Capturing the Uncertainty of Moving-Object Representations

Dieter Pfoser and Christian S. Jensen

Department of Computer Science, Aalborg University, DENMARK
{pfoser|csj}@cs.auc.dk

Abstract. Spatiotemporal applications, such as fleet management and air traffic control, involving continuously moving objects are increasingly at the focus of research efforts. The representation of the continuously changing positions of the objects is fundamentally important in these applications. This paper reports on on-going research in the representation of the positions of moving-point objects. More specifically, object positions are sampled using the Global Positioning System, and interpolation is applied to determine positions in-between the samples. Special attention is given in the representation to the quantification of the position uncertainty introduced by the sampling technique and the interpolation. In addition, the paper considers the use for query processing of the proposed representation in conjunction with indexing. It is demonstrated how queries involving uncertainty may be answered using the standard filter-and-refine approach known from spatial query processing.

1 Introduction

A relatively new research area, spatiotemporal databases concerns the management of objects with spatiotemporal extents, and real-world objects with continuously changing spatial extents are attracting substantial attention. The variety of applications suggests that there is not just one prototypical type of spatiotemporal application.

Spatiotemporal applications may be distinguished based on the data they manage, which may pertain to the past, the present, and the future, or a combination of these. For example, applications managing past data often conduct analyses of movements over time, answering queries such as, "What were the movements of the Vikings in the North Sea between year 1000 and year 1200?" Applications dealing with present and future data capture the current spatial extents of objects in the database and typically make predictions about the future extents of the objects. Sample queries include, "What is the position of flight SAS 286?" and "Where will flight SAS 286 be in 20 minutes?" Next, a specific type of application concerns real-world objects that move continuously and disregards the spatial extents of the objects, representing instead their positions as points. Candidate applications include fleet management, air traffic control, military command-and-control systems, and people tracking. This paper focuses

on the representation of the past and present positions of such moving-point objects.

Fundamental issues in these applications include the acquisition and representation of the movements of objects, including the inherent imprecision in the representation. For example, when representing the positions of vehicles based on sampling, the sampled positions are inherently imprecise, as are the interpolated positions in-between the samples. As a result, the record of the movements of objects as stored in the database differs from the actual movement. The imprecisions due to the measurements and caused by the use of sampling are inherently quite different. It is highly relevant to understand the nature of these imprecisions because this makes it possible to decide on their relative importance.

This paper's contributions are three-fold. First, it offers a proposal for representing the positions of moving-point objects in databases. Second, it quantifies the imprecisions in the proposed representation. The representation is modular, allowing the imprecision to be captured or not, depending on the application requirements. Third, the paper illustrates how the representation may be used in conjunction with indices to answer queries involving uncertainty. The two-step filter-and-refinement process known from spatial query processing is used together with error information.

Past database research has focussed on spatiotemporal applications where only the present and future positions of moving-point objects are relevant. In the context of applications that predict the movements of objects based on their current positions, speeds, and directions, Wolfson et al. (16) address position update policies and the imprecision involved in the database-representation of the positions. Next, Moreira et al. (9) present a data model for moving-point objects that is based on the decomposition of the trajectories of the objects into sections. In addition, so-called superset and subset semantics are proposed that aim to address uncertainty issues. A maximum error occurs when linearly approximating the movement of an object in-between samples, and this error is used in the process of query processing. However, this work is not connected to any specific application or technological context and thus does not cover the ranges of errors and the relationships between different error measures. The query processing aspects also do not consider the availability of indices. Güting et al. (5) present a comprehensive framework of abstract data types for moving objects. This work, however, does not address representation issues, nor does it accommodate uncertainty.

The outline of the paper is as follows. Section 2 presents an application scenario and describes a particular technological context for the application, the Global Positioning System. Section 3 proceeds to describe, quantify, and relate the measurement and the sampling errors in the context of the application scenario and accommodates also error information in the representation. This sets the stage for a proposal for a database representation for moving-point objects, presented in Section 4. Section 5 considers the utilization of this representation in query processing using indices. Finally, Section 6 concludes and offers directions for future research.

2 An Application Scenario—GPS-Based Fleet Management

This section presents a sample spatiotemporal application scenario, fleet management, and briefly introduces the Global Positioning System (GPS), the technology that is assumed used for sampling the positions of moving objects.

2.1 Fleet Management

The optimization of transportation, especially in highly populated areas, is a very challenging task that may be supported by an information system. An example fleet management project, conducted by the Department of Transportation of the State of California, Caltrans (3), aims to design what is termed the Advanced Transportation System. In this application, vehicles equipped with GPS devices transmit their positions to a central computer using either radio communication links or cellular phones. At the central site, the data is processed and utilized. Example queries occurring in such an application are as follows.

– Which taxi is closest to customer A?
– What is optimal taxi distribution over the area (somewhat related to pickups per area)?
– Compute the optimal route for a ride, considering road characteristics such as the actual and theoretical speed limits, congestions, accidents, etc.

Taking uncertainty into account, more sophisticated queries may be formulated.

– Which taxis were, with a 50% probability, within 100 meters of the Ritz hotel at 14.20 on April 22, 1999?
– How likely is it that taxi 1234 had visual contact with (was within 100 meters of) taxi 4321 between 9.00 and 13.00 on April 22, 1999?
– Which taxis were with 50% probability in Central Park at 10.00 on April 22, 1999.

2.2 Global Positioning System

The Global Positioning System is able to determine exact positions on Earth anytime, in any weather, and anywhere. The system consists of 24 satellites that orbit Earth at 20000 km. The satellites transmit signals that can be detected by GPS receivers, which then are able to determine their locations with great precision.

   The principle behind the GPS is the measurement of the distances between a receiver and several satellites. A total of four distances, and thus signals from four satellites, are needed to solve a set of four equations that expresses the latitude, longitude, height, and time (Magellan Corporation 8). The distance from the satellite to the receiver can be calculated by multiplying the time it takes for the signal to arrive by the speed at which it travels–the speed of light.

