The Generic Annular Bucket Histogram for Estimating the Selectivity of Spatial Selection and Spatial Join

CHENG Changxiu, ZHOU Chenghu, CHEN Rongguo

State Key Laboratory of Resources and Environmental Information Systems, Institute of Geographical Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing 100101, China

© Wuhan University and Springer-Verlag Berlin Heidelberg 2011

Abstract Selectivity estimation is crucial for query optimizers choosing an optimal spatial execution plan in a spatial database management system. This paper presents an Annular Bucket spatial histogram (AB histogram) that can estimate the selectivity in finer spatial selection and spatial join operations even when the spatial query has more operators or more joins. The AB histogram is represented as a set of bucket-range, bucket-count value pairs. The bucket-range often covers an annular region like a single-cell-sized photo frame. The bucket-count is the number of objects whose Minimum Bounding Rectangles (MBRs) fall between outer rectangle and inner rectangle of the bucket-range. Assuming that all MBRs in each a bucket distribute evenly, for every bucket, we can obtain serial probabilities that satisfy a certain spatial selection or join conditions from the operations’ semantics and the spatial relations between every bucket-range and query ranges. Thus, according to some probability theories, spatial selection or join selectivity can be estimated by the every bucket-count and its probabilities. This paper also shows a way to generate an updated AB histogram from an original AB histogram and those probabilities. Our tests show that the AB histogram not only supports the selectivity estimation of spatial selection or spatial join with “disjoint”, “intersect”, “within”, “contains”, and “overlap” operators but also provides an approach to generate a reliable updated histogram whose spatial distribution is close to the distribution of actual query result.

Keywords selectivity estimation; AB histogram; annular bucket; spatial selection; spatial join

CLC number P208

Introduction

With the increasing accuracy and volume of spatial data required to be stored and managed, it has become more important to execute spatial data queries efficiently.[1] Many improved methods independent from the internal modules of Database Management System (DBMS) have been explored over the past 10 years. Some examples include compressed storage,[2-3] multi-scale spatial databases,[4-6] and progressive transmission of vector data.[7-9] However, it also is necessary to improve the spatial query optimization technique in the kernel of DBMS, requiring the query optimizer to estimate the selectivity and cost of spatial operations, so that it can choose the query execu-
Selectivity estimation is crucial for a query optimizer to choose a good execution plan in a given spatial query. Selectivity estimation attempts to ascertain how many data items will be retrieved (the selectivity) and what the I/O complexity will be in servicing the query. However, the I/O processing costs are usually more dominant than the CPU processing costs in a query, because I/O processing is a critical bottleneck in the performance of a computer system. Therefore, selectivity is one of the most important parameters in the cost model.

A variety of techniques have been proposed in the literature to estimate the query result size in a relational database. The histogram is the most popular. Common techniques include histograms or those based on parametric techniques that model the data via a standard mathematical distribution. Of the various techniques, histograms, in particular, have proven very popular because they can be computed efficiently, use very little space, and do not require that the input distribution be known in advance. Histogram-based techniques work by partitioning the dataset into a small number of subsets called “buckets” and then using approximations for each bucket to model the distribution of the tuples within. Query result estimations are then obtained by processing the query against the buckets and the approximations used therein.

This paper presents a more generic annular bucket histogram, called the AB histogram, for estimating selectivity for spatial selection and spatial joins with finer spatial operators, even when the spatial query has more operators or more joins. This is the first contribution of the paper. The second contribution is a practical approach to generate an updated AB histogram of query results, which is useful to estimate the selectivity of subsequent spatial query operations.

The rest of this paper is organized as follows. Section 1 reviews related work and presents some problems in the existing spatial histogram research. Section 2 proposes the AB histogram and its probability histograms to indicate that a bucket satisfies one or more spatial predicates, presents various selectivity estimation algorithms and transform methods of updated histograms, and discusses their merits and faults. Section 3 presents an experimental evaluation of the proposed techniques. Section 4 contains concluding remarks and directions for further work.

1 Related work

There are a number of recent histogram techniques for spatial selectivity estimation. These can be broadly categorized into two classes: those for spatial selection and those for spatial join. Spatial selection and spatial join are two commonly used basic operations in spatial DBMS. A spatial selection finds objects in one dataset that satisfy one or some given spatial operators, whereas the spatial join finds pairs of objects from different datasets that satisfy a given spatial operators, such as “intersect”, “overlap”, and “contains”.

