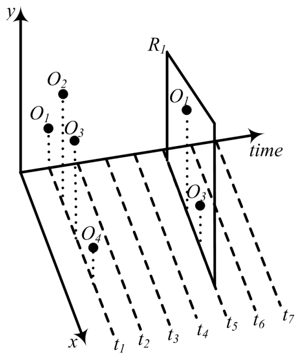


Figure 1: Example of Predictive Range Query Example

about objects locations, velocities, and directions while evaluating the prediction. Second, we highlight the fundamental challenges arising from the new opportunities and the open problems. We illustrate which areas have been well investigated and which still need more digging.

The remainder of the paper is organized as follows. We survey existing work in different areas related to predictive spatio-temporal queries in Section 2. Then, we discuss the challenges in six key topics, namely *query processing* in Section 3, *prediction functions* in Section 4, *uncertainty* in Section 5, *monitoring and tracking* in Section 6, *privacy* in Section 7, and *authentication* in Section 8. Finally, we conclude in Section 9

2. SURVEY OF EXISTING WORK

In this section, we give a survey for the existing work related to different branches of predictive spatio-temporal queries. We start by reviewing the existing work about query processing and optimization, then summarizing the commonly used prediction functions, indexing data structures, and finally discussing the used techniques to handle uncertainty while evaluating predictive queries.

2.1 Query Processing and Optimization

Existing algorithms for predictive query processing and optimization can be classified according to the supported query type into the following categories:

(1) *range queries*, e.g., [24, 56, 70]. A predictive range query has a query region R and a future time t, and asks about the objects expected to be inside R after time t. A mobility model [24] is used to predict the coming path of each of the underlying objects and employ the prediction results to evaluate predictive range queries. Most of existing work considers query region as a rectangle, however the Transformed Minkowski Sum supports range queries with circular regions [70]. This is done by determining whether a time parameterized bounding rectangular, as a moving object, intersects a moving circle that represents a range query. The initial rectangle of the object and the velocity of each edge in this rectangle are considered to compute the position and the rectangle after a certain duration of time in the future.

(2) K*-nearest-neighbor queries*, e.g., [5, 50, 70]. A predictive K-nearest-neighbor query has a location point P, a future time t, and asks about the K objects expected to be closest to P after time t. Two algorithms, RangeSearch and KNNSearchBF, [70] are introduced to traverse spatio-temporal index tree (TPR/TPR*-tree) to find the nodes that intersect with the query circular region for Range and KNN queries, respectively. Sometimes an expiry time interval is attached to a KNN query result [57, 58]. Thus, the KNN query answer is presented in the form of <*result, interval*>, where the interval indicates the future interval during which the reported answer is valid.

(3) *reverse-nearest-neighbor queries*, e.g., [5, 64]. Unlike the predictive KNN query which finds the objects expected to be the nearest to a given query region, predictive reverse nearest neighbor (RNN) query finds out the objects expected to have the query region as their nearest neighbor. This query is useful in service distribution applications such as ad-hoc networking to assign mobile devices to the nearest communication service point. For example, an algorithm [5] is proposed to evaluate RNN queries during some specific future time duration starting at the query time. This work assumes the objects movements to be in a linear behavior.

(4) *aggregate queries*, i.e., [55]. A predictive aggregate query has a query region R and a future time t, and asks about the number of objects $\mathcal{N}$ predicted to be inside R after time t. A comprehensive technique [55] that employs adaptive multi-dimensional histogram (AMH), historical synopsis, and stochastic method is used to provide an approximate answer for aggregate spatio-temporal queries for the future, in addition to the past, and the present.

(5) *continuous queries*, e.g., [26, 36, 41, 63]. The difference between a *snapshot* predictive query and a *continuous* one is that the later needs to be continuously reevaluated many times through out its life in the system. The rate of reevaluation depends on the time gap t_{gap} between each two consecutive reported answers specified in the received query *Q*. Accordingly, continuous query *Q* needs to be stored at the server side until the end of its life. For example, a quadratic-based kNN [36] algorithm is introduced for processing predictive continuous KNN queries, and a differential update technique is used to maintain the query answers. Next, a probabilistic evaluation [12, 18, 63] is considered for processing continuous range queries.

Additional work considers *query selectivity* which plays a complementary role in the area of predictive spatio-temporal query processing and optimization, i.e., [11, 13, 61, 60]. Selectivity prediction is defined as the number of objects expected to be retrieved divided by the size of the underlying objects data set. Accurate estimation for the predicted query selectivity is essential in query optimization and evaluation. For example, spatio-temporal histograms [11, 61] are used to predict spatio-temporal query selectivity.

To sum up, although the extensive investigation for the majority of different types of predictive queries, but each of the existing work supports one or two query types and ignore the rest. Consequently, there is still a lack for one general framework that can support all or at least most of the mentioned query types.

2.2 Prediction Functions

In terms of the underlying prediction function, existing algorithms for predictive spatio-temporal query processing can be classified into three categories:

(1) *Linearity-based prediction*, where the underlying prediction function is based on a simple assumption that objects move in a linear function in time along the input velocity and direction. So, query processing techniques in this category, e.g., [5, 50, 54, 58, 60], take into consideration the position of a moving point at a certain time reference, its direction, and the velocity to compute and store the future positions of that object in a TPR-tree-based index [52]. When a predictive query is received, the query processor retrieves the anticipated position in the given time [54]. Part of the related work in this category concerns with the applications of linearity-based prediction models to answer nearest neighbor

queries [50] and reverse nearest neighbor queries [5], or to estimate the query selectivity [60].