Although four visible satellites are enough to compute a position, the more satellites that are visible, the more precise the computed position becomes.

More information about the GPS can be found in, e.g., Magellan Corporation (8) and Leick (7).

3   Sampling and Uncertainty

This section covers how to acquire and represent the movement of point objects. We first give the technical means of how to determine the time-varying positions of moving point objects, and subsequently give a suitable way to represent the entire movement. An important part of the representation is the uncertainty caused by the acquisition process. The section describes the uncertainty caused by the measurement error and the sampling error, and it concludes with a discussion of the relative importance of these errors.

3.1   Acquiring Movement—Measuring Position in Time

In order to record the movement of an object, we would have to know the position at all times, i.e., on a continuous basis. However GPS and telecommunications technologies only allows us to sample an object's position, i.e., to obtain the position at discrete instances of time such as every few seconds.

The solid line in Fig. 1(a) represents the movement of a point object. Space (x and y axes) and time (t axis) are combined to form one coordinate system. The dashed line shows the projection of the movement in two-dimensional space (x and y coordinates).

A first approach to represent the movements of objects would be to store the position samples. For our database, this would mean we could not answer queries about the objects' movements at times in-between sampled positions. Rather, to obtain the entire movement we have to interpolate. The simplest approach is to use linear interpolation, as opposed to other methods such as polynomial splines (Bartels et al. 1). The sampled positions then become the end points of line segments of polylines, and the movement of an object is represented by an entire polyline in three-dimensional space. In geometrical terms, the movement of an object is termed a *trajectory* (we will use "movement" and "trajectory" interchangeably).

Fig. 1(b) shows the spatiotemporal space (the cube in solid lines) and several trajectories (the solid lines). Time moves in the upward direction, and the top of the cube is the time of the most recent position sample. The wavy-dotted lines at the top symbolize the growth of the cube with time.

3.2   Quantifying Uncertainty

The research on uncertainty in geospatial information is concerned with all sources of incorrectness and incompleteness in the measurement, analysis, and interpretation of digitally-represented, Earth-referenced phenomena (Unwin 13).
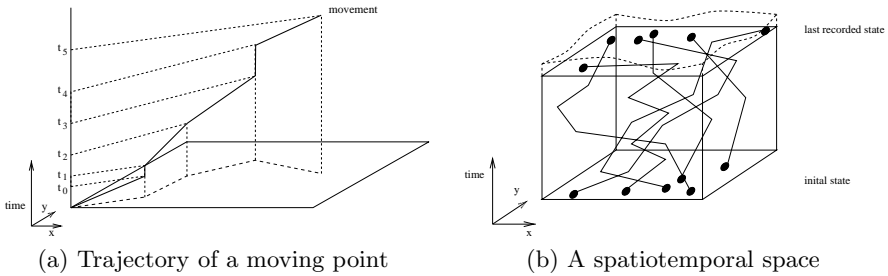
(a) Trajectory of a moving point   (b) A spatiotemporal space

Fig. 1. Movements and spaces

A representation of moving-point trajectories is inherently imprecise: imprecision is introduced by the measurement process used in the sampling of positions and by the sampling approach itself. A useful representation of moving points must take these uncertainties into account.

In this paper, we make the following assumptions.

– We will not consider any error connected to the times of measurements. We assume that we know precisely the time a position sample was taken. This assumption seems to be justified when using the GPS and its precise clocks as a measuring device.
– Within one application, we will only consider objects with similar movement characteristics, such as speed and range. Typical examples of objects with different characteristics include people, cars, and planes.

A first step in incorporating uncertainty into a representation of trajectories is to quantify it. We thus proceed to describe the errors introduced by the trajectory acquisition process.

3.3   Measurement Error

Generally, an error can be introduced by inaccurate measurements (Leick 7). The accuracy and thus the quality of the measurement depends largely on the technique used. This paper assumes that the GPS is used for the sampling of positions.

Two assumptions are generally made when talking about the accuracy of the GPS. First, the error distribution, i.e., the error in each of the three dimensions and the error in time, is assumed to be *Gaussian*. Second, we can assume that the horizontal error distribution, i.e., the distribution in the x-y plane, is circular (van Diggelen 14).

The error in a positional GPS measurement can be described by the probability function in Equation (1). The probability function is composed of two normal distributions in the two respective spatial dimensions. The mean of the distribution is the origin of the coordinate system. Fig. 2 visualizes the error distribution. In addition to the mean, the standard deviation, σ, is a characteristic parameter of a normal distribution . Within the range of $\pm\sigma$ of the mean,

in a bivariate normal distribution (2-dimensional), 39.35% of the probability is
concentrated.

$$\mathcal{P}_1(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$
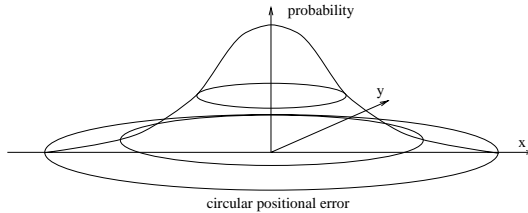

Fig. 2. Positional error in the GPS

Example 1. A typical GPS module used in vehicle navigation systems is the
CrossCheck AMPS Cellular from Trimble Navigation Ltd. This GPS/cellular
phone system has an error of $2m$ (equal to $1\ \sigma$) (Trimble Navigation Ltd. 12).
This measure refers to the standard deviation of a bivariate normal distribution
centered at the receiver's true antenna position.

3.4   Uncertainty in Sampling

We capture the movement of an object by sampling its position using a GPS
receiver at regular time intervals. This introduces uncertainty about the position
of the object in-between the measurements. In this section, we give a model for
the uncertainty introduced by the sampling, based on the sampling rate and the
maximum speed of the object.

