On the Effect of Location Uncertainty in Spatial Querying

Elias Frentzos, Kostas Gratsias, and Yannis Theodoridis, *Member*, *IEEE*

Abstract—An emerging topic in the field of spatial data management is the handling of location uncertainty of spatial objects, mainly due to inaccurate measurements. The literature on location uncertainty so far has focused on modifying traditional spatial search algorithms in order to handle the impact of objects' location uncertainty on the query results. In this paper, we present the first, to the best of our knowledge, theoretical analysis that estimates the average number of false hits introduced in the results of rectangular range queries in the case of data points uniformly distributed in 2D space. Then, we relax the original distribution assumptions showing how to deal with arbitrarily distributed data points and more realistic location uncertainty distributions. The accuracy of the results of our analytical approach is demonstrated through an extensive experimental study using various synthetic and real data sets. Our proposal can be directly employed in spatial database systems in order to provide users with the accuracy of spatial query results based only on known data set and query parameters.

Index Terms—Spatial databases, GIS.

1 Introduction

Spatial data management has been extensively researched during the last two decades [20]. A common assumption adopted in spatial databases is that the position of spatial objects is precisely known. However, a variety of sources such as the error of the GPS devices influence the accuracy of the recorded locations of spatial objects, making the location data obtained from measuring devices inherently imprecise. Moreover, several recent works [4], [7], [10] suggest that the location privacy of mobile users should be protected by adding a controlled degree of noise in each object's measured position. All these errors introduce uncertainty into the answers of traditional queries.

The literature on the management of the location uncertainty of spatial objects so far has dealt with either uncertainty representation issues [25], [26], [29] or probabilistic algorithms [6], [8], [16] that process spatial queries in the presence of uncertainty and estimate the probability of each spatial object to be included in the query result. Existing methodologies for handling uncertainty in commercial Spatial Database Management Systems (SDBMSs) also involve metadata, which are used to provide users with the accuracy or measurement error in each object's location. On the other hand, in this paper, we argue that there are cases where the user would prefer to know the influence of the measurement error in the query results, without actually executing the query. The challenge thus accepted in this paper is to *provide a theoretical framework that estimates the error introduced due to the uncertainty of objects' locations in the results of spatial rectangular range queries*. To the best of our knowledge, this is the first work that tackles this problem.

The estimation model we propose is applicable on data sets consisting of location data points. We initially model the uncertainty in a way similar to the one presented in [26] for spatiotemporal data. In particular, we represent the location uncertainty of each point using a disk (Fig. 1), called *uncertainty disk*, with the actual location of the point assumed to follow a uniform distribution within this disk. The recorded location of the data point and a fixed distance represent the center and the radius of the disk, respectively; although this statistical distribution may be assumed when artificially injecting uncertainty into the data objects [4], [7], [10], it is too simplistic to be able to capture the different distributions describing the measurement errors introduced by various devices. Therefore, in the sequel, we employ real-world statistical distributions [15], [17] and augmented histograms so as to support more realistic scenarios of spatial uncertainty.

A motivating scenario of our work is inspired by the emerging open agora paradigm [12]. Specifically, our scenario consists of an open agora of several distributed subscribe-based data sources containing the same spatial objects represented at different levels of uncertainty due to the different measurement methods and, consequently, different errors. Under this setting, we aim to provide a client-side query optimizer with a model that predicts the number of false hits introduced in the results of rectangular queries over each one of these data sources, given also that during the negotiation step [12], they publish aggregate-only data for their potential customers/users. Then, the optimizer would choose the most accurate among the provided data sources, i.e., the one with the smallest estimated number of false hits introduced in the query results, and proceed by posing the actual query to this particular data source.

The model described in this paper can be directly employed in existing SDBMS so as to estimate the average

• *The authors are with the Information Systems Laboratory, Department of Informatics, University of Piraeus, 80 Karaoli and Dimitriou St., GR-18534 Piraeus, Greece. E-mail: {efrentzo, gratsias, ytheod}@unipi.gr.*

Manuscript received 27 Apr. 2007; revised 14 Dec. 2007; accepted 17 July 2008; published online 30 July 2008.
Recommended for acceptance by C. Bettini.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2007-04-0188.
Digital Object Identifier no. 10.1109/TKDE.2008.164.

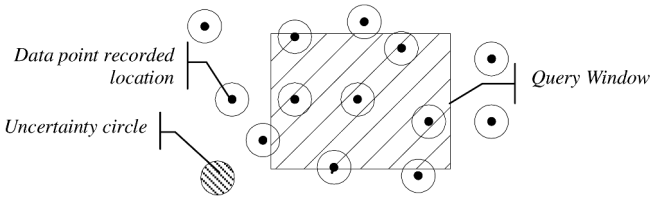


Fig. 1. Problem setting.

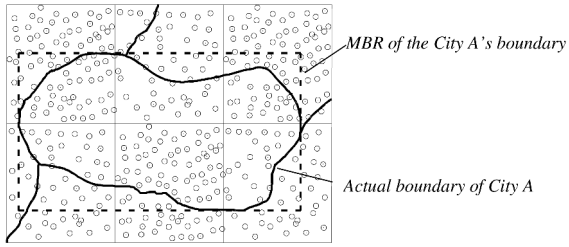


Fig. 2. Partial containment in SDWs.

number of false hits in query results due to location uncertainty. It could be therefore used in an interactive graphical query builder/analyzer, providing online an approximation of the percentage of the false hits due to location uncertainty along with other estimations such as selectivity, execution time, etc. Moreover, in a similar manner, our model can be utilized in order to determine the *maximum permitted* (im)precision of the data that will feed an SDBMS given the required accuracy in the results of rectangular range queries. Then, users can be guided by the SDBMS in the employment of the appropriate, more or less accurate—which also entails a more/less expensive—sampling method to be used for the data that will be fed to the system.

On top of these, the most prominent application of our model is over summary data, which contain aggregate-only information instead of actual data objects, e.g., the number of spatial objects inside a given spatial region. For instance, in a Spatial Data Warehouse (SDW) [14], aggregation may exhibit partial containment relationships instead of the total containment relationships normally assumed in conventional data warehouses; that is, a spatial cell may be contained in city A by 30 percent and in city B by 70 percent. Given that preaggregated information is only stored at the lowest level of the data warehouse location dimension hierarchy, i.e., the cells or base cuboids, a roll-up operation at the *city* level would, among others, aggregate over the number of partially contained cells. This situation is illustrated in Fig. 2, presenting the bounds between cities A, B, C, and D, a set of uncertain data points, and a regular grid standing for representing the cells containing the preaggregated information.

Consider, for example, a spatial data cube with the number of data points N_i contained inside each cell as a measure and also an uncertainty measure such as an aggregated output of a probabilistic range query, e.g., the average probability of each data object to be inside the cell P_i. In this case, each cell would be associated with a tuple (N_i, P_i). Given only this preaggregated information, currently, there is no way to redistribute the uncertainty of the objects contained inside each cell to the rolled-up spatial object, i.e., the city level. On the other hand, our model can still be directly applied utilizing aggregate information, i.e., the number of objects contained inside each cell N_i, and the radius of the uncertainty disk or standard deviation of the normal distribution, finally producing an approximation of the error introduced in the aggregation results. In particular, given that our model is capable of determining the effect of the location uncertainty in the Minimum Bounding Rectangle (*MBR*) of city A, considering it as a range query, it can approximate the effect in the actual spatial object A, involving only the cardinality of each cell, the MBR, and the uncertainty radius.

To the best of our knowledge, a theoretical study on modeling the error introduced in spatial query results in terms of false hits due to the location uncertainty of spatial objects is lacking. Perhaps the most relevant approach to our work is the analysis in [27], which discusses the effect of uncertainty in spatial queries in terms of the cardinalities of the three subsets of a range query result, namely, the *MUST*, *MAY*, and *ANS* sets; among them, the *MAY* set is the set of objects that "may" be located within a range query. Although this approach sounds very relevant to our proposal since we also provide a model that can be used in order to calculate the *false hits* in query results, the two methods are not directly comparable because the number of false hits is a *subset* of the *MAY* set, and it is not straightforward to determine the number of false hits directly from the cardinality of this set.

Outlining the major issues that will be addressed in this paper, our main contributions are the following:

- We prove two lemmas that estimate the average number of false positives and false negatives when executing rectangular range queries over spatial objects with location uncertainty, in the case of a uniformly distributed 2D data set. It is proved that both errors depend on the radius of the data point uncertainty disk and the perimeter of the query window, rather than its area.
- In order to relax the location uncertainty uniformity assumption and to utilize the real-world adapted bivariate normal distribution [15], [17], we efficiently approximate it with the uniform difference distribution. The results are close enough to the ones of the original analysis.
- We show how to utilize histograms in order to estimate the average number of false hits when we relax the uniformity assumption of objects' distribution in the data space, as well as to support various distributions of the uncertainty radius. The same methodology is also employed in other forms of summary data, e.g., data warehouses, in order to describe the effect of uncertainty.
- Finally, we report the results of the comprehensive set of experiments that we conducted over synthetic and real data sets demonstrating the correctness and accuracy of our analysis and also the efficiency of the proposed solution, employed on top of a commercial SDBMS, PostgreSQL [19] with PostGIS spatial extension [18]. It is worth to note that off-the-shelf spatial histograms, already used in SDBMS for query selectivity estimation, support our model without additional requirements.

TABLE 1
Table of Notations

Notation	Description
S, P, N	the unit data space [0,1]×[0,1], the point dataset, and its cardinality (also, density)
p_i, $p_i^\dagger$, d	a (recorded) point in P, its actual location, and the radius of the uncertainty disk
W_j, $W_{j,c1}$ - $W_{j,c4}$	the window of a rectangular range query, and its four corners (clockwise, starting from the lower-left)
$W_j^{x,L}$,$W_j^{x,U}$,$W_j^{y,L}$,$W_j^{y,U}$	the minimum and maximum coordinates of query window W_j along the x- and y- axes.
R, $R_{a\times b}$	the set of rectangular range queries over P and its subset with half-sides a and b along the x- and y- axes, respectively
$C(p_i,d)$, $A_{i,j}$	the uncertainty disk of point p_i with radius d and the portion of its area that lies inside (in the case of false negatives) or outside (in the case of false positives) W_j
$Dist(p_i,W_j)$	the minimum Euclidean distance between point p_i and the boundary of W_j
r_x, r_y	the distance of the closest to p_i point of the boundary of W_j along the x- and y- axes, respectively.
$A_{1x}(r_x,r_y)$, $A_{1y}(r_x,r_y)$	the area encompassed by a chord perpendicular to the x- (or y-) axes with r_x (or r_y, respectively distance from p_i and the respective arc of its uncertainty disk
$A_2(r_x,r_y)$	the overlapping area between the uncertainty disk of p_i and a query window corner being inside the disk, with r_x and r_y coordinates relatively to p_i.
$V_{i,j}$, $V_{1x}(r_x,r_y)$, $V_{1y}(r_x,r_y)$, $V_2(r_x,r_y)$	the volumes of the conical segments, equivalent to areas $A_{i,j}$, $A_{1,x}(r_x, r_y)$, $A_{1,y}(r_x, r_y)$, $A_2(r_x, r_y)$ when following the uncertainty uniformity difference assumption.
$AvgP_{i,P}(R_{a\times b})$ $AvgP_{i,N}(R_{a\times b})$	the average probability of a single point p_i to be false positive (or false negative) with respect to all query windows $W_j \in R_{a\times b}$
$E_P(R_{a\times b})$, $E_N(R_{a\times b})$	the average number of false positives (or false negatives) in the results of a rectangular range query $W_j \in R_{a\times b}$

The rest of the paper is structured as follows: Section 2 describes our theoretical analysis on the effect of location uncertainty under uniformity assumptions. In Section 3, we extend our model to support nonuniform distributions over the problem parameters. Section 4 evaluates the accuracy of our model through an extensive experimental study over synthetic and real data sets. Section 5 discusses related work, and Section 6 provides the conclusions of the paper and some interesting research directions.

2 Modeling Error Due to Location Uncertainty

Consider a data set P consisting of N *recorded* points p_i, $i = 1, \ldots, N$, uniformly distributed in the unit space $S = [0, 1] \times [0, 1]$. Due to location uncertainty, the *actual* position $p_i^\dagger$ of each point, also located in S, is uniformly distributed inside an *uncertainty disk* with center p_i and radius d. Let also R be the set of all rectangular range queries posed over the data set P and $R_{a\times b}$ be the subset of R containing all rectangular range queries having sides of lengths $2a$ and $2b$ along the x- and y-axes, respectively. Two types of errors are introduced when executing a range query $W_j \in R$ over the data set P:

- E_N is the set of *false negatives*, i.e., points qualifying the query window but not retrieved; formally, $E_N = \{p_i \in P : p_i \notin W_j | p_i^\dagger \in W_j\}$.
- E_P is the set of *false positives*, i.e., points retrieved while not qualifying the query window; formally, $E_P = \{p_i \in P : p_i \in W_j | p_i^\dagger \notin W_j\}$.