(2) *Historical-based prediction*, where the predication function uses object historical trajectories to predict the object next trajectory. Then, query processing techniques in this category, e.g., [7, 15, 24, 27, 33, 30, 55] are applied to trajectory of location points. Existing work in this category is based on either mobility model [24] or ordered historical routes [7, 15, 30]. The mobility model [24] is used to capture different possible turning patterns at different roads junctions, and the travel speed for each segment in the road network for each single object in the system. Then, the model is used to predict the future trajectory of each object, and based on that they can answer predictive range queries. The main concern of that model is to put more focus on the prediction of the object behavior in junctions based on historical data of objects trajectories. In the ordered historical routes, the stored past trajectories are ordered according to the similarity with the current time and location of the object and the top route is considered the most possible one [7, 15, 30, 32, 33]. Some of the existing work in this category is employed for predicting the current object trajectory in non-euclidian space [27] such as road-level granularity. For example, a Predictive Location Model (PLM) [27] is proposed to predict locations in location-based services. This model considers the start point as the object current location while the end point could be any of the possible exit points. PLM computes the shortest path trajectory between the current location and each of the exit points, then the trajectory with the highest probability is considered the predicted path. Moreover, a probabilistic prediction function based on Markov models [33] is introduced for short-term route prediction, while Bayes rule is adapted to predict the final destination of a moving object [15, 32, 35].

(3) *Other prediction functions*, where more complicated prediction functions are employed to realize better prediction accuracy. Query processing techniques in this category, e.g., [23, 56, 69, 70] are adjusted based on the outcome of the prediction function. Existing work in this category either exploits a single function [56, 70], or mixes between two or more functions to form a hybrid prediction model [23, 69]. As an example for a single function, a Transformed Minkowski Sum [70] is used to answer predictive queries with circular regions, while Recursive Motion Function (RMF) [56] is used to predict a curve that best fits the recent locations of a moving object and accordingly answer range queries. In the hybrid functions category, two methods [23, 69] are combined to evaluate predictive range and nearest neighbor queries in highly dynamic and uncertain environments.

Unfortunately, all the employed prediction functions either: (a) support only short-term prediction in terms of seconds, minutes, or next edge prediction, (b) support long-term prediction but assume a linearity movement of the underlying moving objects which is not a realistic assumption, or (c) based on complex techniques with a significant computation cost that can not scale up for large number of objects. As a result, there is still a need for prediction models that can support long-term prediction as well as short-term prediction with the ability to scale up to huge query workloads and large number of moving objects.

2.3 Indexing Techniques

A wide variety of data structures are proposed to index spatio-temporal data to support predictive query evaluation and processing. Some of the existing work assumes simple movement pattern for the underlying moving objects while others handles more complex objects behavior. The most popular spatio-temporal indices can be categorized with respect to the base data structure as follows.

(1) *R-tree based*, e.g., [4, 51, 52, 53, 59]. Time Parameterized R-tree (TPR-tree) [52] is an extension of R-tree by adding the time parameter which can be used to support querying current and projected future positions of moving objects. The TPR-tree was enhanced to the TPR*-tree [59] by introducing some improved construction algorithms. R^{exp}-tree [51] is proposed as an access method that indexes the current and predicted locations of moving objects assuming that their positions expire after specified time periods. A different technique based on convex hull property [3] is introduced for indexing objects with nonlinear trajectories using a traditional index structure.

(2) *B-tree based*, e.g., [9, 22, 67]. An indexing schema called B^x-tree [22] based on B$^+$-tree, uses a linear technique to index changes in the underlying data values such as moving-objects locations. Based on this B^x-tree index, some algorithms were provided to answer predictive range and KNN queries on near-future positions of the indexed objects. B^{dual}-tree is an enhancement of B-tree by taking into consideration the velocity in addition to the location while indexing the moving objects [67]. Next, a Self-Tunable Spatio-Temporal B$^+$-tree, termed ST^{2}B-tree, is introduced [9] to handle frequent updates for objects locations. This is done by allowing automatic online rebuilding of its subtrees using a different set of reference points and different grid size without significant overhead. The key for an entry in the ST^{2}B-tree index consists of a time part in addition to a space part. The time part is based on dividing the dimension into equivalent partitions, while the space part is based on the Voronoi diagram for dividing the space.

(3) *kd-tree based*, e.g., [6, 14, 44, 62]. MOVIES is a main memory indexing technique [6, 14] proposed to handle the high frequent updates while guaranteeing fast response to predictive queries on moving objects data. It is based on the kd-tree data structure [44, 62] and its main idea is to create new indexes for the most updated pieces of the main index and throw them away from the main memory after some short time period.

(4) *Quad-tree based*, e.g., [48]. A dual transformation is used in an indexing method called STRIPES [48] to index the predicted trajectories in a dual transformed space. Trajectories for objects in a d-dimensional space become points in a higher-dimensional, (a 2d-dimensional), space. This dual transformed space is then indexed using a regular hierarchical grid decomposition indexing structure. More detailed review and comparison between the existing techniques for indexing the current and predictive locations of moving objects is covered in [8, 40, 44].