Sampling Error The uncertainty of the representation of an object's movement
is affected by the frequency with which position samples are taken, the *sampling
rate*. This, in turn, may be set by considering the speed of the object and the
desired maximum distance between consecutive samples. Let us consider the
running example, in which we want to record the movements of taxis.

Example 2. As a requirement to the application, the distance between two con-
secutive samples should be maximally 10 meters. If the maximum speed of a taxi
is $150km/h$, this means that we would need to sample the position at least 4.2
times per second. If a taxi moves slower than its maximum speed, the distance
between samples is less than 10 meters.

We proceed to consider how the position samples resemble the true movement of
the object. Consider the three trajectories shown in Fig. 3. Each is possible given
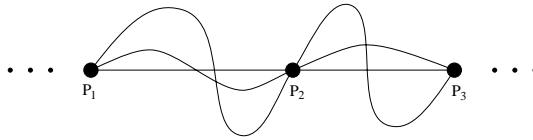the three measured positions P_1 through P_3. However, by just "looking" at the

Fig. 3. Possible trajectories of a moving object

three positions, one would assume that the straight line best resembles the actual trajectory of the object. Since we did not measure the positions in-between two consecutive position samples, the best we can do is to *limit the possibilities of where the moving object could have been.* We have to constrain the trajectory of the object by what we know about the object's actual movement. Considering the trajectory in a time interval $[t_1, t_2]$, delimited by consecutive samples, we know two positions, P_1 and P_2, as well as the object's maximum speed, v_m; see Fig. 4. If the object moves at maximum speed v_m from P_1 and its trajectory is a straight line, its position at time t_x will be on a circle of radius $r_1 = v_m(t_1 + t_x)$ around P_1 (the smaller dotted circle in Fig. 4). Thus, the points on the circle represent the furthest away from P_1 the object can gotten at time t_x. If the object's speed is lower than v_m, or its trajectory is not a straight line, the object's position at time t_x will be somewhere within the area bounded by the circle of radius r_1. Next, we know that the object will be at position P_2 at time t_2. Thus, applying
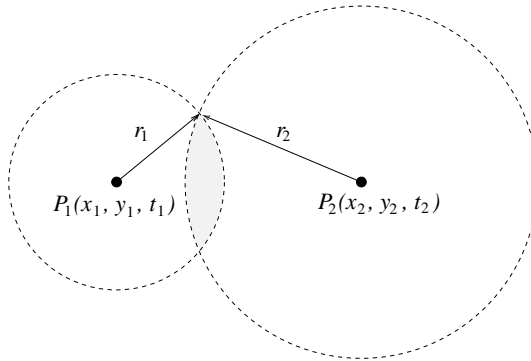

Fig. 4. Uncertainty between samples

the same assumptions again, the object's position at time t_x is on the circle with radius $r_2 = v_m(t_2 - t_x)$ around P_2. If the object moves slower or its trajectory is not a straight line, it is somewhere within the area bounded by this circle.

The above constraints on the position of the object mean that the object can be anywhere in the intersection of the two circular areas at time t_x. This intersection is shown by the shaded area in Fig. 4, and we use the term *lens*

for this intersection. Since we do not have any further information, we assume a uniform distribution for the position within the lens.

Thus, the sampling error at time t_x for a particular position can be described by the probability function shown in (2), where r_1 and r_2 are the two radii described above, s is the distance between the measured positions P_1 and P_2, and A denotes the area of the intersection of the two circles.

$$\mathcal{P}_2(x, y) = \begin{cases} \frac{1}{A} & \text{for } x^2 + y^2 \leq r_1^2 \wedge (x - s)^2 + y^2 \leq r_2^2 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Substituting $v_m(t_1 + t_x)$ and $v_m(t_2 - t_x)$ for the radii r_1 and r_2, respectively, the probability function shown in Equation (3) results. Its parameters are described in Table 1.

$$\mathcal{P}_2(x, y) = \begin{cases} \frac{1}{A} & \text{for } x^2 + y^2 \leq (v_m(t_1 + t_x))^2 \wedge \\ & \quad (x - s)^2 + y^2 \leq (v_m(t_2 - t_x))^2 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

For a visualization of a sampling error, refer to Fig. 5(a), in which the two horizontal axes depict x and y coordinates, and the vertical axis the positional probability.

Table 1. Parameters of the probability function, $\mathcal{P}_2$, describing the sampling error

v_m	maximum speed of the moving object
t_x	time for which the error distribution is computed
t_1	time of the first measured position
t_2	time of the second measured position
s	distance between the two positions, i.e., the length of the line segment
A	lens area, i.e., the area of the intersection of the two circles

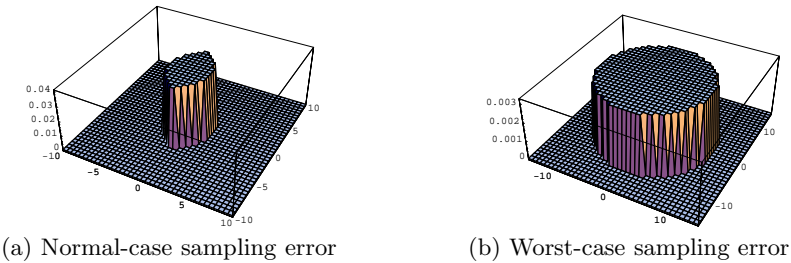(a) Normal-case sampling error   (b) Worst-case sampling error

Fig. 5. Probability functions for sampling errors

Sampling Error Across Time So far, we have quantified the sampling error for the position at a single point in time. To determine the error across time, as a first step, we compute the lens for various $t_x \in [t_1; t_2]$ as shown in Figs. 6(a)–(c).