The problem is to make an as accurate as possible estimation of false negatives and false positives for a random W_j based only on known data set and query parameters.

From the above problem definition, it is clear that we initially make four main assumptions:

- A_I—*uncertainty uniformity assumption*. The actual points $p_i^\dagger$ are uniformly distributed inside the uncertainty disk $C(p_i, d)$.
- A_{II}—*data uniformity assumption*. The recorded points p_i are uniformly distributed in the data space.
- A_{III}—*constant uncertainty radius assumption*. The radius d of the uncertainty disk is constant.
- A_{IV}—*uncertainty size assumption*. Radius d is always less than the half of the length of the smallest side of query window W_j.

Regarding the first three assumptions (A_I-A_{III}), they will be relaxed in the model extension to be presented in Section 3. Regarding assumption A_{IV}, we argue that it is a reasonable property of the involved spatial objects since typical sizes of query window W_j are usually orders of magnitude larger than d; for example, points sampled with GPS devices usually introduce an error of a few meters (usually, less than 10 m), while query windows in real applications are expected to be at least hundreds of square meters. Having described our framework, in the next two sections, we prove two lemmas that are fundamental for our model. Table 1 summarizes the notations used in the rest of the paper.

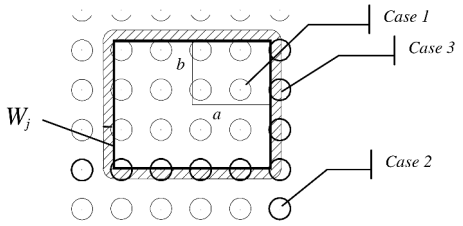


Fig. 3. Points contributing to the number of false negatives.

2.1 Estimating the Number of False Negatives

In this section, we prove a lemma that undertakes the calculation of the average number of false negatives.

Lemma 1. *The average number $E_N(R_{a\times b})$ of false negatives in the results of a rectangular range query $W_j \in R_{a\times b}$ with half-sides of lengths a and b over a point data set that follows the data uniformity and uncertainty uniformity assumptions is given by the following formula:*

$$E_N(R_{a\times b}) = N \cdot \left(\frac{8d}{3\pi}(a+b) - \frac{d^2}{2\pi}\right), \tag{1}$$

where d is the radius of the uncertainty disk.

Proof. The average number $E_N(R_{a\times b})$ of points being false negatives in the results of a rectangular range query $W_j \in R_{a\times b}$, i.e., $p_i \notin W_j | p_i^\dagger \in W_j$, can be obtained by the average probability $AvgP_{i,N}(R_{a\times b})$ of an arbitrary point p_i to be a false negative regarding an arbitrary query window $W_j \in R_{a\times b}$ multiplied by the total number N of data objects:

$$E_N(R_{a\times b}) = N \cdot AvgP_{i,N}(R_{a\times b}). \tag{2}$$

Obviously, our target is to determine $AvgP_{i,N}(R_{a\times b})$. Towards this goal, we formulate the probability that $p_i \notin W_j | p_i^\dagger \in W_j$. This probability is given by the ratio of the area $A_{i,j}$ of the portion of the uncertainty disk $C(p_i, d)$ included inside the query window over the total area of $C(p_i, d)$. However, $A_{i,j}$ is zero in cases where $C(p_i, d)$ does not overlap the query boundary.

Fig. 3 illustrates a query window W_j over a subset of uniformly distributed point data, extended by a buffer of width d: points with uncertainty disks being inside the query window, i.e., those labeled as "Case 1," cannot incur false negatives because they will be actually retrieved by the query. The same is also true for points with uncertainty disks located outside the buffer zone, illustrated as "Case 2" in Fig. 3. The single case where p_i is not retrieved by the query while $p_i^\dagger$ may be found inside W_j is when p_i is located inside the buffer zone that surrounds W_j, which is illustrated as "Case 3" in Fig. 3.

The above discussion expresses the fact that *a point p_i is a candidate to be a false negative if and only if p_i is located outside the query window while its uncertainty disk $C(p_i, d)$ intersects the query boundary*. Alternatively, p_i should be located inside the *Minkowski* region of W_j with radius d in order to be a candidate to be a false negative; this region can be determined by extending W_j with distance d on all directions [22]. Minkowski regions are directly derived from the concept of *Minkowski sum* [2] between the query window W_j and a disk of radius d, which, in our case, is composed by a set of line segments and circular arcs, illustrated as the boundary exterior of W_j in Fig. 3. The probability of a point p_i to be a false negative, regarding a query window W_j, is

$$P\left(p_i \notin W_j | p_i^\dagger \in W_j\right) = \begin{cases} \frac{A_{i,j}}{\pi d^2}, & if\ p_i \notin W_j\ and\ Dist(p_i, W_j) \le d, \\ 0, & otherwise. \end{cases} \tag{3}$$

The area $A_{i,j}$, as illustrated in Fig. 4, is determined by taking into account the uncertainty size assumption and distinguishing between three cases, illustrated in Figs. 4b, 4c, and 4d. In the first two cases, where the distance between p_i and each of the four corners of W_j is larger than d, $A_{i,j}$ is the portion of the uncertainty disk enclosed by 1) the chord c_1c_2 formed by the query side and the uncertainty disk and 2) the respective arc. Thus, it can be computed as the integral of the function of the chord length D, given as an expression of its distance, r_y or r_x (depending on which query side is regarded), from the disk center.

Let the chord c_1c_2 be parallel to x-axis (Fig. 4b), it holds that $D(r_x, r_y) = 2\sqrt{d^2 - r_y^2}$ and

$$\begin{aligned} A_{i,j} &= A_{1y}(r_x, r_y) \\ &= \int_{r_y}^{d} D(r_x, r_y) dr_y \\ &= 2\int_{r_y}^{d} \sqrt{d^2 - r_y^2} dr_y, \end{aligned}$$

resulting in[1]

$$A_{i,j} = A_{1y}(r_x, r_y) = d^2 \arctan\left[\sqrt{\left(\frac{d}{r_y}\right)^2 - 1}\right] - r_y\sqrt{d^2 - r_y^2}. \tag{4}$$

Equivalently, let the chord c_1c_2 be parallel to y-axis (Fig. 4c), the area $A_{i,j} = A_{1x}(r_x, r_y)$ is calculated by substituting r_y with r_x in (4). In the third case, where the distance between p_i and one of the four corners of W_j is less than d (Fig. 4d), $A_{i,j}$ can be determined in a similar way, resulting in

$$\begin{aligned} A_{i,j} &= A_2(r_x, r_y) \\ &= \frac{1}{2}\left[d^2 \operatorname{arccot}\left[\sqrt{\frac{r_y}{R^2 - r_y^2}}\right] - d^2 \arctan\left[\sqrt{\frac{r_x}{R^2 - r_x^2}}\right]\right. \\ &\quad \left. - r_y\sqrt{d^2 - r_y^2} - r_x\sqrt{d^2 - r_x^2} + 2r_x r_y\right]. \end{aligned} \tag{5}$$

1. All advanced calculations in this paper were performed using the *Mathematica* software [28].

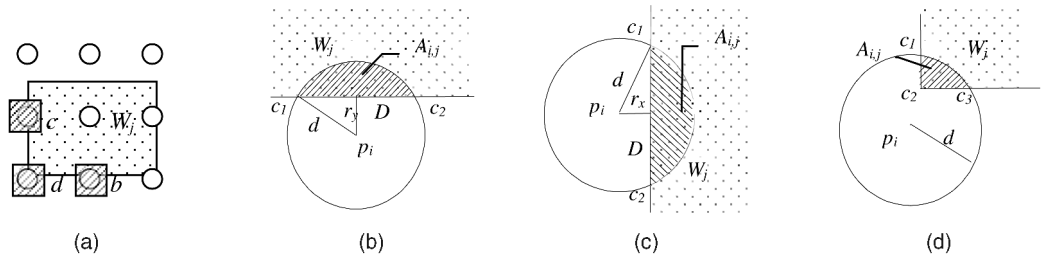


Fig. 4. (a) The unit space (a) and (b), (c), and (d) three details of it.

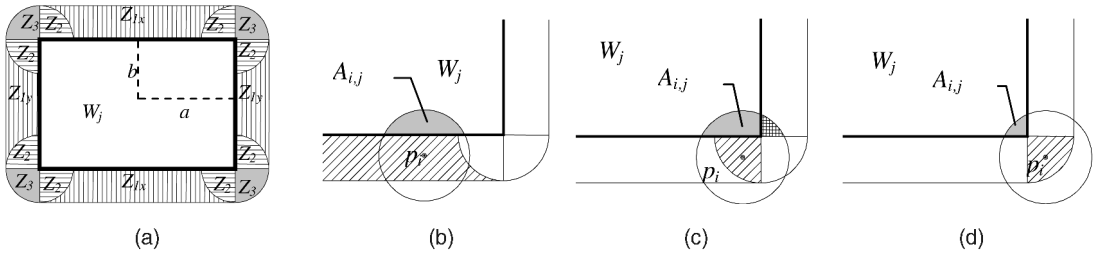


Fig. 5. Zones where area $A_{i,j}$ contributing in false negatives is expressed as a single function.

The average probability of a point p_i to be a false negative, with respect to any query window in $R_{a\times b}$, is calculated by integrating (3) over all possible query windows:

$$\begin{aligned} AvgP_{i,N}(R_{a\times b}) &= \int\limits_{W_j \in R_{a\times b}} P\Big(p_i \notin W_j | p_i^{\dagger} \in W_j\Big) dW \\ &= \iint\limits_{S} P\Big(p_i \notin W_j | p_i^{\dagger} \in W_j\Big) dx dy. \end{aligned} \tag{6}$$

In order to compute the above integral, it is necessary to determine the main zones inside which the area $A_{i,j}$ can be expressed as a single function. To facilitate discussion, Fig. 5a illustrates the fact that the area determined by $Dist(p_i, W_j) \leq d$ can be divided into three sets of zones inside of which a point p_i can be found, regarding the position of the query window: the first drawn with vertical stripes, the second drawn with horizontal stripes, and the shaded one, called Z_1, Z_2, and Z_3, respectively. Z_1 regions contain the data points such that the area resulted by the intersection of their uncertainty area with W_j forms a complete circular segment; alternatively, Z_1 regions are the locus of the points in the space such that they are outside W_j, their distance from W_j is smaller than d, and their distance from the four corners of W_j is greater than d. Z_2 regions are the locus of the points in the space such that points are outside W_j, their distance from W_j is smaller than d, and their x and y coordinates are inside the projection of W_j along the x- or y-axis, respectively; similarly, Z_3 regions differ only on that the x and y coordinates of their points are outside the projection of W_j along the x- or y-axis.

Zones $Z_{1,j}$, $Z_{2,j}$, and $Z_{3,j}$ associated with query window W_j are formally given by

$$\begin{aligned} Z_{1,j} = \big\{ & p \in S : p \notin W_j \wedge Dist(p, W_j) \\ & \leq d \wedge Dist(p, W_{j,c_i}) \geq d, i = 1 \ldots 4 \big\}, \end{aligned} \tag{7}$$

$$\begin{aligned} Z_{2,j} = \Big\{ & p \in S : p \notin W_j \wedge Dist(p, W_j) \\ & \leq d \wedge Dist(p, W_{j,c_i}) \leq d, \; i = 1 \ldots 4 \\ & \wedge \Big(p_x \in \Big[W_j^{x,L}, W_j^{x,U}\Big] \vee p_y \in \Big[W_j^{y,L}, W_j^{y,U}\Big]\Big) \Big\}, \end{aligned} \tag{8}$$

$$\begin{aligned} Z_{3,j} = \Big\{ & p \in S : p \notin W_j \wedge Dist(p, W_j) \\ & \leq d \wedge Dist(p, W_{j,c_i}) \leq d, \; i = 1 \ldots 4 \\ & \wedge \Big(p_x \notin \Big[W_j^{x,L}, W_j^{x,U}\Big] \wedge p_y \notin \Big[W_j^{y,L}, W_j^{y,U}\Big]\Big) \Big\}. \end{aligned} \tag{9}$$

Regarding zones of type Z_1, i.e., those labeled Z_{1x} and those labeled Z_{1y} in Fig. 5b, area $A_{i,j}$ can be computed using (4). When the relative positions of p_i and W_j constrain it to be inside a zone of type Z_2, $A_{i,j}$ can be computed by subtracting the small area at the upper right corner of the uncertainty disk (Fig. 5c), which is given by (5), from the overall uncertainty disk area being above the lower query side (4). Finally, for points inside zones of type Z_3, as illustrated in Fig. 5d, $A_{i,j}$ can be computed using (5). Summarizing, p_i may be found inside