1.1 Selectivity estimation for finer spatial selection operators

A great number of spatial histograms have been used for spatial selection operation estimation. The conventional approach for building a histogram for region objects is as follows. Given a data space divided evenly into grids, allocate a set of buckets with each bucket corresponding to a grid cell. Then, for each object, if it intersects a bucket, increase the value stored in the bucket by 1. This histogram, as in conventional histograms, generally suffers from a severe drawback called multiple-count problem. Euler and CD histograms are designed to address the multiple-count problem. Beigel and Tanin proposed building a spatial histogram based on Euler’s theorem. It has been shown that the techniques developed in [19] can guarantee exact solutions if the given query range aligns with the histogram “grid”. Another technique, the cumulative density algorithm proposed by [12], targeted the same problem as [19]. The Min-Skew algorithm and the SQ-histogram technique investigated the problem of effectively partitioning the data space to accommodate an arbitrary query rectangle.

Maybe CD and Euler algorithms could efficiently return exact selectivity for “intersect” or “disjoin” spatial relation, but they all encounter a native weakness on handling finer spatial relations, that is, the CD and Euler histogram often do not discriminate
some scenarios with Level 2 or 3 spatial relations in Fig. 1. Fig. 1 shows a set of spatial relations at three levels. For instance, the two different scenarios in Fig. 2(a) result in the same CD histograms (Fig. 2(b)) and the same Euler histogram (Fig. 2(c)). Although [14] and [18] proposed some Euler-based algorithms for finer spatial relations, the native weaknesses cannot be bypassed. Thus, it is necessary to find a spatial histogram that can reflect the different spatial relations for different data.

Fig. 1 Spatial relations at different levels

1.2 Selectivity estimation for spatial join

Spatial histograms for spatial join operations have been less explored over past 10 years. Two techniques, however, have been proposed in [20] and [11]. The MP algorithm [20] is based on a sophisticated mechanism to calculate the workspace of two or more selection windows, where objects from different datasets may intersect. The Geometric Histogram (GH) is based on the observation that the intersection of two rectangles always results in four intersection points; therefore, the selectivity of a spatial join can be estimated by first estimating the number of intersection points and then dividing the result by 4.[11]

Even if the estimation errors of these algorithms were acceptable to us, they would still only be able to find intersect pairs of objects from different datasets rather than pairs for any given spatial operator. Thus, it is necessary to find a more generic spatial histogram that can be adapted to selectivity estimation not only for finer spatial selection operators but also for finer spatial joins.

1.3 Generating updated histogram for estimating selectivity of subsequent query

There is scarcely any discussion about how to generate an updated histogram in existing spatial histogram research. In this paper, an updated histogram is defined as an updated version of a original spatial histogram, which is used to indicate the data distribution after a spatial operator has been applied to the original data. The updated histogram is crucial for selectivity estimation in subsequent spatial query operations. For example, the spatial query in Fig. 3(a) may be executed from bottom to top according to the steps of the execution plan tree in Fig. 3(b). The first step is a spatial selection operation to find the forest stands intersecting with a given WINDOW. The next step is a spatial join operation to find the geometric pairs, which satisfy the overlap operator, from the query result of first step and river datasets. The last step is a projection operation to eliminate the same geometric pairs. Thus, it is necessary to generate an updated histogram for a spatial selection node, because the updated histogram will be used to estimate the selectivity of the spatial join operation.

The means to generate updated histograms for estimating selectivity of its parent node is a complicated and crucial problem. All histograms stored in a spatial database are only for the original table data. Additionally, there are no readymade histograms for
intermediate query results. However, it is impossible to build a spatial histogram for intermediate results in real-time. First, it is time-consuming to fetch every object that meets the filter condition and create a spatial histogram thereof. Second, if the elapsed time for query optimization were greater than the execution time of most query plans, query optimization would be meaningless. Therefore, it is necessary to find a way to generate an updated histogram from the original histograms and spatial query restrictions.

2 AB histogram and corresponding algorithms

2.1 AB histograms and its construction algorithms

Both the CD and Euler histograms accumulate various statistical results about certain parts of the Minimum Bounding Rectangle (MBR) and then break the integrality of the MBR of an object. Maybe it results that the different scenarios have the same spatial histogram, as shown in Fig. 2.