The circle around the first point, P_1, measured at time t_1, is initially a point and grows as time advances, and the circle around the second point, P_2, shrinks with the advancement of time and eventually becomes a point. In the first situation in Fig. 6(a), the circle around P_2 contains the one around P_1, meaning that the constraint on how far away the object can be from P_1 at t_x is more restrictive than the constraint on how close it has to be to P_2. The area of intersection is the total circle or radius r_1. In the second situation, Fig. 6(b), the two circles start intersecting, and in Fig. 6(c) they show a clear intersection.

We observe that the intersection points of the two circles over time, i.e., for the cases the circles do actually intersect, lie on an *error ellipse* with positions P_1 and P_2 as its foci (cf. Fig. 7). The length of the semi-major axis is $2a = r_1 + r_2$. This is not surprising if we consider the definition of an ellipse. An ellipse is
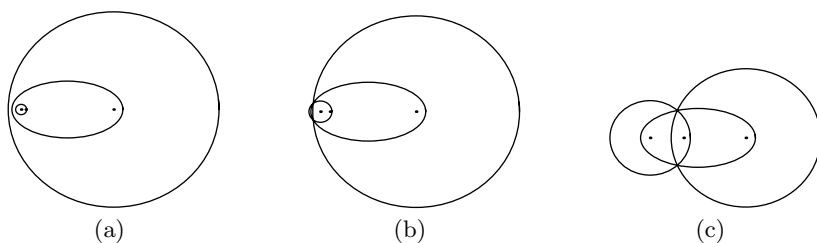
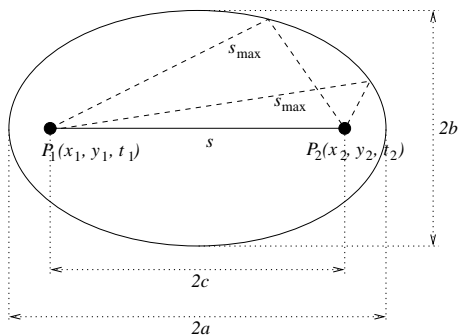

Fig. 6. Evolving sampling error

Fig. 7. Error ellipse

a curve consisting of all points in the plane whose sum of distances, r_1 and r_2, from two fixed points, P_1 and P_2 (the foci) separated by a distance of $2c$, is a given constant, $2a$. The measure $2c$ can be interpreted as the observed distance between P_1 and P_2, whereas $2a$ is the maximum distance the object can travel. The "thickness" of the ellipse, $2b$, is determined by the equation $b^2 = a^2 - c^2$.

This means that the smaller the difference between the observed distance, $2c$, and the maximum distance, $2a$, the "thinner" the ellipse. In the extreme case, the ellipse degrades to a line segment. In the worst case, where the object does not move between consecutive position samples, the ellipse becomes a circle.

Sampling Rate Having derived the general principle behind the sampling error, we give an example of how an increased sampling rate affects the error size. To illustrate the underlying principle, we use the error ellipse given in Fig. 7 as a measure for the size of the sampling error per line segment.

Example 3. In Fig. 8, we show the actual trajectory of a moving object as a bold line. As a first step, we sample the movement of the object at position P_1 and P_2. The time in-between the samples is 10 seconds. The shortest distance from P_1 to P_2 is 300 meters. Thus, to the best of our knowledge the object travels at a speed v of $30m/s$. If we further know the maximum speed of the object to be $42m/s$, we can draw an error ellipse around the line approximating the movement. The error ellipse has an eccentricity $2c = 300m$, a major axis $2a = v_{max} \cdot \Delta t = 42m/s \times 10s = 420m$, and a minor axis $2b = \sqrt{(2a)^2 - (2c)^2} = \sqrt{420^2 - 300^2} = 294m$. This rather large error ellipse means that the position of the object in-between samples is quite uncertain. Quadrupling the sampling rate, i.e., sampling the position every 2.5 seconds, leads to an error ellipse that has an eccentricity $2c = 80m$, a major axis $2a = v_{max} \cdot \Delta t = 42m/s \times 2.5s = 105m$, and a minor axis $2b = \sqrt{(2a)^2 - (2c)^2} = \sqrt{105^2 - 80^2} = 68m$.
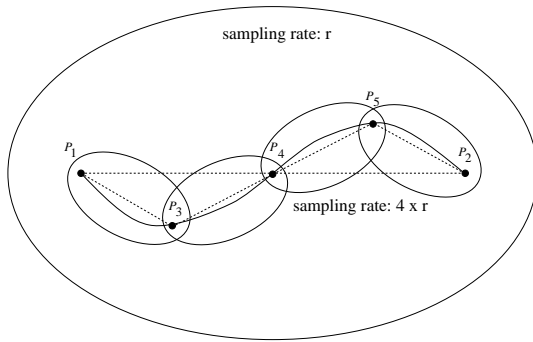

Fig. 8. Varying sampling rate

If we increase the sampling rate, the sample positions better approximate the movement, and the error introduced by sampling is decrease.

Maximum Speed Challenged An underlying assumptions so far has been that the maximum speed of a moving object is fixed at v_{max}. However, the more we know about the object in question, the further we can narrow down v_{max} and thus

reduce the uncertainty. For example, if we know that a taxi can reach $200km/h$, but regulations of the company set $120km/h$ as the upper limit, we may decide to assume that v_{max} is $150km/h$. Further examples of such additional information are local speed limits and road conditions; thus the maximum speed can vary depending on the area the taxi is in. Traffic volumes, which are time dependent, may also be taken into account. Further, there might be individual speed limits for drivers and cars. Generally, the more information we have about an object, the better we can adjust the sampling rate, reduce the error, and, consequently, minimize the uncertainty attached to its polyline trajectory.