- one out of two zones Z_{1x} (top and bottom in Fig. 5a), and two zones Z_{1y} (left and right in Fig. 5a); in these cases, $A_{i,j}$ is calculated by A_{1x} and A_{1y}, respectively,
- one out of four zones Z_3, one for each query window corner; in these cases, $A_{i,j} = A_2$, and
- one out of four zones Z_2, for each query window corner along the x-axis and another four along the y-axis; in these cases, $A_{i,j} = (A_{1x} - A_2)$, and $A_{i,j} = (A_{1y} - A_2)$, respectively,

Bearing in mind that 1) (6) integrates $P(p_i \notin W_j | p_i^{\dagger} \in W_j) = A_{i,j}/\pi d^2$ over the whole space S and 2) the value of $A_{i,j}$ is equal to zero in any other place, except of the zones Z_1, Z_2, and Z_3 where $A_{i,j}$ is provided in terms of

the relative position between p_i and W_j, i.e., r_x and r_y, (6) can be rewritten as follows:

$$AvgP_{i,N}(R_{a\times b}) = \frac{1}{\pi d^2}\left(\begin{array}{l} 2\iint_{Z_{1x}} A_{1x}(r_x,r_y)dr_ydr_x + 2\iint_{Z_{1y}} A_{1y}(r_x,r_y)dr_ydr_x \\ \quad + 4\iint_{Z_3} A_2(r_x,r_y)dr_ydr_x \\ 4\iint_{Z_2}\big(A_{1x}(r_x,r_y) - A_2(r_x,r_y)\big)dr_ydr_x \\ \quad + 4\iint_{Z_2}\big(A_{1y}(r_x,r_y) - A_2(r_x,r_y)\big)dr_ydr_x \end{array}\right) \Rightarrow$$

$$AvgP_{i,N}(R_{a\times b}) = \frac{1}{\pi d^2}\left(2\int\int_{Z_{1x}+2Z_2} A_{1x}(r_x,r_y)dr_ydr_x + 2\int\int_{Z_{1y}+2Z_2} A_{1y}(r_x,r_y)dr_ydr_x - 4\iint_{Z_3} A_2(r_x,r_y)dr_ydr_x\right). \quad (10)$$

The two $Z_{1x}+2Z_2$ areas involved in the above integrals may be regarded as the top and down rectangles in Fig. 5a formed by the Z_{1x} and the two Z_2 areas surrounding it, and their size along the x- and y-axes is $2a$ and d, respectively. The same also holds regarding the two $Z_{1y}+2Z_2$ areas, also having extents d and $2b$ along the x- and y-axes, respectively. According to this discussion, the above formula can be rewritten as follows:

$$AvgP_{i,N}(R_{a\times b}) = \frac{1}{\pi d^2}\left(2\int_0^d\int_0^{2a} A_{1x}(r_x,r_y)dr_xdr_y + 2\int_0^{2b}\int_0^d A_{1y}(r_x,r_y)dr_xdr_y - 4\int_0^d\int_0^{\sqrt{d^2-x^2}} A_2(r_x,r_y)dr_xdr_y\right). \quad (11)$$

Substituting $\int_0^d A_{1y}(r_x,r_y)dr_y = \int_0^d A_{1x}(r_x,r_y)dr_x$ with $2d^3/3$ and $\int_0^d\int_0^{\sqrt{d^2-x^2}} A_2(r_x,r_y)dr_ydr$ with $d^4/8$ in the above long expression, we get the following simple formula:

$$AvgP_{i,N}(R_{a\times b}) = \frac{8d}{3\pi}(a+b) - \frac{d^2}{2\pi}. \quad (12)$$

Substituting (12) into (2), we have proved Lemma 1. □

2.2 Estimating the Number of False Positives

In the sequel, we prove a similar lemma regarding the average number of false positives.

Lemma 2. *The average number $E_P(R_{a\times b})$ of false positives in the results of a rectangular range query $W_j \in R_{a\times b}$ with half-sides of lengths a and b over a point data set that follows the data uniformity and uncertainty uniformity assumptions is given by the following formula:*

$$E_P(R_{a\times b}) = N\cdot\left(\frac{8d}{3\pi}(a+b) - \frac{d^2}{2\pi}\right), \quad (13)$$

where d is the radius of the uncertainty disk.

Proof. The average number $E_P(R_{a\times b})$ of points being false positives in the results of a rectangular range query $W_j \in R_{a\times b}$, i.e., $p_i \in W_j | p_i^\dagger \notin W_j$, can be obtained by the average probability $AvgP_{i,P}(R_{a\times b})$ of an arbitrary point p_i to be a false positive regarding an arbitrary query window $W_j \in R_{a\times b}$ multiplied by the total number N of objects in the data space:

$$E_P(R_{a\times b}) = N\cdot AvgP_{i,P}(R_{a\times b}). \quad (14)$$

Then, following a methodology similar to that followed in the proof of Lemma 1, it is proven that the probability that $p_i \in W_j | p_i^\dagger \notin W_j$ is

$$P\Big(p_i \in W_j | p_i^\dagger \notin W_j\Big) = \begin{cases} \frac{A_{i,j}}{\pi d^2}, & if\ p_i \in W_j\ and\ Dist(p,W_j) \le d, \\ 0, & otherwise, \end{cases} \quad (15)$$

and the average, with respect to any W_j in $R_{a\times b}$, probability of a point p_i to be a false positive is

$$AvgP_{i,P}(R_{a\times b}) = \int_{W_j\in R_{a\times b}} P\Big(p_i \in W_j | p_i^\dagger \notin W_j\Big)dW = \iint_S P\Big(p_i \in W_j | p_i^\dagger \notin W_j\Big)dxdy. \quad (16)$$

The above integral is again computed by determining the zones inside which the area $A_{i,j}$ is expressed as a single function. These zones are found within the region formed by the original query window W_j and the *Minkowski difference* of W_j with a disk of radius d [26]. The Minkowski difference, also called *offsetting* [3], is a complementary operation to the Minkowski sum [26]; it is extensively studied in the field of computer graphics, while its calculation for convex polygons is a straightforward application of the known algorithms for computing the straight skeleton (equivalently, medial axis), requiring linear time [26]. Fig. 6a illustrates the three sets of zones, contained inside this area, namely, Z_1, Z_2, and Z_3, which can be defined in a way similar to the ones of the false negative computation. Formally

$$Z_{1,j} = \Big\{p\in S : p\in W_j \wedge Dist(p,W_j) \le d \wedge \Big(p_x \in \Big[W_j^{x,L}+d, W_j^{x,U}-d\Big] \vee p_y \in \Big[W_j^{y,L}+d, W_j^{y,U}-d\Big]\Big)\Big\}, \quad (17)$$

$$Z_{2,j} = \big\{p\in S : p\in W_j \wedge p\notin Z_{1,j} \wedge Dist(p,W_j) \le d \wedge Dist(p,W_{j,c_i}) \ge d,\ i=1\ldots 4\big\}, \quad (18)$$

$$Z_{3,j} = \big\{p\in S : p\in W_j \wedge Dist(p,W_{j,c_i}) \le d,\ i=1\ldots 4\big\}. \quad (19)$$

Regarding zones Z_{1x} and Z_{1y}, the area $A_{i,j}$ is computed using (4) (Fig. 6b). When in zone Z_2, $A_{i,j}$ is determined by summing up (4) along the x- and y-axes (Fig. 6c). Finally,

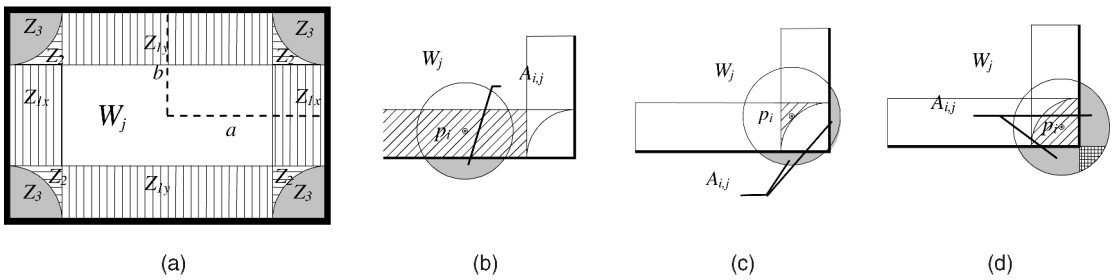


Fig. 6. Zones where area $A_{i,j}$ contributing in false positives is expressed as a single function.

points inside Z_3 are also computed by summing up (4) along the x- and y-axes and subtracting the small area in the lower right corner of the uncertainty disk (Fig. 6d), which is given by (5). Summarizing, p_i may be found inside

- one out of two zones Z_{1x} (top and bottom in Fig. 6a); in these cases, $A_{i,j}$ is calculated by A_{1x},
- one out of two zones Z_{1y} (left and right in Fig. 6a); in these cases, $A_{i,j}$ is calculated by A_{1y},
- one out of four zones Z_3, one for each query window corner; in these cases, $A_{i,j} = A_{1x} + A_{1y}$, and
- one out of four zones Z_2, for each query window corner; in these cases, $A_{i,j} = A_{1x} + A_{1y} - A_2$,

and (16) is reformulated as follows:

$$AvgP_{i,P}(R_{a\times b}) = \frac{1}{\pi d^2}\cdot\begin{pmatrix} 2\iint\limits_{Z_{1y}} A_{1y}(r_x,r_y)dr_ydr_x + 2\iint\limits_{Z_{1x}} A_{1x}(r_x,r_y)dr_ydr_x \\ +\,4\iint\limits_{Z_2}\big(A_{1x}(r_x,r_y)+A_{1y}(r_x,r_y)\big)dr_ydr_x \\ 4\iint\limits_{Z_3}\big(A_{1x}(r_x,r_y)+A_{1y}(r_x,r_y)-A_2(r_x,r_y)\big)dr_ydr_x \end{pmatrix}, \tag{20}$$

which, after the necessary calculations, results in

$$AvgP_{i,P}(R_{a\times b}) = \frac{8d}{3\pi}(a+b) - \frac{d^2}{2\pi}. \tag{21}$$

Substituting (21) into (14), we have proved Lemma 2. □

2.3 Discussion

Summarizing, the analytical model for the prediction of the number of false positives and false negatives when executing a rectangular range query over uniformly distributed point data consists of Lemmas 1 and 2 proved in the previous sections. It turns out that the average number of false negatives and false positives of an arbitrary query window with known sizes $2a$ and $2b$ along the x- and y-axes, respectively, is a function of a, b, the uncertainty radius d, and the cardinality N of the data set. Another result is the corollary that theoretically, the average number of false negatives is equal to the average number of false positives:

$$E_N(R_{a\times b}) = E_P(R_{a\times b}). \tag{22}$$

Such a result at a first thought seems counterintuitive, since from the geometry of the problem illustrated in Figs. 5a and 6a and the uniformity assumption, the number of false negatives is expected to be slightly higher than the number of false positives. However, it turns out to be correct when we take into consideration that on the one hand, the number of points contributing to the number of false negatives, represented by the shaded area in Fig. 5a, is greater than the respective one for false positives (Fig. 6a), and on the other hand, the area $A_{i,j}$ of the uncertainty disk of each point contributing to the number of false negatives (Fig. 5d) is smaller than the respective one for false positives (Fig. 6d). Our analytical calculation of $E_N(R_{a\times b})$ and $E_P(R_{a\times b})$ proves that the above two complementary factors finally result into two equal values for the number of false positives and false negatives, thus resulting in (22).

Moreover, it notably arises from (1) and (13) that the average number of false negatives and false positives of a rectangular range query depends on the query perimeter $(a+b)$, rather than the query area $(a\cdot b)$. A last observation is that when our model is utilized to determine the maximum permitted (im)precision of the data that will feed an *SDBMS*, (1) and (13) can be solved for the value of the uncertainty radius d, given the values of the required accuracy in terms of false hits and the query's typical extent. Intuitively, the two parts of the multiplier of N in (1) and (13), i.e., $8d(a+b)/3\pi$ and $d^2/2\pi$, represent the contribution in the total number of false hits of the length of the query perimeter and the four corners of the query window, respectively. This detail will turn out to be very useful in the next section when we will relax the data uniformity assumption with the aim of histograms.

A final issue regarding the model proposed in this section is the boundary effect, i.e., what happens in the boundaries of the unit space. Under such rare circumstances, it is only the points within distance d from the space boundary that are affected. The effect is described as follows:

- The number of false positives is expected to be less than the one calculated by our analysis, since there are no points outside the space boundaries which, after the injection of uncertainty, could be located inside the query and thus retrieved as false positives.
- The number of false negatives is expected to be equal with that of our original estimation, since the number of objects that qualified the query window but not retrieved is not affected.