2.4 Uncertainty

Unlike most of existing work in predictive queries that assume the deterministic behavior of objects movements, few research trials assume uncertainty about these movements. Basically, uncertainty deals with stochastic, (probabilistic movement patterns), of the underlying objects with respect to object locations and velocities at different time stamps. The existing few trials in the area of predictive queries over uncertain moving objects data can be classified according to their aim as follows.

(1) *Indexing* e.g., [69]. To index uncertain motions of a set of moving objects, the B^x-tree is enhanced and two movement inferencing techniques are introduced to obtain anticipated objects locations in non-deterministic format. This work assumes uncertainty for the given past locations and velocities. The adapted B^x-tree and the inference techniques are employed to evaluate predictive range and KNN queries.

(2) *Modeling* e.g., [49, 56]. Unlike the work that is based on a deterministic linear movement, a Recursive Motion Function (RMF) [56]

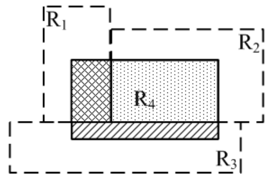


Figure 2: Example of Space Sharing in Overlapped Regions

Query Table

ID	Region	Answer
Q_1	R_1	$\{O_5, O_9, O_{18}\}$
...	...	...

(a) Range Query

Query Table

ID	Region	Answer
Q_2	R_1	3
...	...	...

(b) Aggregate Query

Figure 3: Example of Data Structure Sharing to Serve Different Query Types

is presented to model uncertain motion patterns in different shapes e.g. polynomial, sinusoid, circle, and ellipse. This work assumes uncertainty about the representation of objects movements patterns. Moreover, a Spatio-Temporal Prediction tree, STP-tree [56], is introduced to index these uncertain movement patterns and to answer predictive queries. Next, a model called PutMode [49] is introduced to predict next trajectory using uncertain data about objects locations. However, none of the predictive spatio-temporal queries are explicitly supported.

3. CHALLENGE 1: QUERY PROCESSING

In this section, we study the challenges and opportunities that are still open problems and need to have an insight look in the area of query processing and optimization. Basically, existing solutions suffers from the following limitations.

(1) Each one of the existing work is applicable for only one predictive query type or two at most. So, there is a lack for a generic predictive query processor that can support to a wide variety of query types i.e., range, KNN, aggregate, reverse-KNN. Generality of predictive query processor is not only in terms of query type but also it expands to include the shape of query region. Thus, a generic query processer should be able to answer queries in different shapes such as rectangular, circular, or any other irregular polygons. Another generality dimension is query continuity. Whether a predictive query is snapshot or continuous, or stationary or moving, a complete query processor should be able to support all of these essential variations.

(2) Existing solution do not introduce a sufficient level of scalability. Most of existing techniques were tested against low workloads i.e., in terms of thousands of objects and/or queries, while real needs require processing technique to scale to millions of objects and queries within very short time unit. Yet, realistic scalability with low computation cost and fast response time is still an important challenging problem.

(3) To the best of our knowledge, none of the existing solutions can be supported by the existing infrastructure of database management systems. The reason for that is because existing query processing techniques rely on special indexing structures i.e., STRIPES or TPR-tree or assumes a certain framed processing environment i.e., linear movements of all moving objects. Yet, in addition to the generality and scalability features illustrated in the previous lines, it is a vital research point to introduce predictive query processor that is feasible to integrate with conventional data management systems without major modification in their infrastructure.

One of the common concepts employed to allow generality of query processor is *sharing* which can be used to maximize the benefit from the overlapping among received queries. Using this concept can significantly reduce the total computation cost. Examples of sharing include.

(a) Sharing parts of the queries interest, e.g., query Q_1 asks "find out all vehicles expected to be in region R_1 after 30 minutes" and query Q_2 asks "find out taxi vehicles expected to be in region R_1 after 30 minutes". Clearly, there is a relationship between the interest of Q_1 and the one for Q_2 that can be expressed as $Q_2 \subset Q_1$. So, we can make use of Q_1 answer to get the answer for Q_2 without the need to reprocess Q_2 from scratch.

(b) Sharing parts of the space between query regions. For example, assume we have region R_1 of query Q_1 and region R_2 of another query Q_2 where $R_1 \cap R_2 \neq \phi$. By applying the sharing concept, we can use the overlapped space from one query to prepare the answer for other queries. The example given in Figure 2 provides that the region R_4 is fully covered by the other three regions. Consequently, the predicted answer in R_4 can be returned without further computation as it simply can be composed from the existing answers of the overlapped parts from other queries, given they have a common interest in other variables such as future time period.

(c) Sharing the same piece of data structure. For example an *answer* field can by used to answer range query by carrying a list objects inside a given region and also can be customized to answer an aggregate query by counting the number of objects expected to show up in that region. Figure 3 illustrates sharing the same data structure element to serve different type of queries. This concept of sharing is employed by the *Panda* system [19] to support a wide variety of predictive spatio-temporal queries including predictive range, KNN, and aggregate queries. However, there is still a question about its ability to scale up to large network graph with millions of nodes and edges rather than a space partitioned into hundreds or thousands of grid cells.

4. CHALLENGE 2: PREDICTION FUNCTION

Except few attempts, most of the existing work assumes a linear movement pattern for moving objects. Initially, that assumption is not realistic, since actual objects have more complex movement patterns as they rarely move in straight lines. Although some of the used prediction functions to answer predictive queries can support long-term prediction, but they share major drawbacks such as poor accuracy, scalability limitation, and bad response time. In fact, the majority of existing prediction functions can fit only in short-term prediction. The term short-term prediction can be defined with respect to time in terms of few seconds in the future and with respect to space in terms of next turn, junction or destination, while the term long-term prediction can be defined in terms of tens of minutes in the future time, next complete trajectory, or anticipated final destination.