Worst-Case Sampling Error Previously we identified the size and extent of the sampling error for a particular line segment and time. However, for use in Section 5, we also need an error measure in the situation where an object does not move between consecutive samples. In this case, the sampling error is determined by a circle of radius, r, equal to half the sampling interval multiplied by the maximum speed.

Example 4. Consider again the taxi from Example 3, whose position is sampled every 2.5 seconds. If the taxi is stopped, the eccentricity is $2c = 0$ (the foci coincide) and the error ellipse degrades to a circle. The major axis, $2a = v_{max} \cdot \Delta t = 42m/s \cdot 2.5s = 105m$, is equal to the minor axis. The radius of the circle then is 52.5 meters.

If we have no further information about the position of the object, all positions within the circle have the same possibility, yielding a circular uniform, worst-case error distribution, for which the probability function is given below, where r is the radius.

$$\mathcal{P}_3(x, y) = \begin{cases} 1/(r^2\pi) \text{ for } \sqrt{x^2 + y^2} \leq r \\ 0 \quad\quad \text{otherwise} \end{cases} \tag{4}$$

For a visualization of the worst-case sampling error, refer to Fig. 5(b), in which the two horizontal axes depict x and y coordinates and the vertical axis the positional probability.

3.5   Comparison of Error Sources

With current GPS technology, a moving object's position can be determined instantaneously with an accuracy of $2m$ (cf. Example 1), and this error will be reduced further with the advancement of GPS technology. How frequently position samples are taken depends on the particular application. In fleet management, determining the position every 2.5 seconds leads to a worst-case error of roughly $50m$. This is the radius of a circular distribution assuming that the maximum speed of the objects is $150km/h$, cf. Example 4). In practice the sampling rates will be much lower, thus allowing for worst-case errors of $200m$ or more.

It follows that the measurement error is small compared to the sampling error in fleet management. Therefore, we will consider only the uncertainty that stems

from the sampling, and disregard the uncertainty caused by the measurement technique, in the remainder of the paper.

4   A Representation for Moving Point Objects

Section 3 proposed a technique for capturing the movement of point objects that utilized polylines, and the section characterized the error introduced by this technique, this way also revealing the uncertainty inherent to the polyline representation. This section's objective is to provide a format for representing the history of the positions of continuously moving point objects, along with the uncertainty associated with our records of their positions. For this, we propose a relational database schema that incorporates all the spatiotemporal and error information previously presented in this paper.

Specifically, the schema in Table 2 defines relations for objects, for the line segments constituting the trajectories of the objects, and for the error information associated with the recorded trajectories. Relation **Object** has attributes

Table 2. Relational schema for capturing moving-point objects, their trajectories, and associated error information

Object	$< $ object_id, max_speed, *etc.* $>$
Line_segment	$<$ line_id, object_id, t_1, t_2, x_1, x_2, y_1, y_2, error_id $>$
Error	$<$ error_id, error_type, param1, param2 $>$

object_id, which is the key attribute, and max_speed, which determines the maximum speed at which the object can move. In addition this relation may include any number of attributes unrelated to the objects' spatial extents. Relation **Line_segment** captures the line segments that compose the trajectories of the objects. Attribute line_id is the key attribute; object_id is a foreign key referencing relation **Object**; and t_1 and t_2 are the times when the two positions, (x_1, y_1) and (x_2, y_2), constituting the line segment, were measured. Finally, relation **Error** contains the error information associated with the line segments. Attribute error_id is the key; error_type specifies the type of error that a tuple refers to, and thus specifies how parameters param1 and param2 are to be interpreted. In the current schema, there is only one type of error. However, if we consider more error sources in our application, additional types of errors may occur.

The domains of the attributes are as follows. Define $dom(x)$ to be a function that returns the domain of its argument attribute x. Then $dom(\text{object\_id}) = dom((\text{line\_id}) = dom(\text{error\_id}) = dom(\text{max\_speed}) = dom(t_1) = dom(t_2) = \mathcal{N}$, where $\mathcal{N}$ is the natural numbers, $dom(\text{param1}) = dom(\text{param2}) = \mathcal{N} \cup \text{NIL}$, $dom(x_1) = dom(x_2) = dom(y_1) = dom(y_2) = \mathcal{Z}$, where $\mathcal{Z}$ is the integers, and $dom(\text{error\_type}) = \{\text{worst-case\_sampling}\}$.

The following example illustrates how the above schema can be put to use.

Example 5. Our taxi company operates a number of taxis in a city. The database in Fig. 9 captures the movement of the taxis together with the associated errors.

This database permits the company to reconstruct the trajectories of its taxis and to compute the associated error information. All taxis are recorded in relation **Object**, and their trajectories are kept in **Line_segment** and are referenced through the foreign key object_id.

Object

object_id	max_speed
1234	120
4321	150
1235	140
⋮	

(a)

Line_segment

line_id	object_id	error_id	t_1	t_2	x_1	x_2	y_1	y_2
1	1234	1	...					
2	1234	1	...					
3	4321	1	...					
⋮			⋮					

(b)

Error

error_id	error_type	param1	param2
1	worst-case_sampling	25	NIL

(c)

Fig. 9. An example database containing positional and error information connected to a fleet management application of a taxi company

To utilize the error information, the parameters for the various probability functions are recorded. The parameter of the worst-case sampling error, $\mathcal{P}_3$, is the radius r, which in our database is stored as the param1 attribute value (25) of the only tuple in relation **Error**.

The parameters of the sampling error, $\mathcal{P}_2$, as shown in Table 1, are the distance s, which is computed from the attribute values for x_1, x_2, y_1, and y_2 in relation **Line_segment**, together with the times t_1, and t_2. The maximum speed v_m is stored in relation **Object**. Finally, t_x is not a static parameter that can be stored in a relation, but is an input parameter from a query. Thus, the intersection area A, which is different for each position in time contained in a particular line segment, can also only be computed once t_x is known.