In fact, during our experiments, we observed that when invoking query windows near the space boundaries, the accuracy of the estimation for both false positives and false negatives is only slightly affected, and the estimation error is very close to the one reported

regarding the general case; nevertheless, these experiments are not included in our experimental study due to space constraints.

3 Relaxing the Uniformity Assumptions

In this section, we relax the three assumptions, A_I-A_{III}, made in the problem definition in Section 2. This will be done in a gradually increasing order. We first show how to support real-world nonuniform uncertainty distributions, thus relaxing A_I (Section 3.1); we then employ spatial histograms in order to relax A_{II} (Section 3.2) and finally show how such histograms can be augmented so as to relax A_{III} (Section 3.3).

3.1 Relaxing the Uncertainty Uniformity Assumption

The analysis made in Section 2 was based on the uncertainty uniformity assumption, meaning that the actual position of each data point is uniformly distributed inside an uncertainty disk with the data point in the center and a known radius. Nevertheless, in this section, we extend our model towards nonuniform distributions of location uncertainty. The rationale behind this extension is that if the actual point $p_i^\dagger$ is located inside a circular neighborhood of p_i, it is more likely that the probability of a location being the actual location of $p_i^\dagger$ decreases as its distance from p_i increases. The argument that the uncertainty in real spatial data tends to be normally distributed is well established [5], [16], [17].

Even more, the error associated with GPS-tracked positions is usually assumed to be normally distributed, i.e., following the *bivariate normal distribution* with uncorrelated variables x and y, which is the extension of the normal distribution in 2D space [15]. Therefore, our goal in this section is to relax the uniformity assumption in location uncertainty and support the bivariate normal distribution. The respective probability density function (*pdf*), when variables x and y are uncorrelated, is given by the following expression [17]:

$$P_{BN}(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{23}$$

where σ^2 is the variance, along the x- and y-axes; then, σ is the corresponding standard deviation. However, the computation of the respective formulas as done in Section 2 is a hard task since it involves the integration of several exponential functions.

On the other hand, the density function of the bivariate normal distribution can be efficiently approximated by the *2D uniform difference distribution (2d-UDD)*, which is the extension of the *uniform difference distribution* in 2D space, i.e., the distribution of the difference between two uniformly distributed variables in $[0, d]$. The *pdf* of 2d-UDD forms a conical surface with base radius d and unit volume and is given by

$$P_{2d-UDD}(x,y) = \frac{3}{\pi d^2} \cdot \begin{cases} 1 - \frac{1}{d}\sqrt{x^2+y^2}, & if\ \sqrt{x^2+y^2} \le d, \\ 0, & otherwise. \end{cases} \tag{24}$$

Toward the reformulation of our model, we replace the uncertainty uniformity assumption with the following *uncertainty uniformity difference assumption*: the actual position of each data point is handled by $P_{2d\text{-}UDD}$ described above. Based on this assumption, we provide the following lemma.

Lemma 3. *The average numbers $E_N(R_{a\times b})$ and $E_P(R_{a\times b})$ of false negatives and false positives, respectively, in the results of a rectangular range query $W_j \in R_{a\times b}$ with half-sides of lengths a and b over a point data set that follows the data uniformity and uncertainty uniformity difference assumptions are given by the formula:*

$$E_N(R_{a\times b}) = E_P(R_{a\times b}) = N \cdot \left(\frac{2d}{\pi}(a+b) - \frac{3d^2}{10\pi}\right), \tag{25}$$

where d is the radius of the uncertainty disk.

Proof. $E_N(R_{a\times b})$ and $E_P(R_{a\times b})$ can be obtained from the average probabilities $AvgP_{i,N}(R_{a\times b})$ and $AvgP_{i,P}(R_{a\times b})$, respectively, multiplied by the total number N of objects in the data space. The probability of a point p_i to be a false negative or false positive, with respect to a query window W_j, is

$$P\left(p_i \notin W_j | p_i^\dagger \in W_j\right) = \begin{cases} V_{i,j}, & if\ p_i \notin W_j\ and\ Dist(p_i, W_j) \le d, \\ 0, & otherwise, \end{cases} \tag{26}$$

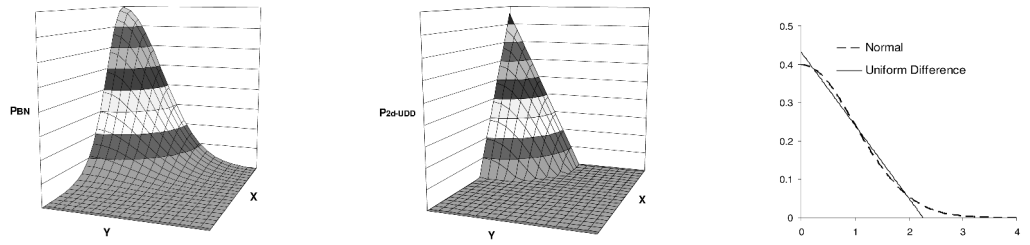
respectively

$$P\left(p_i \in W_j | p_i^\dagger \notin W_j\right) = \begin{cases} V_{i,j}, & if\ p_i \in W_j\ and\ Dist(p_i, W_j) \le d, \\ 0, & otherwise, \end{cases} \tag{27}$$

where $V_{i,j}$ is the volume of the *2d-UDD pdf* $P_{2d\text{-}UDD}$, contained fully inside or outside W_j, respectively.

The volume $V_{i,j}$ of the $P_{2d\text{-}UDD}$ being inside (outside, respectively) the query window is determined following the same methodology as in the proof of Lemma 1 (Lemma 2, respectively), taking also into account the uncertainty size assumption. In particular, bearing in mind that Figs. 4b, 4c, and 4d illustrate also the projection of $P_{2d\text{-}UDD}$ in the xy plane, we can employ them in our discussion: in the two first cases (Figs. 4b and 4c) where the distance between p_i and each of the four corners of W_j is more than d, $V_{i,j}$ is equal to $V_{1x}(r_x, r_y)$ (or $V_{1y}(r_x, r_y)$), which is the portion of the $P_{2d\text{-}UDD}$ being above (or right of, respectively) the vertical plane passing from c_1c_2. In the third case, where the distance between p_i and one of the four corners of W_j is less than d (Fig. 4d), $V_{i,j}$ is equal to $V_2(r_x, r_y)$, which is the portion of the $P_{2d\text{-}UDD}$ being right of the vertical plane passing from c_1c_2 and above the one passing from c_2c_3.

The average, with respect to any query window in $R_{a\times b}$, probability of a point p_i to be a false negative (false positive, respectively) is calculated by integrating (26) ((27), respectively) over all query positions as in (6) ((16), respectively). The corresponding integral is computed in the same way as the one followed in the proof of Lemma 1 (Lemma 2, respectively) by replacing the values of $A_{1x}(r_x, r_y)$, $A_{1y}(r_x, r_y)$, and $A_2(r_x, r_y)$ with $V_{1x}(r_x, r_y)$,



(a) (b) (c)

Fig. 7. (a) Bivariate normal distribution. (b) Two-dimensional uniform difference distribution. (c) Best fitting in a single dimension.

$V_{1y}(r_x, r_y)$, and $V_2(r_x, r_y)$ into (11) ((20), respectively). Then, by substituting $\int_0^d V_{1y}(r_x, r_y)dr_y = \int_0^d V_{1x}(r_x, r_y)dr_x = d/2\pi$ and $\int_0^d \int_0^{\sqrt{d^2-x^2}} V_2(r_x, r_y)dr_y dr_x = 3d^2/40\pi$ and performing the necessary calculations, we get

$$AvgP_{i,N}(R_{a\times b}) = \frac{2d}{\pi}(a+b) - \frac{3d^2}{10\pi} \tag{28}$$

and

$$AvgP_{i,N}(R_{a\times b}) = \frac{2d}{\pi}(a+b) - \frac{3d^2}{10\pi}. \tag{29}$$

By multiplying the above formulas with N, we have proved Lemma 3. □

Up to this point, given that the distribution of the actual data point follows the uncertainty uniformity difference assumption, our model constitutes of (25), which is much alike the ones in Section 2 under the uncertainty uniformity assumption. In particular, when (25) is compared with (1) and (13), the formulas differ only in the weights of the function variables $d(a+b)$ and d^2. Although the model described above does not directly consider the bivariate normal distribution, it can be used to efficiently approximate it. Consider, for example, Fig. 7, which illustrates the probability function of the bivariate normal distribution with uncorrelated variables (Fig. 7a), the probability function of the 2d-UDD (Fig. 7b), and the silhouette of the two distributions in 1D space (Fig. 7c); the two probability functions turn out to be close to each other. Hence, we can utilize least squares and estimate the radius of the cone that fits best in the Gaussian "bell" of the bivariate normal distribution.

Formally, we provide the following lemma.

Lemma 4. *The 2d-UDD that best approximates the bivariate normal distribution with uncorrelated variables is taken when the radius d of the uncertainty disk is*

$$d \approx 2.36533 \times \sigma, \tag{30}$$

where σ is the standard deviation of the bivariate normal distribution along the x- and y-axes.

Proof. The proof can be found in the Appendix, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TKDE.2008.164. □

Concluding, our model for normally distributed uncertainty constitutes of (25) and (30); the value of d provided by (30) can be directly used as input in (25) in order to approximate the normal distribution quite effectively, as it will be shown later in the experimental study.

3.2 Relaxing the Data Uniformity Assumption

Sections 2 and 3.1 assumed that data points are uniformly distributed in the data space. In this section, we relax the data uniformity assumption and apply our approach to arbitrarily distributed data with the employment of *histograms* [11], [13]. Histograms have been widely used in query optimization issues such as spatial selectivity estimation [1], [24], in order to overcome similar assumptions made when estimating the number of disk page accesses required to answer a query. The background idea is that when data are included in a small space, they may be considered as uniform even though the distribution of the entire data set may differ significantly. The goal, therefore, when using histograms is to break down the space into small regions, called *buckets*, which can be assumed to contain uniform data. Among the schemes proposed, we adopt the concept of [1], since it can be directly applied in our requirements.

In particular, Acharya et al. [1] present several space partitioning techniques for the construction of spatial histograms utilized in selectivity estimation of range queries. Among them, the *MinSkew* technique has been shown to provide the most accurate selectivity estimates for spatial queries. *MinSkew* is a binary space partitioning (BSP) technique employing the *spatial skew* definition provided in [1]. More specifically, the spatial skew of a bucket is the statistical variance of the spatial densities of all points grouped within this bucket, and the spatial skew of the entire set is the weighted sum of spatial skews of all buckets. The proposed technique uses a uniform grid of regions and their spatial densities as input. Then, the histogram construction algorithm repeatedly partitions the given set of regions such that the spatial skew is minimized at each step until no more buckets are available for the histogram. Since it always partitions an existing region into two, the result is a BSP. As a result, the constructed histogram H is the set of n buckets $H = \{B_i : \cup(B_i) = S \wedge \cap(B_i) = \emptyset\}$ and $B_i = \{[x_{i,L}, x_{i,U}], [y_{i,L}, y_{i,U}]\}$. The main advantage of this technique is that the initial cells grouped together within the same bucket have small spatial skew, i.e., variance. It is therefore expected that the cells contained inside each bucket should enclose approximately the same number of data points; as a result, it is usually assumed that the data distribution inside each bucket B_i is uniform. Actually, this assumption, as demonstrated both in [1] and in our experiments, is rather reasonable

even in the presence of totally skewed spatial data such as the *Charminar* data set [1].

Regarding our case, we utilize *MinSkew* histograms in order to apply our analysis in nonuniform data and estimate the error introduced in the query results without actually executing the query. In the sequel, we propose two alternative approaches for estimating $E_P(R_{a\times b})$ and $E_N(R_{a\times b})$. The first approach determines the histogram buckets that overlap the query and then calculates the local density N' by producing the weighted average of the overlapped bucket densities N_i. This happens by weighting the density N_i of each bucket B_i with the corresponding area A_i that partially covers the query window, normalized by the total query area:

$$N' = \frac{1}{4ab} \sum_{i=1..n} (N_i \cdot A_i). \quad (31)$$

Hence, $E_P(R_{a\times b})$ and $E_N(R_{a\times b})$ can be estimated by evaluating (1), (13), or (25) using the local density N', derived by (31), instead of the overall space density N.

As an alternative approach, instead of computing a global local density N' for the total query window, we consider the different contributions of the query window sides and query window corners in the total number of false hits, as discussed in Section 2.3. Therefore, given a spatial histogram containing n disjoint buckets B_i, the estimation of the number of false positives and false negatives in the results of a query window under the uncertainty uniformity assumption can be determined using the following formula:

$$E_P(R_{a\times b}) = E_N(R_{a\times b}) = \sum_{i=1..n} \left[N_i \cdot \left(\frac{2d}{3\pi} L_i - \frac{d^2}{8\pi} s_i \right) \right], \quad (32)$$

where L_i is the length of the part of the query perimeter that overlaps B_i, and s_i is the number of query window corners being inside B_i.