Figure 4 gives an example for a long-term prediction function used to predict a final destination [15, 32] of a given moving object. The way this prediction function works is demonstrated as follows. Initially, the given space in which objects move is partitioned into 6×6 squared cells numbered from 1 to 36. The current trajectory of object O_1 is drawn as a line started at cell C_{15} and headed to cell C_{18}. The sequence of cells representing O_1 in its current trip is $S_{O_1} = \{C_{15}, C_{16}, C_{22}, C_{23}, C_{18}\}$. The color of a cell indicates

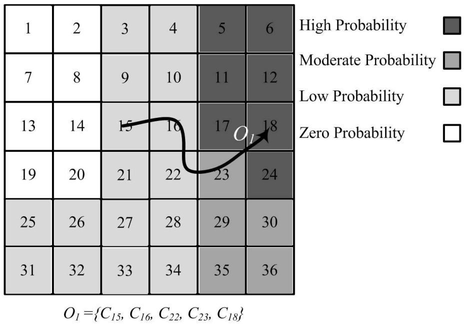
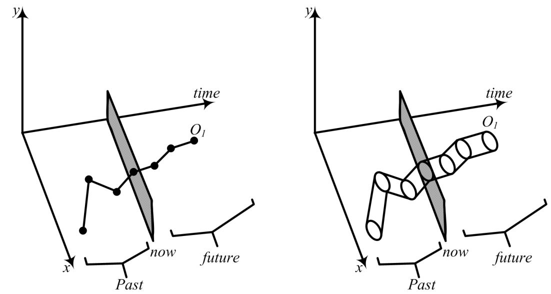


Figure 4: Final Destination Prediction



(a) Exact Trajectory (b) Imprecise Trajectory

Figure 5: Impact of Uncertainty on Prediction

its probability of being a destination to the object O_1 given its sequence S_{O_1}. The darker the cell color, the higher the probability of this cell to be a destination to O_1. As the object moves toward its final trip destination, the prediction function updates its computation. So, some of the grid cells become more likely destination (e.g., C_{24}), and others become less, (e.g., C_{31}). However, this function suffers from two main shortcomings. First, it can only predict a final destination for a moving object and it fails to predict its next complete pass. Second, it does not have the sense of time, which means it can not predict the location of object O_1 after a specified time period t in the future.

Not only predicting forthcoming location/trajectory of a moving object is challenging, but also predicting its anticipated size and shape is challenging, specially in highly dynamic environments. For example if a tornado, as a moving object, has a rectangular shape in the current time and its current location is known, it is not an easy task to predict the answer for a question like "find out commuters that might be hit by this tornado in the next 30 minutes". This example is one of the most difficult predictive tasks, as commuters change their locations and velocities and also the tornado modifies its shape, size, speed, and direction. Taking all of these factors into consideration makes prediction more complex, and challenging.

Challenges in the area of prediction functions can be summarized in the following statement. There is still an essential need to find a well-designed prediction function that is able to precisely support long-term as well as short-term prediction, has the sense of time, able to capture changes in objects recent behavior, and efficiently scale up to large numbers of moving objects.

5. CHALLENGE 3: UNCERTAINTY

In this section, we discuss the challenging points that should be addressed while designing a model for predicting next/final destination, or complete trajectory of moving objects given imprecise spatio-temporal data.

The importance of dealing with uncertainty while processing predictive queries comes from the fact that spatio-temporal data is usually imprecise. Data uncertainty results from many sources such as the erroneous in GPS readings, accuracy limitation in measuring devices, infrequent readings due to battery shortage, communication delays and computation limitations, inexact velocity estimations, environmental obstacles such as buildings and severe weather that obstruct the communication between reading devices and satellites, and the stochastic objects motion behavior. As a result, there is a vital need to take into consideration data uncertainty while supporting predictive query processing. Basically, not considering the imprecise nature of data leads to inaccurate prediction for object future location.

Figure 5 illustrates the difference between defining an object trajectory using exact location points (Figure 5)a versus the one defined using imprecise locations (Figure 5b). In Figure (Figure 5b), object locations at different time stamps are expressed in circular regions rather than exact points. Accordingly, giving these imprecise locations to a predictive query processor as an input produces imprecise prediction for its next path.

Little work has been done in this area. For example, in a recent work [65], a new concept based on u-bisector is used to evaluate two types of KNN queries, namely, Possible Nearest Neighbor Query (PNNQ) and Trajectory Possible Nearest Neighbor Query (TPNNQ) given imprecise object location represented as circular regions rather than exact points. However, this work does not support evaluation of predictive queries. Even for those few trials that attempt to support predictive queries, they still have accuracy, generality, and scalability issues. Consequently, it is challenging to process different kinds of predictive queries given imprecise and uncertain data about objects locations, velocities, and nondeterministic motion behaviors. Expected solutions should consider being online, which means consuming less computation time as objects dynamically change their locations and velocities.