5   Query Processing and Indexing

The objective of this section is to explore the use of error information when using indices for processing queries involving the positions of moving objects. The section first sets the context within spatiotemporal indexing for its contribution. Subsequently, it shows how a moving-point index may be put to use in the processing of spatiotemporal range queries involving positional uncertainty. A discussion of what types of queries that can be answered in the given framework is given. The section ends with a summary of the section's proposed approach.

5.1   Context

The purpose of spatiotemporal indexing is to efficiently support the retrieval of those objects, from a large set of objects, with spatiotemporal extents that satisfy a specified query predicate. The most commonly considered predicate is intersection with a specified region.

Substantial research is currently ongoing in spatiotemporal indexing, and a number of spatiotemporal indices have already been proposed; see Theodoridis et al. (11) for an overview. Although an index well suited for indexing the trajectories of the kinds of moving-point objects considered here still does not exist, it is expected that such an index will be invented.

In terms of the representation proposed in this paper, this means that we can expect to be able to index the polyline segments that represent trajectories. However, taking the uncertainty of the trajectories into consideration corresponds to the indexing of (non-point) objects with spatial extents, and the envisioned moving-point indices are no longer readily applicable.

Based on the assumption that it will be substantially more attractive to index the trajectories of moving-point objects than to index the trajectories of objects with spatial extents, which are more complex, this section offers an approach to using moving-point trajectory indices while taking into account the uncertainty of the trajectories and also taking into account query predicates relating to the uncertainty.

The approach employs the fundamental technique from spatial indexing of using approximations for the spatial extents to be indexed (Güting 4). For instance, R-trees generally use minimum bounding boxes. This use leads to a filter-and-refine strategy for query processing. First, based on the approximations, a filtering step is executed that returns a superset of the objects fulfilling the query predicate. Second, in the refinement step, the exact extents of the objects resulting from the first step are checked against the query predicate (Brinkhoff et al. 2).

5.2   Processing Uncertainty Queries

The goal here is to be able to use a moving-point index to answer queries such as "Retrieve the positions of taxis that were inside area A (specified as a rectangle) between times B and C with a probability of at least 30%?"

The first step is to specify the meaning of an object's position being within an area A with a probability of 30%. An object's position is described by means of a probability function centered around the positional mean (e.g., recall the probability function of the worst-case sampling error in Fig. 5(b)).

If all of an object's positional probability is within an area A, we say that the object is within area A for certain. We can determine if this is the case by integrating over the probability function with area A as the limit. If the result is 1, this is the case.

If the object is within area A with a probability of at least 30%, at least 30% positional probability has be concentrated within area A. This case is shown in

Fig. 10, where the circle represents the probability function of the worst-case sampling error, the rectangular shape is the query rectangle, and the shaded region represents the probability in the query window. The result of integrating over the probability function with rectangle A $= ([x_{min}, x_{max}], [y_{min}, y_{max}])$ as the limit thus has to be 0.3 or higher. Further, if the positional error is rota-
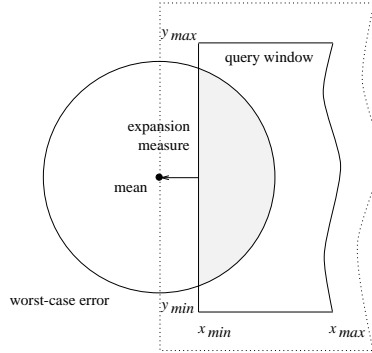

Fig. 10. Summing up the probability

tionally symmetric around the positional mean, as is the case for the worst-case sampling error, we can determine the maximum distance of the query window to the positional mean such that the probability of the position to be within the query window is 30% or higher. We term this computed distance the *expansion measure.* In Fig. 10 this distance is indicated by the arrow from the edge of the query window to the center of the probability function (the mean). The expansion measure can be interpreted as the measure by which the query window has to be expanded to contain the positional mean (the dotted query window in the figure).

Note that we are assuming that the query window is longer and wider than $2r$, the diameter of the error distribution; later in this section, we will revisit this assumption. However, for now, we proceed to show how the expansion measure can be used in the filter step.

The Filter Step We record the positions in time of the moving objects by means of line segments. The points on these segments are the mean values of the positional probability functions Our objective for the filter step is to retrieve those line segments that contain positions in time qualifying for the query result, i.e., those positions that, with the probability specified in the query, are in the query window. To retrieve these line segments, we intersect the expanded query window with the indexed line segments. Expanding the query window means that all positions with a probability higher or equal the one specified in the query (the one used to compute expansion measure) are contained in the query window.

For the filter step, the error measure used to determine the expansion measure can be coarse, but has to be universal so that it applies for all positions in the database. This is true for the worst-case sampling error described in Section 3.4.

As we shall see next, this method can only be applied if the probability specified in the query is less than 50%.

Consider again the above query, but with a probability of 60%. Using the worst-case sampling error leads to a negative query expansion measure (cf. Fig. 11(b)). If we use this smaller query window, we would retrieve a subset rather than a superset of the qualifying objects, since, e.g., positions that have no error (or a small error) and lie on (or close within) the borders of the query window would be disregarded. An example here is position P that has no error associated in Fig. 11(b). Shrinking the query window by the size of the negative expansion measure would eliminate this position from our set of candidate solutions.

This problem is solved by simply using the original query window with no expansion (shrinking) for probabilities higher than 50%. This means that we retrieve a superset of the qualifying objects.
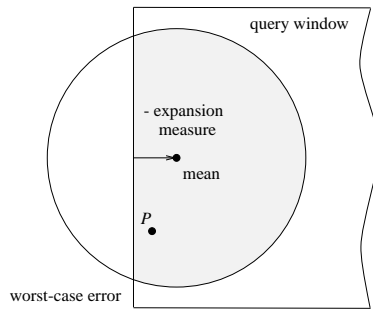

Fig. 11. Query window expansion: high probability

The Refinement Step To determine the final result, we have to evaluate the query predicate on all objects identified during the filtering step. In our case we have identified line segments that intersect with the transformed query region. As the final answer, we would like to have a set of positions, and so the refinement step extracts those parts of the line segments that qualify for this set.