Equation (32) formulates the fact that the total number of false negatives or positives is the summation of the contributions of the different query components, as discussed in Section 2.3. More specifically, the $2dL_i/3\pi$ part of (32) is derived from the $8d(a+b)/3\pi$ of (1) and (13) multiplied by the length of the query perimeter L_i that overlaps bucket B_i and divided by the total query length $4(a+b)$; in the same manner, the $d^2 s_i/8\pi$ part of (32) is the transformation of the $d^2/2\pi$ part of (1) and (13) multiplied by the actual number of query window corners s_i inside bucket B_i and divided by their total number, i.e., four.

Consider, for example, Fig. 8a, which illustrates a query window W overlapping four histogram buckets $(B_1 \dots B_4)$. Since false hits may only be found close to the boundary of W, the number of false positives or negatives on bucket B_1 depends on the length of the query perimeter that overlaps it, that is, the length of lines $|m_1c_1| + |c_1m_2|$ and the number of corners $s_1 = 1$. It is also worth to note that using the above procedure, the query window is not dissected across the histogram buckets' boundaries, as such an approach would increase the total perimeter and consequently decrease the accuracy of the model. Moreover, in the *2d-UDD* uncertainty distribution case, the formula for estimating the number of false positives and false negatives is

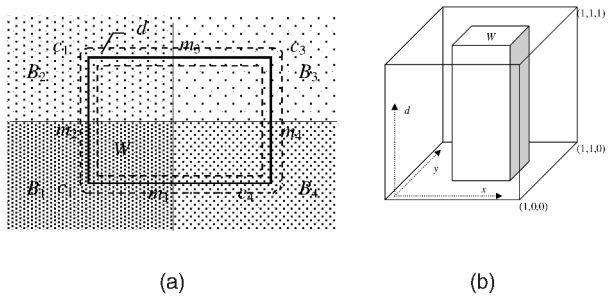


Fig. 8. (a) A query window over of a spatial histogram. (b) A spatial query window over the augmented space.

$$E_P(R_{a\times b}) = E_N(R_{a\times b}) = \sum_{i=1..n} \left[N_i \cdot \left(\frac{d}{2\pi} L_i - \frac{3d^2}{40\pi} s_i \right) \right]. \quad (33)$$

The above formula is derived by counting the different contributions of the query sides and corners of (25) in a way similar with the above. In particular, the $dL_i/2\pi$ part of (33) is computed by multiplying the $2d(a+b)/\pi$ of (25) by the part of the query perimeter L_i that overlaps bucket B_i, divided by the total query length $4(a+b)$, while the $3d^2 s_i/40\pi$ part of (33) is obtained by multiplying the $3d^2/10\pi$ part of (25) by the actual number of query window corners s_i inside bucket B_i, divided by their total number, i.e., four.

The same methodology can be applied to any bucket-based data storage scheme containing summary information, such as data cubes in SDWs. Since a spatial data cube consists of disjoint spatial buckets, i.e., the base cuboids, along with summary information, (32) and (33), depending on the type of uncertainty distribution, can be applied in OLAP operations and produce an estimation for the total number of false positives or false negatives. For example, when aggregating from the *cell* to the *city* level as discussed in the introduction, i.e., performing a roll-up operation, the MBR of a city can be considered as a query window and be used to estimate the false hits introduced in the aggregation. Given, however, that the density between the boundary of the actual city and its MBR can be much different, the N_i involved in (32) or (33) should be determined by using the actual perimeter of the city polygon in place of its MBR, and the L_i lengths should be weighted accordingly using the MBR and the polygon perimeter. This approach will be tested in the experimental section, and it will be shown to produce very good estimations.

3.3 Relaxing the Constant Uncertainty Radius Assumption

The third extension of our model in order to support real-world application scenarios is to deal with point data with different values of the uncertainty radius or standard deviation for each one. Consider, for example, m sets P_j containing N_j points each, obtained by using different positioning technologies such as GPS, Wi-Fi positioning, etc. Then, the union of all sets $P = \bigcup_{i=1...m} \{P_j\}$ contains points having several uncertainty radii depending on each point's original data source. A straightforward approach in order to determine the error E_P or E_N introduced in the results of a rectangular range query over P is to calculate

the specific errors $E_{P,j}$ or $E_{N,j}$ for each one P_j separately and then summarize the resulting errors. More formally

$$E_P(R_{a\times b}) = \sum_{j=1..m} E_{P,j}(R_{a\times b}) \text{ and } E_N(R_{a\times b}) = \sum_{j=1..m} E_{N,j}(R_{a\times b}). \quad (34)$$

Such an approach would reasonably be successful when dealing with uniformly distributed data. However, when dealing with real-world usually skewed data, the methodology provided in Section 3.2 should be applied, meaning that we would have to maintain m different histograms, one for each different possible value of the uncertainty radius. Nevertheless, in this paper, we provide a more sophisticated solution to the above challenge. Specifically, we propose to augment a simple spatial histogram such as *MinSkew* [1] with the uncertainty radius considered as the third dimension. In other words, we propose to use the MinSkew histogram in the normalized 3D space formed by the two spatial dimensions and the length of the uncertainty radius d.

More formally, the proposed MinSkew histogram is $H=\{B_i : \cup(B_i)=S\times[0,1] \wedge \cap(B_i)=\emptyset\}$ and $B_i=\{[x_{i,L}, x_{i,U}], [y_{i,L}, y_{i,U}], [d_{i,L}, d_{i,U}]\}$. It is built by applying a uniform grid in $S\times[0,1]$ and counting the number of data points found inside each cell in the 3D space and then recursively partitioning the data space, minimizing the total spatial skew at each step. Following the respective discussion of the previous section regarding simple spatial histograms, it is assumed that the data distribution inside each 3D bucket B_i is uniform. Then, the estimation of the number of false hits can be easily calculated in the case of the uncertainty uniformity assumption as follows:

$$E_P(R_{a\times b}) = E_N(R_{a\times b}) = \sum_{i=1..n}\left[\frac{N_i}{d_{i,U}-d_{i,L}}\cdot\int_{d_{i,L}}^{d_{i,U}}\left[\left(\frac{2d}{3\pi}L_i-\frac{d^2}{8\pi}s_i\right)dd\right]\right], \quad (35)$$

where L_i is the length of the query perimeter that overlaps bucket B_i in the two spatial dimensions, s_i is the number of query window corners being inside bucket B_i, and $d_{i,L}$ and $d_{i,U}$ are the lower and upper values of the third dimension d in B_i, respectively. Equation (35) is directly derived when integrating (32) over all possible values of d in the data space, bearing also in mind that the actual number of objects found at each slice of the third dimension is $N_i/(d_{i,U}-d_{i,L})$ and $(d_{i,U}-d_{i,L})$ is the bucket's extent along this dimension. Intuitively, the above formula expresses the fact that the total error is the summation of the errors encountered on each histogram bucket the query window boundary overlaps; moreover, in this case, the spatial query window W is also augmented in the third dimension, forming a box entirely covering the third dimension, as illustrated in Fig. 8b. Finally, (35), after the necessary calculations, turns into

$$E_P(R_{a\times b}) = E_N(R_{a\times b}) = \sum_{i=1..n}\left[N_i\cdot\left(\frac{d_{i,U}+d_{i,L}}{3\pi}L_i-\frac{d_{i,U}^2+d_{i,L}^2+d_{i,L}d_{i,U}}{24\pi}s_i\right)\right]. \quad (36)$$

Following a similar approach, the estimation of the number of false hits in the case of the uncertainty uniformity difference assumption is calculated as

$$E_P(R_{a\times b}) = E_N(R_{a\times b}) = \sum_{i=1..n}\left[N_i\cdot\left(\frac{d_{i,U}+d_{i,L}}{4\pi}L_i-\frac{d_{i,U}^2+d_{i,L}^2+d_{i,L}d_{i,U}}{40\pi}s_i\right)\right]. \quad (37)$$

The proposed approach has two basic advantages regarding the alternative of maintaining different histograms for the m sets of recorded points; the first is that the space requirements are sufficiently reduced, especially in the case where the number of different uncertainty radii increases significantly. However, the most important advantage of our proposal is revealed bearing in mind that data belonging to the same class may have different accuracy; for example, the uncertainty due to GPS depends on a large number of parameters such as the number of visible satellites, the frequency interference, and the satellite signal reflection in large glass surfaces inside urban areas, resulting in a different uncertainty radius for each individual point; the naïve approach could not fulfill such requirements since we would have to maintain a separate histogram for each possible value of the uncertainty radius. On the other hand, our proposal can absorb these necessities and handle an unrestrained number of different radii without increasing the memory space requirement of the constructed histogram, producing at the same time a very good estimation.

4 Experimental Study

In this section, we present several experiments using synthetic and real spatial data in order to demonstrate the correctness and accuracy of our analysis in the various settings examined, as well as the efficiency of the proposed solutions. Concisely, in the experimental study that follows, we

- demonstrate the accuracy of the analytical model ((1) and (13)), as well as its sensitivity with respect to the involved parameters, i.e., the uncertainty radius or standard deviation, and the length of the query perimeter,
- show the quality of the approximation of normally distributed location uncertainty by *2d-UDD* utilizing the model supported by (25) and (30),
- present the accuracy of the estimation provided by our analytical models—(32), (33), (36), and (37)—over real spatial data utilizing histograms and also demonstrate their advantage to the alternative of utilizing the histogram as a local density estimator using (31),
- show how our proposal can be used in the context of SDWs, and

- reveal the efficiency of the provided solutions implemented on top of a commercial SDBMS.

4.1 Experimental Setup

Our experimental study is based on both synthetic and real point data sets. In particular, we used a synthetic data set (Rnd_0) of 100,000 2D points randomly distributed in the unit data space, as well as two real data sets, namely, the North East (*NE*) and the Digital Chart of the World (*DCW*) data sets, both obtained from [23].

Then, as suggested in [4], [7], and [10], we added noise in each data set point in a controlled way. In particular, we modified the location of each point in all three data sets by adding noise, either uniformly distributed inside an uncertainty disk of radius d, producing the respective U-d data set, or following a bivariate normal distribution with standard deviation σ, producing the respective N-σ data set; for each U-d and N-σ data set, we produced five different data sets, that is, $Rnd_{U\text{-}d\text{-}1}$ to $Rnd_{U\text{-}d\text{-}5}$, $NE_{U\text{-}d\text{-}1}$ to $NE_{U\text{-}d\text{-}5}$, and $DCW_{U\text{-}d\text{-}1}$ to $DCW_{U\text{-}d\text{-}5}$, and also the same five data sets for each one of the $Rnd_{N\text{-}\sigma}$, $NE_{N\text{-}\sigma}$, $DCW_{N\text{-}\sigma}$ cases. In order also to test the accuracy of our estimations under the settings in Section 3.3, we produced the $NE_{N\text{-}v0.02}$ data set on which we have added noise following the bivariate normal distribution with σ varying between 0 and 0.02. Unless otherwise indicated, all experimentations involving spatial queries were performed by running 1,000 randomly distributed square, i.e., with $a = b$, queries over all five data sets of the respective case; elongated query windows reported similar behavior. We conducted our experiments on a Windows XP workstation with AMD Athlon 64 3-GHz processor CPU, 1 Gbyte of main memory, and several gigabytes of disk space. All evaluated methods were implemented on both VB.NET and PostgreSQL [19] with the PostGIS [18] extension.

4.2 Experiments on the Quality

Two statistical measures were used so as to demonstrate the behavior of our model: the *average number of false negatives and false positives*, $\overline{E_N}$ and $\overline{E_P}$, respectively, and the *average error in the estimation of false negatives and false positives in each individual query*, $\overline{ES_N}$ and $\overline{ES_P}$, respectively. Formally, these measures are defined as

$$\overline{E_N} = \frac{1}{n}\sum_{i=1..n} E_{N,i}, \overline{E_P} = \frac{1}{n}\sum_{i=1..n} E_{P,i} \tag{38}$$

and

$$\begin{aligned}\overline{ES_N} &= \frac{1}{n}\sum_{i=1..n} \left|E_{N,i} - E_N(R_{a\times b})\right|, \overline{ES_P} \\ &= \frac{1}{n}\sum_{i=1..n} \left|E_{P,i} - E_P(R_{a\times b})\right|,\end{aligned} \tag{39}$$

where n is the number of executed queries, and $E_{P,i}$ ($E_{N,i}$) is the actual number of false positives (false negatives, respectively) in the ith query. We distinguish between, e.g., $\overline{E_P}$ and $\overline{ES_P}$ in order to uncover the details of the behavior of our model, as will be shown in the following experiments.