6. CHALLENGE 4: MONITORING

Tracking and monitoring moving objects is a fundamental challenge in the area of predictive spatio-temporal queries. The objective of this challenge is to capture moving objects updates that result from changing their locations, directions, and/or velocities, and efficiently handle the effect of those updates. To the best of the authors' knowledge, existing literatures do not address the moving objects monitoring and tracking problem while considering predictive queries. They only consider it for queries in the current time such as KNN and range queries [38, 43, 68]. The challenging in this point arises from the fact that many factors should be optimized. These factors not only include correctly capturing objects updates, but also reducing the number of updates to capture and send to the server, hence, reducing the communication cost between the server and the moving objects, which in turns safes the computation time required to handle the received updates. To achieve this objective, expected solution can benefit from the techniques used in the previous work which include, (a) specifying some regions in the given space in which objects movements are known to mostly affect queries results, and outside those regions updates are neglected, and (b) predicting the next trajectory of an object and report only updates if its real movement is different from the predicted one.

7. CHALLENGE 5: PRIVACY PRESERVING

Although preserving users privacy while evaluating their queries is a fundamental issue for the database filed in general and for the spatio-temporal data processing in specific, but none of the existing work, to the best of authors's knowledge, discusses the privacy issue in predictive spatio-temporal query processing. Ignoring privacy means we assume that users are willing to reveal their exact trajectories data while asking for some location-based services which is not always true assumption in many applications, specially with data management outsourcing.

Concealing moving objects' trajectories while evaluating predictive query is challenging because of the fact that there is no accurate prediction output if there is no precise history as input. So at one side, query processor needs to have an image of objects movements to be able to answer predictive queries. However, on the other side, users do not want to uncover their privacy by revealing all of their movements. In order to resolve this issue, many common techniques for dealing with privacy while answering location-based queries in the current time can be adapted for protecting privacy while responding to location-based queries for the future. Examples for these techniques include *anonymization* [1, 29] which aims to make an object's location unrecognized among K other objects, *cloaking* [17, 25, 39] which aims at expressing locations into bounding rectangles rather than exact points, *perturbation* which replaces or stuffs the real location values with synesthetic values such as noise addition [34], *cryptography* [16] which turns location readings into unreadable format to anyone except those who have the decryption key, and *transformation* [28] which protects users locations by converting the underlying space into another space.

Each of the aforementioned techniques has some merits and drawbacks. For example *cloaking* can achieve sufficient privacy while providing an accurate answer for predictive queries about tornado movements in a given city or zip code rather than an exact point. However, for predictive nearest neighbor queries, *cloaking* can not provide such accurate answer. A comprehensive study is required to assert which technique can be adjusted to support predictive spatio-temporal queries while preserving privacy of the underlying moving objects.

8. CHALLENGE 6: AUTHENTICATION

Authentication is a consequence issue to the age of cloud computing and data management outsourcing. The emersion of the clouds with its offered services in affordable cost, encourages the outsourcing of data management from the data owner $\mathcal{DO}$ side to be located at the data management service provider $\mathcal{SP}$ side. With this phenomenon, user's query is received, processed and the answer is returned by the service provider $\mathcal{SP}$. With the assumption that service providers are not always trust worthy , the returned answers from their side are not trusted to be accurate. For Example, The $\mathcal{SP}$ might be hacked such that a certain instance is added to all returned results, or the $\mathcal{SP}$ itself might be not trust worthy, so it might change the results by adding, removing or modifying some parts of the returned answer.

This schema triggers the need for techniques to check the accuracy of the returned answer at the user side. Consequently, some techniques for authenticated query processing [10, 66] in which the query answer returned by the untrusted $\mathcal{SP}$ can be tested against the completeness and soundness are introduced. Completeness of answer means that no correct piece of data that should be included in the final results is disappeared from the returned answer to users. Soundness means that no modification, neither by adding non existence record nor modifying an existing one, takes place on the result [47].

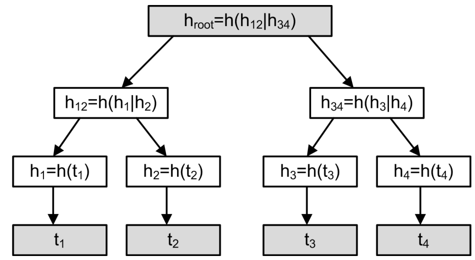


Figure 6: Example of Authenticated Index (MH-tree)

Although authentication is an essential issue in the paradigm of data management outsourcing, but we did not come across any related work that addresses this issue on moving objects data neither for current time queries nor predictive queries. Even for recent work [20], it handles the authentication problem with preserving users' privacy for queries on static objects at current time instance. To the best of our knowledge, until now, all existing work in this area deals only with authentication for current time queries over stationary objects.

The big portion of those existing techniques for authenticated query processing is based on, (a) an authenticated data structure (ADS) such as MB-tree and MH-tree [37], and MR-tree [47], which stores the outsourced data along with hash values computed for each tuple *t* and signed by the $\mathcal{DO}$, Figure 6 , and (b) the concept of verification object $\mathcal{VO}$ for carrying out the answer records with additional digest (hash) values. At the client side, this verification object is used to check the soundness and the completeness of the answer where the summation of the received hash values should be equal to the hash value of the root.

The second main technique used in the authentication existing work is signature aggregation [45, 46], in which each record in the database has a signature from the data owner $\mathcal{DO}$. The difference between the authenticated data structure technique and the signature aggregation technique is that the later guarantees higher concurrency processing for authenticated transactions, and it needs smaller number of verification objects which reduces the communication cost for sending those objects to the users. However, signature aggregation suffers from significant update overhead at the $\mathcal{DO}$ side and also costs more for verifying the received answer at the client side.