In the filter step the intersection of line segments with the query window was determined with the help of the worst-case sampling error. To evaluate positions in time in the refinement step we will use the sampling error, unique for every position. A very straightforward way to achieve this is to apply the brute-force method of computing the probability functions in turn for all positions in time contained in a line segment (cf. Section 3.4) and check whether at least 30% probability is concentrated within area A. Fig. 12 shows two positions in time, P_1 and P_2, and their respective sampling errors (depicted by dotted lines). For

each of the positions, the probability concentrated within the query window is depicted by a shaded area. The set of solutions after the refinement step
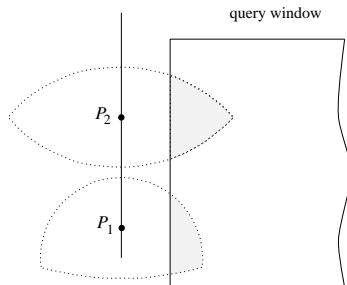

Fig. 12. Refinement step

comprises all positions in time whose positional probability within a given query window is at least as high as specified in the query.

On the Size of Query Windows Some query types deserve special attention within the presented framework. Point queries such as "Which taxis were in location A (point) at time B with 50% probability?" cannot be answered within this framework, since we cannot compute how much probability is concentrated within a point.

However, some "point" queries actually might be translated into window queries, e.g., location A might refer to a road crossing or a waiting area for taxis. In this case we are confronted with a small-window query.

Consider the above query where the query window of location A has an extent of, e.g., a taxi stand. If the sampling rate of the taxis' positions was very coarse, the positions have a high degree of uncertainty, and the sampling error is very large. To find an answer to our query, we have to determine the positions for which at least 50% of the probability is concentrated within the query window. If the query window is too small with respect to the error measure, no positions will qualify, e.g., consider query window QW_1 in Fig. 13.

On Non-Empty Query Results To derive a first minimum size of a query window for which the result is not guaranteed to be empty, we assume the worst case for both error and query. The largest possible error is the worst-case sampling error. The worst case of a query is to specify 100% probability, e.g., "Which taxis were in location A (point) at time B with 100% probability?" The smallest square query window we can consider has side length $2r$, the diameter of the worst-case sampling error, e.g., QW_2 shown in Fig. 13. If the query window is smaller, the probability of the worst-case sampling error cannot by contained entirely within the query window any more, i.e., less than 100% probability is concentrated within the query window, and the result is guaranteed to be empty.
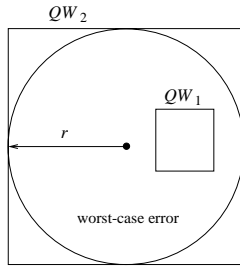
Fig. 13. Small-window query

However, this only depicts the worst-case scenario. For query windows with side length larger than $2r$, queries specifying varying degrees of uncertainty may have non-empty answers. If the query window is smaller and the query specifies 100% probability, certain positions are eliminated because their associated error is too large for them to be for certain (100%) in the query window. Although this seems only to eliminate solutions, it has significant consequences for the use of the worst-case error with the query window in the filter step, as will be explained next.

The Filtering Step Revisited In the case the query window is smaller than $2r$, the filtering as outlined earlier might eliminate positions that satisfy the query predicate. Consider the example shown in Fig. 14(a). First, we determine the expansion measure for a probability of 30% and the query window (the rectangle). The shaded area symbolizes the intersection of the error measure and the query window. The size of the area corresponds to the positional probability concentrated within the query window.

Using this expansion measure, however, would exclude qualifying positions, e.g., position P would be discarded in the filter step, although 30% or more of its actual positional probability (dotted lens shape of the sampling error) is concentrated within the query window.

To avoid the elimination of qualifying positions, we will initially expand the sides of small query windows that are smaller than $2r$ to be of size $2r$ and then use the resulting window to determine the expansion measure as describe earlier in this section. This is illustrated in Fig. 14(b), where the probability concentrated in the window of height $2r$ is symbolized by shading, and where the expansion measure is symbolized by the longer arrow. With this measure, position P will be in the set of candidate solutions. To recap, small window queries are addressed as follows. A query window can be arbitrarily small. In connection with databases considering spatial uncertainty, the size of the query window is also determined by the uncertainty specified in the query. Further, the "extent" of a spatial position stored in the database is determined by the associated error measure. Consequently, when specifying a query, one has to keep all these measures in mind not to retrieve an unwantingly small result.
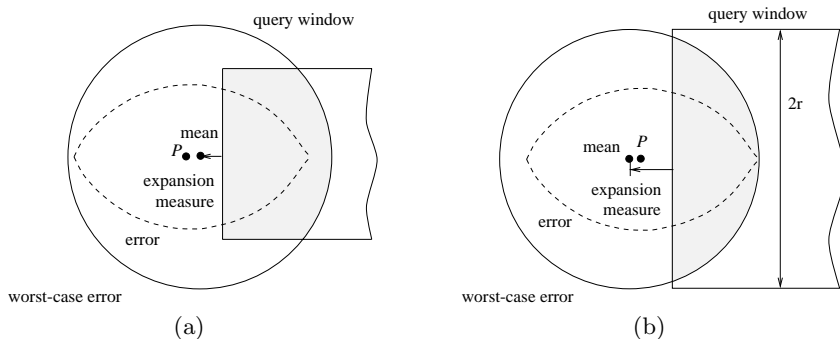
Fig. 14. Query window expansion: small query window

5.3   Summary of Approach

Or goal for this section was to give a method of how to use a moving point-index to process queries of the form "Retrieve the moving-object positions that were inside query rectangle A at some time between times B and C with a probability of at least X%." The trajectories are indexed using a moving point index that supports range queries. A superset of the qualifying positions are retrieved in a filtering step, in which we expand the query window to retrieve all line segments containing positions that are in the query window with probability at least X%. The expansion is determined using probability X and the worst-case sampling error, stored in the database.