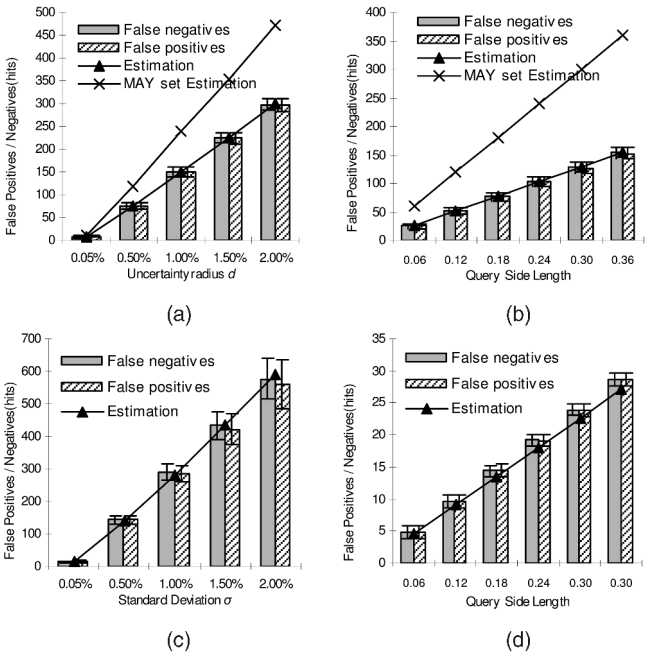


Fig. 9. Average false negatives/positives and their estimations scaling with (a) d and (b) the query size (synthetic data—uniform distribution of uncertainty) and with (c) σ and (d) the query size (synthetic data—normal distribution of uncertainty).

4.2.1 Experiments over Synthetic Data Following All Three Original Assumptions A_I, A_{II}, and A_{III}

In the first series of experiments, we utilize the synthetic data sets in order to demonstrate the accuracy and the behavior of our analytical model scaling the two influencing factors: the radius d of the uncertainty disk and the size (a, b) of the query window. Note that in all figures, the query size is exposed in terms of its side length $2a = 2b$, e.g., for query side length 0.30, the size of the query window is equal to $0.30 \times 0.30 = 0.09$ of the unit space.

In particular, in our first experiment, we scaled the value of d between 0.05 percent and 2 percent of the space extent along the x- and y-axes, querying both Rnd_0 and the respective $Rnd_{u\text{-}d}$ data set, with fixed side length 0.18 (i.e., $a = b = 0.09$, resulting in a query window sized 3.24 percent of the data space). The results of this experiment are illustrated in Fig. 9a; as a first result, the number of false positives and false negatives turn out to be almost equal, verifying the correctness of the corollary in (22). Moreover, the estimations $E_P(R_{a\times b})$ and $E_N(R_{a\times b})$ are extremely accurate with respect to $\overline{E_P}$ and$\overline{E_N}$, with the error being always below 3 percent, whereas the error bars in each graph column, illustrating $\overline{ES_P}$ and $\overline{ES_N}$, are shown to be relatively low. Specifically, the average error in individual queries is below 10 percent in the vast majority of the experimental settings and is up to 29 percent in a single extreme case where the uncertainty radius d is set to its minimum ($d = 0.05$ percent).

We have also included in our study the methodology provided in [27], which estimates the cardinality of the *MAY* set. As already stated, the *MAY* set is actually a superset containing, among others, the false hits calculated by our analysis; nevertheless, we evaluate the assumption

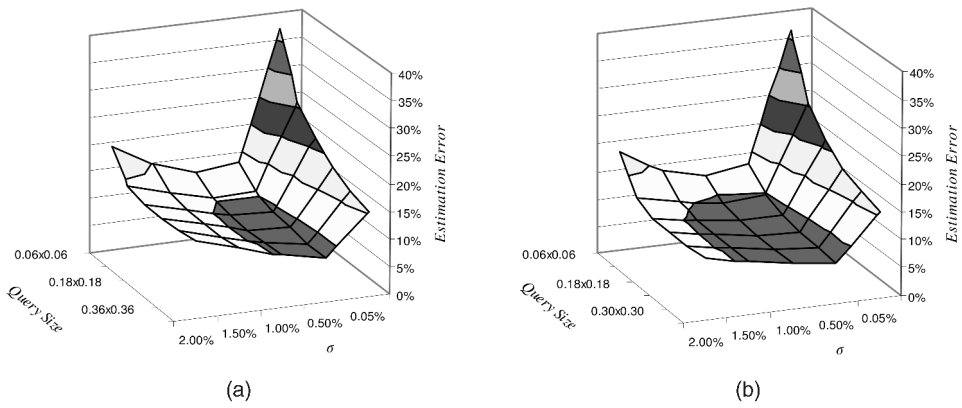


Fig. 10. Average estimation error of (a) false positives $\overline{ES}_P$ and (b) false negatives $\overline{ES}_N$ in each query, scaling with d and the query size (synthetic data—normal distribution of uncertainty).

that 50 percent of the *MAY* set are false hits, that is, an object in the *MAY* set may be either a true or false hit with the same probability. However, as illustrated in Fig. 9, in the *MAY set estimation* curve, the above assumption does not result in correct estimations. It is worth to note, however, that the goal of the analysis presented in [27] is not to provide the number of false hits the way our analysis does. Our assumption regarding the portion of the *MAY* set encountering false hits, i.e., the 50 percent, is used due to the lack of any other suggestions on this subject included in [27]. Moreover, Fig. 9a could also lead to the presumption that a simple multiplier on the *MAY* set estimation, i.e., lowering the corresponding curve in Fig. 9, could force it to produce better results. Still, in order to determine this multiplier, it is the methodology provided in our analysis that should be followed.

Similar results are exposed in the second experiment, illustrated in Fig. 9b, where we scale the query size. In particular, we set the uncertainty radius to 0.5 percent and scaled the length of the query side between 0.06 and 0.36, resulting in query sizes covering between 0.36 percent and 12.96 percent of the data space. When comparing the estimation of the number of false negatives and false positives and the respective average values $\overline{E}_P$ and $\overline{E}_N$, the reported estimation error is below 1 percent, regardless of the query size. Furthermore, the estimation based on the *MAY* set cardinality once again could not yield comparable results; as such, based on the observation that this estimation systematically overestimates $\overline{E}_P$ and $\overline{E}_N$, we will exclude it from the rest of the experimental study. Regarding the error bars in each graph column, illustrating the respective $\overline{ES}_P$ and $\overline{ES}_N$, they are relatively small in the majority of the experiments being below 16 percent; the only case where it reached higher values, i.e., 35 percent, occurred when both σ and the query size were set to their minimum values.

4.2.2 Experiments over Synthetic Data Relaxing Assumption A_I

In order to evaluate the accuracy of the estimation of the number of false positives and false negatives $E_P(R_{a\times b}) = E_N(R_{a\times b})$ calculated by (25) and (30), we performed a similar experimentation with the $Rnd_{N\text{-}\sigma}$ data sets where we scaled σ and the query size. The results of these experiments are illustrated in Figs. 9c and 9d, and it is clear that the estimation error regarding $\overline{E}_P$ and $\overline{E}_N$ is always below 5 percent. Moreover, the respective error bars, illustrating $\overline{ES}_P$ and $\overline{ES}_N$, are shown to be relatively small, being usually below 12 percent, while reaching 36 percent only in the case where both d and the length of the query side were set to their minimum values.

A more detailed presentation of the average estimation error in each individual query $\overline{ES}_P$ and $\overline{ES}_N$ is illustrated in Figs. 10a and 10b, as a percentage of the number of false positives and false negatives, respectively. Both figures illustrate that $\overline{ES}_P$ and $\overline{ES}_N$ vary from small values, i.e., less than 10 percent for high values of σ, to higher ones for very small values of σ. They also depend on the query size, increasing as the size decreases. In general, it appears that $\overline{ES}_P$ and $\overline{ES}_N$ are essentially ruled by the standard deviation σ and, at a smaller extent, on the query size. Furthermore, for small values of σ and small query sizes, while the estimation is still accurate regarding $\overline{E}_P$ and $\overline{E}_R$ (Figs. 9a and 9b, respectively), $\overline{ES}_P$ and $\overline{ES}_N$ increase significantly up to 40 percent.

4.2.3 Experiments over Real Data Relaxing Assumption A_{II}

In order to support real arbitrarily distributed data by employing histograms, we utilized the NE data set along with the respective $NE_{N\text{-}\sigma}$ data sets. Subsequently, we created the *MinSkew* partitioning of each modified data set using a uniform grid of original grid size set to 0.001×0.001, as discussed in [1]. The experiments over the $NE_{U\text{-}d}$ data sets, i.e., with uniform uncertainty distribution, reported similar behavior and thus are omitted. In particular, in order to evaluate the accuracy of the analysis in Section 3.2, i.e., the estimation of $E_P(R_{a\times b})$ and $E_N(R_{a\times b})$ using (33), we experimented with the NE and $NE_{N\text{-}\sigma}$ data sets, first scaling σ with the query size fixed to 0.18×0.18 and then scaling the query size with σ fixed to 0.5 percent.

Fig. 11 illustrates the actual and estimated values of false negatives and false positives using the above experimental settings. Clearly, the estimations are accurate, with the reported error being always lower than 6 percent. Additionally, the average estimation error in each individual

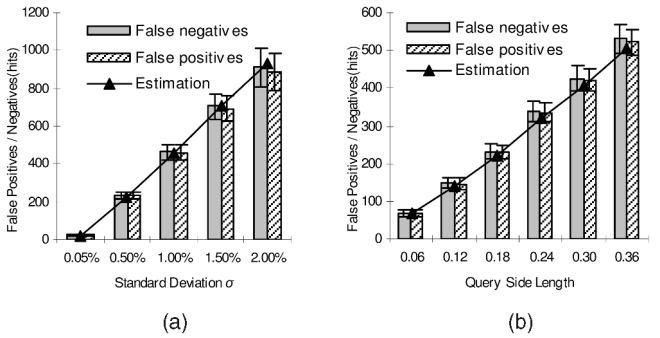


Fig. 11. Average false negatives/positives and their estimations scaling with (a) σ and (b) the query size (real data—bivariate normal distribution of uncertainty).

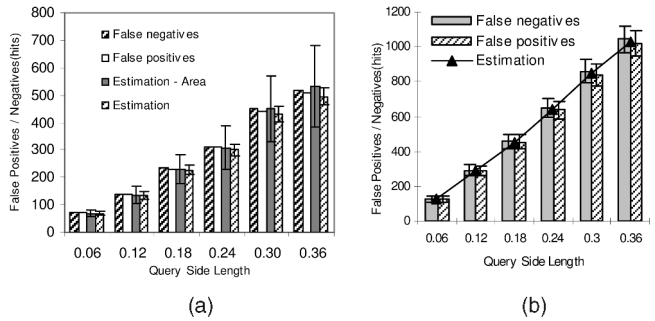


Fig. 13. (a) Average false negatives/positives and estimation error in each individual query using different model approaches (real data—normal distribution of uncertainty). (b) Average false negatives/positives and their estimations scaling with the query size (real data—bivariate normal distribution of uncertainty).

query $\overline{ES}_P$ and $\overline{ES}_N$, which is illustrated in the error bars in Fig. 11 and, in more detail, in Figs. 12a and 12b, respectively, is considerably small, being below 12 percent in the majority of the experimental settings. It is also clear that as the query size increases, $\overline{ES}_P$ and $\overline{ES}_N$ decrease to values lower than 11 percent. On the other hand, small query sizes lead to increased $\overline{ES}_P$ and $\overline{ES}_N$ values, between 12 percent and 24 percent regarding query sizes of 0.06×0.06, nevertheless with a smaller error peak than the ones reported for random data without the usage of histograms, e.g., the reported 36 percent in Fig. 10 versus 24 percent in Fig. 12. The above observation can be explained by the fact that histograms provide a locally more accurate value of the estimated error than the global formula does over synthetic data, since they help the model absorb the local density changes of real arbitrarily distributed spatial data.

The impact of our analysis in real data sets with the aid of histograms is demonstrated by performing a set of experiments over the NE and $NE_{N\text{-}\sigma}$ data sets, computing our model by two different approaches: 1) producing the local density via (31) and then using it in (1) and (13) and 2) directly utilizing (33). In our experiment, we set $\sigma = 0.5$ percent and scaled the side of the query window from 0.06 to 0.36. The results of this experiment are illustrated in Fig. 13a, which demonstrates that although the first approach, labeled as *Estimation-Area* in Fig. 13, provides an accurate average estimation, the obtained values for $\overline{ES}_P$ and $\overline{ES}_N$ are higher than those produced by the second approach, labeled as *Estimation* in Fig. 13a. This confirms that the appropriate use of histograms in our model is according to the analysis in Section 3.2 by directly employing (33).