Addressing the problem of authenticated processing for predictive spatio-temporal queries is a challenging. The core difficulty of this challenge comes from the dynamic nature of moving objects updates about their locations, velocities, and directions. With each update, the data owner $\mathcal{DO}$ needs to refresh her signature on the outsourced data and resend the latest copy to the service provider $\mathcal{SP}$. Obviously, this will lead to a significant communication and computation overhead that overwhelms any gains from data outsourcing service.

9. CONCLUSION

In this paper, we address the problem of predictive spatio-temporal query processing from different dimensions, namely *query optimization*, *prediction functions*, *indexing and access methods*, and *uncertainty*. We argue that it is the time for data base community to think in providing a complete, generic, and scalable query processor over spatio-temporal data, and being feasible to integrate with conventional DBMSs. We discussed some of the key challenges researches should face while thinking to build a solution for this vital problem. To handle these challenges, the target solution should answer questions such as: (a) how to make the query processor in the target solution able to support a wide variety of predictive queries including KNN, range, and aggregate queries, (b) how to provide a prediction function that acts accurately for long-term prediction as well as short-term prediction, (c) how to deal the imprecise data about objects locations, directions, and velocities, (d) how to monitor the underlying moving objects such that relevant updates are only captured, (e) how to provide answers for users' queries while preserving their location privacy, and finally (f) how to give users a tool to validate the correctness, and the completeness of the received answers in case that the data management is outsourced.

10. REFERENCES

[1] O. Abul, F. Bonchi, and M. Nanni. Never walk alone: Uncertainty for anonymity in moving objects databases. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 376–385, Cancun, Mexico, Apr. 2008.

[2] P. K. Agarwal, L. Arge, and J. Erickson. Indexing Moving Points. In *Proceedings of the ACM Symposium on Principles of Database Systems, PODS*, pages 175–186, Texas, USA, May 2000.

[3] C. C. Aggarwal and D. Agrawal. On Nearest Neighbor Indexing of Nonlinear Trajectories. In *Proceedings of the ACM Symposium on Principles of Database Systems, PODS*, pages 252–259, California, USA, June 2003.

[4] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger. The R^*-tree: An Efficient and Robust Access Method for Points and Rectangles. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 322–331, New Jersey, USA, May 1990.

[5] R. Benetis, C. S. Jensen, G. Karciauskas, and S. Saltenis. Nearest and Reverse Nearest Neighbor Queries for Moving Objects. *VLDB Journal*, 15(3):229–249, 2006.

[6] J. D. L. Blunschi and M. A. V. Salles. Indexing moving objects using short-lived throwaway indexes. *Proceedings of the International Symposium on Advances in Spatial and Temporal Databases, SSTD*, pages 189–207, 2009.

[7] A. Brillingaite and C. S. Jensen. Online Route Prediction for Automotive Applications. In *Proceedings of teh World Congress and Exhibition on Intelligent Transport Systems and Services, ITS*, London, United Kingdom, Oct. 2006.

[8] S. Chen, C. S. Jensen, and D. Lin. A Benchmark for Evaluating Moving Object Indexes. *VLDB Journal*, 1(2):1574–1585, Aug. 2008.

[9] S. Chen, B. C. Ooi, K. L. Tan, and M. A. Nascimento. ST^2B-tree: A Self-tunable Spatio-temporal B^+-tree Index for Moving Objects. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 29–42, Vancouver, Canada, June 2008.

[10] W. Cheng and K.-L. Tan. Query assurance verification for outsourced multi-dimensional databases. *Journal of Computer Security*, 17(1):101–126, 2009.

[11] Y.-J. Choi and C.-W. Chung. Selectivity Estimation for Spatio-temporal Queries to Moving Objects. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 440–451, Wisconsin, USA, June 2002.

[12] H. D. Chon, D. Agrawal, and A. El-Abbadi. Range and kNN Query Processing for Moving Objects in Grid Model. *Mobile Networks and Applications, MONET*, 8(4):401–412, 2003.

[13] A. Corral, M. Torres, M. Vassilakopoulos, and Y. Manolopoulos. Predictive Join Processing between Regions and Moving Objects. In *Advances in Databases and Information Systems Lecture Notes in Computer Science, ADBIS*, volume 5207, pages 46–61, 2008.

[14] J. Dittrich, L. Blunschi, and M. A. V. Salles. MOVIES: indexing moving objects by shooting index images. *International Journal on Advances of Computer Science for Geographic Information Systems, GeoInformatica*, 15(4):727–767, 2011.

[15] J. Froehlich and J. Krumm. Route Prediction from Trip Observations. In *Society of Automotive Engineers (SAE) World Congress*, Michigan, USA, Apr. 2008.

[16] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan. Private queries in location based services: anonymizers are not necessary. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 121–132, Vancouver, Canada, June 2008.

[17] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of the international conference on Mobile systems, applications and services, MobiSys*, pages 31–42, California, USA, May 2003.

[18] Y. Gu, G. Yu, N. Guo, and Y. Chen. Probabilistic Moving Range Query over RFID Spatio-temporal Data Streams. In *Proceedings of the International Conference on Information and Knowledge Managemen, CIKM*, pages 1413–1416, Hong Kong, China, Nov. 2009.

[19] A. M. Hendawi and M. F. Mokbel. Panda: A Predictive Spatio-Temporal Query Processor. In *Proceedings of the ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM SIGSPATIAL GIS*, California, USA, Nov. 2012.