In the refinement step, the positions contained in the retrieved line segments that actually are within query rectangle A with probability at least X% are identified. Here we use the sampling error, which is distinct for all positions. The following pseudo-algorithm summarizes the full retrieval procedure. We assume that the diameter of the worst-case sampling error is $2r$.

IF query window A has either height or width less than $2r$ THEN
    Increase the smaller side(s) to be of size $2r$;
IF X < 50% THEN
    Apply query window expansion to A;
Let S contain the result of searching the index with A;
Apply refinement to the line segments in S and return the resulting points;

6   Conclusions and Future Research

The paper investigates the representation of moving-point objects in databases. First, a set of queries derived from requirements to an application managing moving-point objects is presented. The Global Positioning System is the technology used for obtaining samples of the positions of these objects.

The paper proposes a method for acquiring and representing the movements of point objects. The positions of objects are sampled at selected points in time,

and the positions in-between these points in time are obtained using interpolation, thus capturing the complete movement.

The representation of movements is inherently imprecise, and the paper considers two types of errors, the measurement error and the sampling error. Two measures were derived for the sampling error, one pertaining to each position in time, and a global worst-case error. It is further shown that the measurement error can be ignored in the application context considered.

A database schema is proposed that incorporates both the polyline representation of movements as well as the parameters of the various error distributions associated with the polyline representation. The schema is illustrated by an example database suited for the taxi management application.

Finally, the paper shows how to use this database to answer spatiotemporal queries derived from the example application. The error information is used in connection with an arbitrary moving-point index to answer spatiotemporal queries using the standard filter-and-refinement process.

This work points to several directions for future research. First, for the representation of the movement, we chose to linearly interpolate in-between measured positions. More advanced techniques may be used for this purpose as well, e.g., polynomial splines. Second, two types of error measures were considered, namely the measurement error and the sampling error. Additionally manipulating the measured positions before storing them in the database introduces another error that needs to be considered. Thus, generalizing the approach to an arbitrary number of error measures poses an interesting challenge. Third, in our work we only consider uncertainty in the spatial dimensions (cf. Section 3.2). This is partly because of the high precision with respect to time of the positional measurement device, GPS, we use. Using motion sensors or other techniques instead poses the question of quantifying uncertainty with respect to time as well. Fourth, one of the underlying assumptions in our work is that objects are not restricted in their movements through space. In reality, the space considered will typically contain roads, railroad tracks, walls, floors, mountains, lakes, or other "infrastructure" that facilitate or inhibit movement. This infrastructure may be taken into account to yield a reduced overall uncertainty and error in the database, as well as other benefits.

Acknowledgements

References

Bartels, R. H., Beatty, J. C., and Barsky, B. A.: An Introduction to Splines for Use in Computer Graphics & Geometric Modeling. Morgan Kaufmann Publishers, Inc., 1987.

Brinkhoff, T., Kriegel, H. P., and Seeger, B.: Efficient Processing of Spatial Joins Using R-trees. *SIGMOD Record*, 22(2):237–246, 1993.

Caltrans New Technology and Research Program. Advanced Transportation Systems Program Plan. URL: <http://www.dot.ca.gov/hq/newtech/nt_page.html>.

Güting, R. H.. An Introduction to Spatial Database Systems. *VLDB Journal*, 3:357–399, 1994.

Güting, R.H., Böhlen, M., Erwig, M., Jensen, C. S., Lorentzos, N. A., Schneider, M., and Vazirgiannis, M.: A Foundation for Representing and Querying Moving Objects. Technical Report, Informatik-Bericht 238, FernUniversität Hagen, Germany, 1998.

Greenwalt, C. R. and Shultz, M. E.: Principals of Error Theory and Cartographic Applications. Technical Report, ACIC Technical Report No. 96, Aeronautical Chart and Information Center, St. Louis, MO, 1962.

Leick, A.: GPS Satellite Surveying. John Wiley & Sons, Inc., 1995.

Magellan Corporation, The : About Global Positioning - the Basics of GPS and GLONASS. URL: <http://www.ashtech.com/Pages/gpsndx.html>.

Moreira, J., Ribeiro, C., and Saglio, J.: Representation and Manipulation of Moving Points: An Extended Data Model for Location Estimation. *Cartography and Geographical Information Systems*, to appear.

Nascimento, M. A., Silva, J. R. O., and Theodoridis, Y.: Access Structures for Moving Points. Technical Report TR-33, TIMECENTER, Aalborg University, Denmark, 1998.

Theodoridis, Y., Sellis, T., Papadopoulos, A., and Manolopoulos, Y.: Specifications for Efficient Indexing in Spatiotemporal Databases. In *Proceedings of the 10th International Conference on Scientific and Statistical Database Management*, 1998.

Trimble Navigation Ltd.: CrossCheck AMPS Cellular. Product Datasheet, 1998.

Unwin, D. J.: Geographical Information Systems and the Problem of Error and Uncertainty. In *Progress in Human Geography 19*, 549–558, 1995.

van Diggelen, F.: GPS Accuracy: Lies, Damn Lies, and Statistics. In *GPS World*, 5(8), 1998.

Wolfram, S.: The Mathematica Book. Cambridge University Press, 1996.

Wolfson, O., Chamberlain, S., Dao, S., Jiang, L., and Mendez, G.: Cost and Imprecision in Modeling the Position of Moving Objects. In *Proceedings of the Fourteenth International Conference on Data Engineering*, 1998.