4.2.4 Experiments over Real Data Relaxing Assumption A_{III}

In order to demonstrate the high-quality estimations obtained when using the augmented histogram methodology in Section 3.3, we performed an experiment by employing the NE and $NE_{N\text{-}\upsilon 0.02}$ data sets; as already mentioned, $NE_{N\text{-}\upsilon 0.02}$ contain data with *variable* known size of standard deviation σ, varying between 0 and 0.02. We then scaled the side of the query window from 0.06 to 0.36. The respective results, illustrated in Fig. 13b, show that there is no significant difference between this case and the one where σ was set to a constant value (Fig. 11b) and the estimations of $\overline{E}_P$ and $\overline{E}_N$ are again very accurate. Moreover, the obtained values for $\overline{ES}_P$ and $\overline{ES}_N$, i.e., the error bars, vary between 7 percent and 14 percent, while in the case of Fig. 11b, the respective error varied between 6 percent and 13 percent. We can therefore conclude that the analysis in Section 3.3 regarding variable uncertainty radii is verified to be at least as accurate as the respective analysis in Section 3.2, which assumes a constant uncertainty radius.

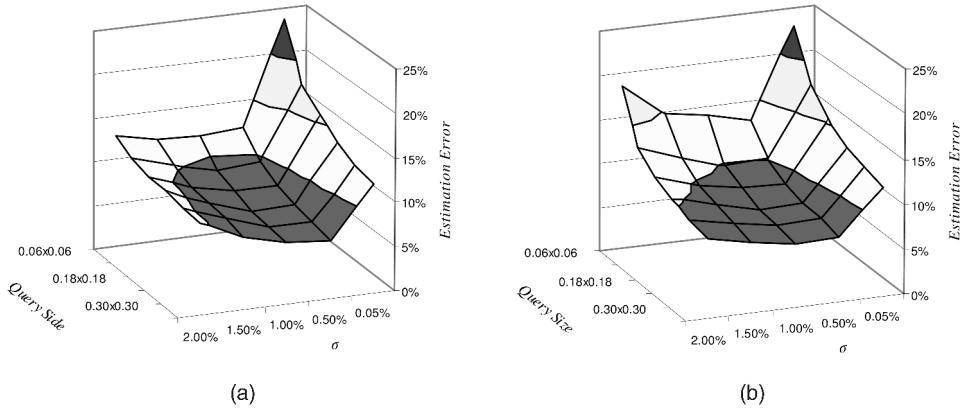


Fig. 12. Average estimation error of (a) false positives $\overline{ES}_P$ and (b) false negatives $\overline{ES}_N$ in each query, scaling with σ and the query size (real data—bivariate normal distribution of uncertainty).

Fig. 14. (a) Error between the actual number of false hits and their estimation in the roll-up operation from the cell to state level in the USA map. (b) A bad approximation of a state by its MBR.

TABLE 2
Histogram Statistics

	Dataset	*# Objects*	*grid size*	*# grid cells*	*# buckets*	*Construction Execution time (sec)*
Histogram	$NE_{N\text{-}0.01\text{-}1}$	123K	0.001×0.001	920K	1K	21
Augmented Histogram	$NE_{N\text{-}v0.02}$	123K	0.005×0.005 ×0.0001	7078K	1K	29

4.2.5 Experiments over Real Data Warehouses

In order to demonstrate the application of our model in a data cube operation, we used the DCW and $DCW_{N\text{-}0,5}$ data sets, where the added Gaussian noise in the location of each point has σ equal to 0.5 percent of the space extent along the x-axis, since the size of the space is different along the x- and y-axes. Then, we applied a uniform 60 × 30 grid along the x- and y-axes, forming 1,800 buckets overlaying the USA map, and counted the number of objects contained inside each cell. We subsequently performed a *roll-up* operation at the *state* level, as discussed in Section 1. In particular, we calculated the estimation of false positives and false negatives by the MBRs of US states as range queries, as discussed in Section 3.2. Finally, we used the original data sets in order to determine the actual number of false positives and false negatives.

The error between the estimated and the actual number of false hits obtained as the sum of false negatives and false positives is illustrated in Fig. 14a. Clearly, the error in the majority of the US States is below 25 percent, while the actual weighted average is 16 percent. Regarding the four outliers, labeled with the name of the state in Fig. 14a, the high error presented is due to either the tiny size of the query polygon, i.e., the Delaware case, verifying the result of a previous experiment that the error increases as the query size decreases, or the irregular shape of the query polygon that is not well approximated by its MBR, i.e., the California, Florida, and Michigan cases, with their shapes illustrated in Fig. 14b.

4.3 Experiments on the Efficiency

We also experimented with the performance of the proposed solutions using an implementation of our model in the PostgreSQL [19] DBMS along with the PostGIS [18] spatial extension. Since the selected DBMS does not natively support MinSkew [1] spatial histograms, we have extended it towards this direction; moreover, we have included in our implementation the augmented histogram proposed in Section 3.3. All methods were implemented as functions of the spatial DBMS in the PL/pgSQL language; the developed software is ported in a template database.

In the first experiment, we utilized the $NE_{N\text{-}0.01\text{-}1}$ and $NE_{N\text{-}v0.02}$ data sets and counted the time required to construct the MinSkew and the augmented MinSkew histograms, respectively; the results are shown in Table 2. Clearly, the processing time is reasonable given the fact that this is an offline operation, executed only once; then, the constructed histogram buckets are permanently stored in a relational table. Here, it is worth to note that since the MinSkew construction algorithm initially overlays a regular grid on top of the data set, being subsequently used instead of the original data set, the time required for constructing a MinSkew histogram does not depend on the data set size; this is also confirmed in the respective experimental study in [1]. Therefore, the execution times illustrated in Table 2 can be considered as representatives, given also the other histogram parameters, i.e., the number of buckets and the number of the overlaid grid cells.

In our second experiment, we employed the $NE_{N\text{-}0.01\text{-}1}$ data set and 1,000 randomly distributed rectangular queries in order to evaluate the average execution time of the function that implements our model; we also scaled the query size in a way similar to that in Section 4.2 from 0.06 × 0.06 to 0.36 × 0.36. The respective results showed that regardless of the query size, the execution time required by the DBMS to estimate the false hits introduced in a query was approximately 16 ms, while the time required to process the actual query was 120 ms. Clearly, our proposal can be employed as an estimator, since its execution time is restrained to a few milliseconds, while the actual query execution typically needs one order of magnitude more time. Moreover, it is revealed that the expected result that

the overhead introduced by the estimator is independent from the query size.

4.4 Summary of the Experimental Results

Summarizing the results of our experimental study, the model proposed in Sections 2 and 3 is shown to provide high accuracy with an average error on $\overline{E}_P$ and $\overline{E}_N$ never exceeding 6 percent for either random synthetic or real data. Regarding the uniform case, the estimation of the number of false hits is accurate regardless of the value of the query size and the radius d of the uncertainty disk or σ in the case of data with normally distributed uncertainty. Moreover, it has been shown that simple modifications in the single work that is very related to ours [27] could not yield to an accurate estimation of the average number of false hits. The experiments over real data demonstrate accuracy even higher than the one reported for synthetic data, with very low $\overline{ES}_P$ and $\overline{ES}_N$ errors, indicating the advantage introduced by the employment of histograms, even in the case of variable σ. Furthermore, it is verified that in the presence of histograms, it is much more appropriate to use the model expressed by (32) and (33) than using the local density estimated by the histogram via traditional operations, i.e., via (31). The results on the application of our model in spatial data cubes and spatial OLAP operations are also very promising. Finally, the implementation of the proposed solutions in real-world environments has shown the efficiency of our proposal when employed as an estimator, since its execution time is typically only a few milliseconds.

5 Related Work

Wolfson et al. [29] address the imprecision problem of the location of moving objects by proposing a set of updating policies of the database that stores the object locations. The basic idea is that the database is updated whenever the distance between the actual location of an object and that stored in the database value exceeds a threshold. In this way, an uncertainty factor of every object's location is introduced, since objects are within distance of 1 km from the last recorded locations. Adopting the utilization of *pdf*s, they describe an algorithm that processes a probabilistic spatial range query applied in the above database. The output of this type of query, which returns the set of objects being within a region R, consists of pairs of the form (O_i, P_i), where P_i is the probability that object O_i intersects query region R. Cheng et al. [6] adopt the definition of the probabilistic query introduced in [29] and extend it in the case of nearest neighbor (NN) queries. Under the setting set in [29] and the open agora scenario discussed in [12], our work may be used as a client-side optimizer of rectangular time-slice queries executed over moving object databases.

Location uncertainty of moving objects is also discussed by Trajcevski et al. [25], [26], where a trajectory of an object is modeled as a 3D cylindrical volume around the tracked trajectory. Furthermore, two categories of operators for querying trajectories with uncertainty are introduced, concerning spatiotemporal point and range queries, respectively, and efficient algorithms are presented for their implementation. Reference [26] also discusses the *may* versus *must* in terms of the meaning of their proposed operators; as such, they distinguish between the *sometime* and *always* in the temporal domain and the *possibly* and *definitely* operators in the spatial domain, and they provide a set of spatiotemporal query types under uncertainty. The work in [26] gave us the intuition to use the *uncertainty disk*, since they also model each spatiotemporal trajectory as a cylindrical volume of constant radius; moreover, the extension of our model in the spatiotemporal domain can be justified when employing the way trajectories are modeled in [26].

Cheng et al. [8] investigated the problem of indexing uncertain data in order to efficiently answer probabilistic threshold queries, in which the appearance probability of each data point in the result of the query exceeds a given threshold. Two index structures are proposed. The pruning power of the first index is based on the utilization of uncertain information augmented to the internal nodes of the index, while in the second index, data points with similar degrees of uncertainty are clustered together. Recently, Tao et al. [21] studied a similar type of query, the probabilistic range query, which retrieves the objects that appear in a rectangular area with probabilities of at least a predefined value. Based on the notion of probabilistic constrained rectangle (PCR), they introduced a fully dynamic index structure on uncertain data. This structure, called U-tree, maintains "auxiliary information" at all of its levels for the respective indexed objects that can be used to validate the presence of an object in the results of a probabilistic range query, without calculating its computationally expensive appearance probability. Our work can be considered as complementary to the work in [21] when employed in the context of [12]; in particular, our work can be used as a client-side optimizer, which optimizes the query in terms of quality of output, while the U-tree proposed in [21] could be employed at the server-side for performing the final user request.

Ni et al. [16] propose a probabilistic spatial data model for the positional accuracy of polygon data. According to this model, each polygon is partitioned into disjoint independent chunks. Each chunk is a contiguous series of vertices with fully correlated locational uncertainties. Based on the above model, a probabilistic spatial join algorithm is described, in which the object pairs of the result are associated with the intersection probability between each pair. A variation of the R-tree, called probabilistic R-tree, is introduced for the support of the probabilistic filtering of the join algorithm, in which each MBR approximation is augmented with the probability distribution of MBR's boundary. The extension of our work to support nonpoint data sets could also enable it to be made complementary to the work in [16] under the open agora scenario [12].

Dai et al. [9] have studied the problem of evaluating spatial queries for existentially uncertain data; in this case, uncertainty does not refer to the locations of the objects but to their existence. The authors define two probabilistic query types: the so-called thresholding and ranking queries in which the output is controlled by thresholding the results of low probability to occur or ranking them and selecting the ones with the highest probability, respectively. In the sequel,

probabilistic variants of spatial range and NN queries are presented for objects indexed by a 2D index such as the R-tree. Finally, in order to improve the efficiency of their proposed algorithms, they propose an extension of the R-tree, in which the nonleaf entries are augmented with the maximum existential probability of the objects indexed under them. Existentially uncertain data are fundamentally different from the locationally uncertain data taken into account in our work; as such, the estimation of the number of false hits in the context of probabilistic ranking and thresholding queries over existentially uncertain data can be considered as a future extension of our work.

Perhaps the most relevant to our work is the study by Yu and Mehrotra [27], where the effect of uncertainty in probabilistic spatial queries, similar to the work presented in [16], is discussed. By performing a theoretical analysis, they provide a novel technique that can be used in order to set the data precision in the data collection process, so that a probabilistic guarantee on the uncertainty in answers of spatial queries can be provided. The first outcome of the analysis are the cardinalities of the three subsets of a range query result, namely, the *MUST*, *MAY*, and *ANS* sets: *MUST* is the set of objects that "must" be located within the query range, *MAY* is the set of objects that "may" be located within the query range, and *ANS* is the approximate answer set of objects whose recorded locations are in the query region. The second outcome is a method for determining the largest possible imprecision, i.e., the *uncertainty radius* of our analysis, given that the answer to a random *COUNT* query should include an uncertainty $\delta \leq \delta_0$, i.e., the cardinality of the *MAY* set should be less than a given value, with a probability $P \geq P_0$.

Comparing our model with [27], the first remark is that the numbers E_N and E_P of false hits that we estimate are actually a *refinement*, i.e., a subset, of the *MAY* set estimated in [27], and it is not straightforward to remove the overestimation provided in [27] unless our model is used; this overestimation was clearly shown in the experimental results presented in Section 4.2.1. A second remark is that the model presented in [27] is based on the uniformity assumption, whereas our study addresses more realistic requirements.