[20] H. Hu, J. Xu, Q. Chen, and Z. Yang. Authenticating location-based services without compromising location privacy. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 301–312, Arizona, USA, May 2012.

[21] H. Hu, J. Xu, and D. L. Lee. A Generic Framework for Monitoring Continuous Spatial Queries over Moving Objects. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 479–490, Maryland, USA, June 2005.

[22] C. S. Jensen, D. Lin, and B. C. Ooi. Query and Update Efficient B^+-tree Based Indexing of Moving Objects. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 768–779, Toronto, Canada, Aug. 2004.

[23] H. Jeung, Q. Liu, H. T. Shen, and X. Zhou. A Hybrid Prediction Model for Moving Objects. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 70–79, Cancún, México, Apr. 2008.

[24] H. Jeung, M. L. Yiu, X. Zhou, and C. S. Jensen. Path Prediction and Predictive Range Querying in Road Network Databases. *VLDB Journal*, 19(4):585–602, Aug. 2010.

[25] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias. Preventing location-based identity inference in anonymous spatial queries. *IEEE Transactions on Knowledge and Data Engineering, TKDE*, 19(12):1719–1733, 2007.

[26] J. Kang, M. F. Mokbel, S. Shekhar, T. Xia, and D. Zhang. Continuous Evaluation of Monochromatic and Bichromatic Reverse Nearest Neighbors. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 806–815, Istanbul, Turkey, Apr. 2007.

[27] H. A. Karimi and X. Liu. A Predictive Location Model for Location-Based Services. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM GIS*, pages 126–133, Louisiana, USA, Nov. 2003.

[28] A. Khoshgozaran and C. Shahabi. Private Information Retrieval Techniques for Enabling Location Privacy in Location-Based Services. In *Proceedings of the International Symposium on Advances in Spatial and Temporal Databases, SSTD*, pages 59–83, Aalborg, Denmark, July 2009.

[29] H. Kido, Y. Yanagisawa, and T. Satoh. Protection of location privacy using dummies for location-based services. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 1248–1248, Tokyo, Japan, Apr. 2005.

[30] S.-W. Kim, J.-I. Won, J.-D. Kim, M. Shin, J. Lee, and H. Kim. Path Prediction of Moving Objects on Road Networks Through Analyzing Past Trajectories. In *Proceedings of the International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, KES*, pages 379–389, Vietri sul Mare, Italy, Sept. 2007.

[31] G. Kollios, D. Gunopulos, and V. J. Tsotras. On Indexing Mobile Objects. In *Proceedings of the ACM Symposium on Principles of Database Systems, PODS*, pages 261–272, Philadelphia, Pennsylvania, May 1999.

[32] J. Krumm. Real Time Destination Prediction Based on Efficient Routes. In *Proceedings of the Society of Automotive Engineers World Congress, SAE*, Michigan, USA, Apr. 2006.

[33] J. Krumm. A markov model for driver turn prediction. *Proceedings*

of the Society of Automotive Engineers World Congress, SAE, 2193(1), 2008.

[34] J. Krumm. A survey of computational location privacy. *Personal and Ubiquitous Computing*, 13(6):391–399, 2009.

[35] J. Krumm and E. Horvitz. Predestination: Inferring Destinations from Partial Trajectories. In *Proceedings of the International Conference on Ubiquitous Computing, UbiComp*, pages 243–260, California, USA, Sept. 2006.

[36] K. C. K. Lee, H. V. Leong, J. Zhou, and A. Si. An Efficient Algorithm for Predictive Continuous Nearest Neighbor Query Processing and Result Maintenance. In *Proceedings of the International Conference on Mobile Data Management, MDM*, pages 178–182, Ayia Napa, Cyprus, May 2005.

[37] F. Li, M. Hadjieleftheriou, G. Kollios, and L. Reyzin. Dynamic authenticated index structures for outsourced databases. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 121–132, Illinois, USA, June 2006.

[38] T. Li, W. Le, Z. Peng, J. Yan, and L. Aiping. Continuous monitoring of skyline query over highly dynamic moving objects. In *Proceedings of the ACM international workshop on Data engineering for wireless and mobile access, MobiDE*, pages 59–66, Beijing, China, June 2007.

[39] M. F. Mokbel, C.-Y. Chow, and W. G. Aref. The new Casper: query processing for location services without compromising privacy. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 763–774, Seoul, Korea, Sept. 2006.

[40] M. F. Mokbel, T. M. Ghanem, and W. G. Aref. Spatio-temporal Access Methods. *IEEE Data Engineering Bulletin, DEBull*, 26(2):40–49, June 2003.

[41] M. F. Mokbel, X. Xiong, and W. G. Aref. SINA: Scalable Incremental Processing of Continuous Queries in Spatio-temporal Databases. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 443–454, Paris, France, June 2004.

[42] M. F. Mokbel, X. Xiong, M. A. Hammad, and W. G. Aref. Continuous Query Processing of Spatio-temporal Data Streams in PLACE. In *Proceedings of the workshop on Spatio-Temporal Database Management, STDBM*, pages 57–64, Toronto, Canada, Aug. 2004.

[43] K. Mouratidis, M. Hadjieleftheriou, and D. Papadias. Conceptual Partitioning: An Efficient Method for Continuous Nearest Neighbor Monitoring. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 634–645, Maryland, USA, June 2005.