6 Conclusions and Future Work

In this work, we presented a theoretical model that estimates the error introduced by each object's location uncertainty in the results of rectangular range queries over spatial point data. We provided a closed formula of the average number of false positives and false negatives, under three assumptions: uniform location uncertainty, uniformly distributed data, and constant radius of uncertainty disk. Then, we relaxed these assumptions towards more realistic settings, using the bivariate normal distribution over location uncertainty and *MinSkew* histograms for data and radius distributions. The accuracy of our model was evaluated through extensive experimentation using various synthetic and real data sets.

The applications of our proposal include query optimization under the open agora scenario [12], interactive database querying, imprecision settings, and data warehouse operations, as extensively discussed. The proposed model can be directly employed in spatial database systems in order to provide users with the accuracy of spatial query results based only on known data set and query features, while off-the-self histograms already employed in spatial databases for query optimization purposes can serve our model without the need for any additional adjustments.

There are numerous interesting research directions arising from this work, including the application of our model in data spaces of higher dimensionality and its extension in order to support nonpoint data sets and nonrectangular query windows, as well as NN queries.

Acknowledgments

This research was partially supported by FP6/IST Programme of the European Union under the GeoPKDD Project (2005-2008) (http://www.geopkdd.eu). The authors would like to thank the anonymous reviewers for their insightful comments.

References

[1] S. Acharya, V. Poosala, and S. Ramaswamy, "Selectivity Estimation in Spatial Databases," *Proc. ACM SIGMOD '99,* pp. 13-24, 1999.

[2] P.K. Agarwal, E. Flato, and D. Halperin, "Polygon Decomposition for Efficient Construction of Minkowski Sums," *Computational Geometry,* vol. 21, nos. 1/2, pp. 39-61, 2002.

[3] G. Barequet, A.J. Briggs, M.T. Dickerson, and M.T. Goodrich, "Offset-Polygon Annulus Placement Problems," *Computational Geometry: Theory and Applications,* vol. 11, nos. 3/4, pp. 125-141, 1998.

[4] A.R. Beresford and F. Stajano, "Location Privacy in Pervasive Computing," *IEEE Pervasive Computing,* vol. 2, no. 1, 2003.

[5] J. Chen and R. Cheng, "Efficient Evaluation of Imprecise Location-Dependent Queries," *Proc. 23rd Int'l Conf. Data Eng. (ICDE '07),* pp. 586-595, 2007.

[6] R. Cheng, D. Kalashnikov, and S. Prabhakar, "Querying Imprecise Data in Moving Object Environments," *IEEE Trans. Knowledge and Data Eng.,* vol. 16, no. 9, pp. 1112-1127, Sept. 2004.

[7] R. Cheng, Y. Zhang, E. Bertino, and S. Prabhakar, "Preserving User Location Privacy in Mobile Data Management Infrastructures," *Proc. Sixth Workshop Privacy Enhancing Technologies (PET '06),* June 2006.

[8] R. Cheng, Y. Xia, S. Prabhakar, R. Shah, and J.S. Vitter, "Efficient Indexing Methods for Probabilistic Threshold Queries over Uncertain Data," *Proc. 30th Int'l Conf. Very Large Data Bases (VLDB '04),* pp. 876-887, 2004.

[9] X. Dai, M.L. Yiu, N. Mamoulis, Y. Tao, and M. Vaitis, "Probabilistic Spatial Queries on Existentially Uncertain Data," *Proc. Ninth Int'l Symp. Spatial and Temporal Databases (SSTD '05),* pp. 254-272, 2005.

[10] B. Gedik and L. Liu, "A Customizable *k*-Anonymity Model for Protecting Location Privacy," *Proc. 25th Int'l Conf. Distributed Computing Systems (ICDCS),* 2005.

[11] Y. Ioannidis, "Universality of Serial Histograms," *Proc. 19th Int'l Conf. Very Large Data Bases (VLDB '93),* pp. 256-267, 1993.

[12] Y. Ioannidis, "Emerging Open Agoras of Data and Information," *Proc. 23rd Int'l Conf. Data Eng. (ICDE '07),* pp. 1-5, 2007.

[13] Y. Ioannidis and V. Poosala, "Balancing Histogram Optimality and Practicality for Query Result Size Estimation," *Proc. ACM SIGMOD '95,* pp. 233-244, 1995.

[14] C.S. Jensen, A. Kligys, T.B. Pedersen, C.E. Dyreson, and I. Timko, "Multidimensional Data Modeling for Location-Based Services," *VLDB J.,* vol. 13, no. 1, pp. 1-21, 2004.

[15] A. Leick, *GPS Satellite Surveying*. John Wiley & Sons, 1995.

[16] J. Ni, C.V. Ravishankar, and B. Bhanu, "Probabilistic Spatial Database Operations," *Proc. Eighth Int'l Symp. Spatial and Temporal Databases (SSTD '03),* pp. 140-158, 2003.

[17] D. Pfoser, N. Tryfona, and C.S. Jensen, "Indeterminacy and Spatiotemporal Data: Basic Definitions and Case Study," *GeoInformatica,* vol. 9, no. 3, pp. 211-236, 2005.
[18] *PostGIS,* http://postgis.refractions.net, (accessed November 2007).
[19] *PostgreSQL,* http://www.postgresql.org, (accessed November 2007).
[20] P. Rigaux, M. Scholl, and A. Voisard, *Spatial Databases with Application to GIS.* Morgan Kaufmann, 2002.
[21] Y. Tao, X. Xiao, and R. Cheng, "Range Search on Multi-Dimensional Uncertain Data," *ACM Trans. Database Systems,* vol. 32, no. 3, 2007.
[22] Y. Tao, J. Zhang, D. Papadias, and N. Mamoulis, "An Efficient Cost Model for Optimization of Nearest Neighbor Search in Low and Medium Dimensional Spaces," *IEEE Trans. Knowledge and Data Eng.,* vol. 16, no. 10, pp. 1169-1184, Oct. 2004.
[23] *The R-Tree Portal,* Y. Theodoridis, ed., http://www.rtreeportal.org, (accessed November 2007).
[24] Y. Theodoridis and T. Sellis, "A Model for the Prediction of *R*-Tree Performance," *Proc. 15th ACM Symp. Principles of Database Systems (PODS '96),* pp. 161-171, 1996.
[25] G. Trajcevski, "Probabilistic Range Queries in Moving Objects Databases with Uncertainty," *Proc. Third ACM Int'l Workshop Data Eng. for Wireless and Mobile Access (MobiDE '03),* pp. 39-45, 2003.
[26] G. Trajcevski, O. Wolfson, K. Hinrichs, and S. Chamberlain, "Managing Uncertainty in Moving Objects Databases," *ACM Trans. Database Systems,* vol. 29, no. 3, pp. 463-507, 2004.
[27] X. Yu and S. Mehrotra, "Capturing Uncertainty in Spatial Queries over Imprecise Data," *Proc. 14th Int'l Conf. Database and Expert Systems Applications (DEXA '03),* pp. 192-201, 2003.
[28] Wolfram Research, *Mathematica Version 5.2,* http://www.wolfram.com/, accessed, November 2007, 2005.
[29] O. Wolfson, P.A. Sistla, S. Chamberlain, and Y. Yesha, "Updating and Querying Databases that Track Mobile Units," *Distributed and Parallel Databases,* vol. 7, no. 3, pp. 257-387, 1999.

Elias Frentzos received the diploma in civil engineering and the MSc degree in geoinformatics from the National Technical University of Athens, Greece, in 1997 and 2002, respectively. He is currently a PhD student in the Information Systems Laboratory, Department of Informatics, University of Piraeus (UniPi). His research interests include spatial and spatiotemporal databases, location-based services, and geographical information systems.

Kostas Gratsiasreceived the diploma and MSc degree in computer engineering and informatics from the Computer Engineering and Informatics Department, University of Patras, in 2001 and 2003, respectively. He is currently a PhD student in the Information Systems Laboratory, Department of Informatics, University of Piraeus (UniPi). His research interests include spatiotemporal databases, geographical information systems (GISs), and data mining.

Yannis Theodoridis received the diploma and PhD degree in electrical and computer engineering from the National Technical University of Athens, Greece, in 1990 and 1996, respectively. He is an associate professor with the Information Systems Laboratory, Department of Informatics, University of Piraeus (UniPi). Currently, he is the scientist in charge for UniPi in the EC-funded GeoPKDD project (2005-2008) on geographic privacy-aware knowledge discovery and delivery, being also involved in several national-level projects. His research interests include spatial and spatiotemporal databases, geographical information management, knowledge discovery, and data mining. He has coauthored three monographs and more than 50 publications in scientific journals (including *Algorithmica*, *ACM Multimedia*, and the *IEEE Transactions on Knowledge and Data Engineering*) and conferences (including ACM Sigmod, PODS, VLDB, and ICDE) with more than 400 citations for his work. He serves in the program committee for several major conferences in databases and data mining and in the editorial board of the *International Journal on Data Warehousing and Mining*. He is member of the ACM and the IEEE.

APPENDIX: Proof of Lemma 4

According to the Least Squares Theory, the best approximation of a function f by another function g in the same domain D is given by minimizing the integral $\iint_D \left(f(x)-g(x)\right)^2 dx$ of the square of their difference along D. Subsequently, in order to prove Lemma 4 we have to determine the value of d that minimizes $\iint_{\mathbb{R}^2}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy$. Towards this goal, it holds that:

$$\begin{aligned}&\iint_{\mathbb{R}^2}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy=\\&=\iint_{C(0,d)}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy+\iint_{\mathbb{R}^2-C(0,d)}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy\end{aligned} \tag{40}$$

where $C(0,d)$ is the disk with center (0,0) and radius d. Applying Eq.(24) and Eq.(23) into Eq.(40), we get:

$$\begin{aligned}&\iint_{\mathbb{R}^2}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy=\\&\iint_{C(0,d)}\left(\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}-\frac{3}{\pi d^2}\left(1-\frac{\sqrt{x^2+y^2}}{d}\right)\right)^2 dxdy+\iint_{\mathbb{R}^2-C(0,d)}\left(\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}-0\right)^2 dxdy\end{aligned} \tag{41}$$

At this point, we utilize the Cartesian-to-Polar transformation, which transforms (x, y) to (ρ, θ) according to the following formula:

$$\iint f(x,y)dxdy=\iint f(\rho\cos\theta,\rho\sin\theta)\rho d\rho d\theta \tag{42}$$

Applying the above transformation to Eq.(41), we get:

$$\begin{aligned}&\iint_{\mathbb{R}^2}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy=\\&\int_0^{2\pi}\int_0^d\left(\frac{1}{2\pi\sigma^2}e^{-\frac{\rho^2}{2\sigma^2}}-\frac{3}{\pi d^2}\left(1-\frac{\rho}{d}\right)\right)^2\rho\, d\rho d\theta+\int_0^{2\pi}\int_d^{\infty}\left(\frac{\rho}{2\pi\sigma^2}e^{-\frac{\rho^2}{2\sigma^2}}\right)^2\rho d\rho d\theta\end{aligned}$$

This results in

$$\iint_{\mathbb{R}^2}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy=\frac{d^3-18d\sigma^2+12\sqrt{2\pi}\sigma^3\,\mathrm{Erf}\left[\frac{d}{\sqrt{2}\sigma}\right]}{4d^3\pi\sigma^2} \tag{43}$$

where Erf[x] is the error function encountered in integrating the normal distribution. In the sequel, we calculate the first derivative of Eq.(43) with respect to d:

$$\frac{\partial\iint_{\mathbb{R}^2}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy}{\partial d}=-\frac{9d+6de^{-\frac{d^2}{2\sigma^2}}-9\sqrt{2\pi}\,\mathrm{Erf}\left[\frac{d}{\sqrt{2}\sigma}\right]}{d^4\pi} \tag{44}$$

and by substituting d/σ with a variable a ($a\neq 0$), we result in the following expression:

$$\frac{\partial\iint_{\mathbb{R}^2}\left(P_{2d-UDD}(x,y)-P_{BN}(x,y)\right)^2 dxdy}{\partial d}=\frac{9a+6ae^{-\frac{a^2}{2}}-9\sqrt{2\pi}\,\mathrm{Erf}\left[\frac{a}{\sqrt{2}}\right]}{ad^3\pi} \tag{45}$$

which is zeroed when the numerator becomes zero. Hence, the first derivative of Eq.(43) is zeroed

when

$$9a + 6ae^{-\frac{a^2}{2}} - 9\sqrt{2\pi}\,\mathrm{Erf}\left[\frac{a}{\sqrt{2}}\right] = 0 \tag{46}$$

After numerically evaluating Eq.(46) it turns out that

$$a \approx 2.36533 \tag{47}$$

Recalling that $a = d/\sigma$ we have proved Lemma 4. ■