[44] L.-V. Nguyen-Dinh, W. G. Aref, and M. F. Mokbel. Spatio-Temporal Access Methods: Part 2 (2003 - 2010). *IEEE Data Engineering Bulletin*, 33(2):46–55, 2010.

[45] H. Pang, A. Jain, K. Ramamritham, and K.-L. Tan. Verifying Completeness of Relational Query Results in Data Publishing. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 407–418, Maryland, USA, June 2005.

[46] H. Pang, J. Zhang, and K. Mouratidis. Scalable Verification for Outsourced Dynamic Databases. *Proceedings of the VLDB Endowment, PVLDB*, 2(1):802–813, 2009.

[47] S. Papadopoulos, Y. Yang, and D. Papadias. Continuous authentication on relational streams. *VLDB Journal*, 19(2):161–180, 2010.

[48] J. M. Patel, Y. Chen, and V. P. Chakka. STRIPES: An Efficient Index for Predicted Trajectories. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 637–646, Paris, France, June 2004.

[49] S. Qiao, C. Tang, H. Jin, T. Long, S. Dai, Y. Ku, and M. Chau. PutMode: prediction of uncertain trajectories in moving objects databases. *Applied Intelligence*, 33(3):370–386, 2010.

[50] K. Raptopoulou, A. Papadopoulos, and Y. Manolopoulos. Fast Nearest-Neighbor Query Processing in Moving-Object Databases. *GeoInformatica*, 7(2):113–137, June 2003.

[51] S. Saltenis and C. S. Jensen. Indexing of Moving Objects for Location-Based Services. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 463–472, California, USA, Feb. 2002.

[52] S. Saltenis, C. S. Jensen, S. T. Leutenegger, and M. A. Lopez. Indexing the Positions of Continuously Moving Objects. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 331–342, Texas, USA, May 2000.

[53] C. Shahabi, L.-A. Tang, and S. Xing. Indexing Land Surface for Efficient kNN Query. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 1020–1031, Aucklan, New Zealand, Aug. 2008.

[54] A. P. Sistla, O. Wolfson, S. Chamberlain, and S. Dao. Modeling and Querying Moving Objects. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 422–432, Birmingham U.K, Apr. 1997.

[55] J. Sun, D. Papadias, Y. Tao, and B. Liu. Querying about the Past, the Present, and the Future in Spatio-Temporal. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 202–213, MASSACHUSETTS, USA, Mar. 2004.

[56] Y. Tao, C. Faloutsos, D. Papadias, and B. L. 0002. Prediction and Indexing of Moving Objects with Unknown Motion Patterns. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 611–622, Paris, France, June 2004.

[57] Y. Tao and D. Papadias. Time-parameterized Queries in Spatio-temporal Databases. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 334–345, Wisconsin, USA, June 2002.

[58] Y. Tao and D. Papadias. Spatial queries in dynamic environments. *ACM Transactions on Database Systems, TODS*, 28(2):101–139, 2003.

[59] Y. Tao, D. Papadias, and J. Sun. The TPR*-tree: An Optimized Spatio-Temporal Access Method for Predictive Queries. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 790–801, Berlin, Germany, Sept. 2003.

[60] Y. Tao, J. Sun, and D. Papadias. Analysis of predictive spatio-temporal queries. *ACM Transactions on Database Systems, TODS*, 28(4):295–336, Dec. 2003.

[61] Y. Tao, J. Sun, and D. Papadias. Selectivity Estimation for Predictive Spatio-Temporal Queries. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 417–428, Bangalore, India, Mar. 2003.

[62] H. Tropf and H. Herzog. Multidimensional Range Search in Dynamically Balanced Trees. *Angewante Informatik*, 23(2):71–77, 1981.

[63] H. Wang, R. Zimmermann, and W.-S. Ku. Distributed Continuous Range Query Processing on Moving Objects. In *Proceedings of the International Conference on Database and Expert Systems Applications, DEXA*, pages 655–665, Krakow, Poland, Sept. 2006.

[64] T. Xia and D. Zhang. Continuous Reverse Nearest Neighbor Monitoring. In *Proceedings of the International Conference on Data Engineering, ICDE*, page 77, Georgia, USA, Apr. 2006.

[65] X. Xie, R. Cheng, and M. L. Yiu. Evaluating Trajectory Queries over Imprecise Location Data. In *Proceedings of the International Conference on Scientific and Statistical Database Management, SSDBM*, pages 56–74, Chania Crete, Greece, June 2012.

[66] M. L. Yiu, E. Lo, and D. Yung. Authentication of moving kNN queries. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 565–576, Hannover, Germany, Apr. 2011.

[67] M. L. Yiu, Y. Tao, and N. Mamoulis. The B^{dual}-tree: Indexing Moving Objects by Space Filling Curves in the Dual Space. *VLDB Journal*, 17(3):379–400, May 2008.

[68] X. Yu, K. Q. Pu, and N. Koudas. Monitoring K-Nearest Neighbor Queries Over Moving Objects. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 631–642, Tokyo, Japan, Apr. 2005.

[69] M. Zhang, S. Chen, C. S. Jensen, B. C. Ooi, and Z. Zhang. Effectively Indexing Uncertain Moving Objects for Predictive Queries. *PVLDB*, 2(1):1198–1209, 2009.

[70] R. Zhang, H. V. Jagadish, B. T. Dai, and K. Ramamohanarao. Optimized Algorithms for Predictive Range and KNN Queries on Moving Objects. *Information Systems*, 35(8):911–932, Dec. 2010.