The AB histogram would keep the integrality of MBR and make it easy to reason the spatial relation between spatial data and query windows. It is similar to the other spatial histograms dividing the entire data spatial into same-sized grid cells and using an object’s MBRs instead of the object itself. The difference is that this paper takes a collection of cells, which thread by the line of an object’s MBR, as a bucket range instead of those cells in CD or Euler histograms. Therefore, a bucket in AB histogram often covers a single-cell-sized photo frame scope, as the gray frame shows in Fig. 4(a). Then, the bucket count is the number of objects whose MBRs fall between its outer rectangle and inner rectangle. In this paper, the AB histogram of the geometry attribute \( g \) in a recorder set is represented as a set of bucket-count pairs: 

\[
DH(j, g) = \{(R_i^j, C_i^j)\}, \text{ where } 1 \leq i \leq N \text{ and } N \text{ is the number of buckets in the histogram, and } j \text{ is the version of AB data histogram.}
\]

\( DH(0, g) \) is the AB histogram of original spatial data (i.e., the geometry data in a table). \( DH(1, g) \), \( DH(2, g) \), etc., are used to indicate updated versions of the data distribution after some spatial operators have been subsequently applied. Fig. 4(a) graphically shows some spatial objects and their annular buckets. Fig. 4(b) graphically gives the AB histogram of Fig. 4(a).

For example, the bucket-count of bucket \( A \) is 2, which means that there are two objects’ MBRs that fall into its bucket-range, as shown in Fig. 4(a). In Fig. 4(b), the bucket-range is represented as \((x_a, y_a; x_b, y_b)\), where \((x_a, y_a)\) and \((x_b, y_b)\) are the coordinates of the bottom-left and top-right corners, respectively, of the outer rectangle of the annular bucket. Thus, the outer rectangle of the bucket is given as \( BOX(x_a, y_a) \) and the inner rectangle as \( BOX(x_a+1, y_a+1; x_b-1, y_b-1) \).

The algorithm to construct AB histogram is as follows. First, divide the entire space into the same-sized grid cells and initialize an AB histogram without any bucket. Then, for each object, if its MBR does not fall into any bucket existing in the histogram (which means that its MBR is not located between any bucket’s outer rectangle and inner rectangle), add a new annular bucket to the AB histogram and initialize its count as 1. The new bucket-range covers the region of those
neighbor cells threaded by that object’s MBR line. Otherwise, it increases the bucket count found by 1.

There are still two points similar to the other selectivity estimation research on the AB histogram. One is that the geometry of a spatial object must be a replacement for its MBR. The selectivity estimation focuses mainly on the I/O cost of filtering step, \cite{11,21} in which step candidate objects will be filtered out by their MBRs. The I/O cost is determined predominantly by the number of candidate objects in that step. The other is that the AB histogram of a geometry column in a table should be constructed and be stored beforehand in a database system table to accelerate the speed of selectivity estimation.

2.2 Distribution assumption about the AB histogram

The AB histogram maintains the integrality of the MBR because of taking the MBRs of objects as a whole to considerate its falling. Thus, it is easy to reason out an approximate spatial distribution for spatial data. This paper assumes that all MBRs of objects in each bucket are evenly distributed between inner rectangle and outer rectangle. Fig. 5 gives an approximate even distribution of bucket $A$ in Fig. 4(b).

![Fig. 5 An approximate even distribution of bucket A](image)

2.3 Probability histograms for single spatial selection operator

Based on the assumption of even distribution, for every annular bucket in AB histogram, the paper could obtain a probability, that is, the objects in each bucket satisfy a certain spatial selection operator, from the semantic of spatial operator and the spatial relationship between bucket-range and a query range. In DBMS, a spatial selection with single operator is called a “spatial predicate”. An spatial predicate has the form

\[ sp: \quad \text{attribute} \ opr \ constant \]

where $opr$ belongs to the set \{“disjoint”, “intersect”, “contains”, …\} for geometric attributes and to the set \{\text{\textless},\text{\textless},=,\geq,\leq,\text{\textless}\} for traditional attributes; \text{constant} belongs to a query range for geometric attributes. Then, the probability satisfying a spatial predicate is also represented as a set of bucket-range, bucket-probability pairs: \(PH(opr, g) = \{(R_g(i), P_{opr}^g(i, qr))\} \) where $1 \leq i \leq N$ and $N$ is the number of buckets in the histogram, $g$ is the name of geometry attribute, $opr$ is a spatial operator, and $qr$ is a query range constant for spatial operator.

Taking bucket $i$ as an example, the paper gives its probability satisfying different spatial operators under different cases, respectively, as follows. Note that the length, width, or distance in follow logics are calculated in grid reference systems.

(1) Intersection probability between bucket $i$ and a query range. When the outer range of bucket $i$ is disjoint from a query range (“$qr$”), as shown in Fig. 6(a), the probability of intersection with the “$qr$” should be 0. Only when the inner rectangle of bucket $i$ is disjoint from the “$qr$”, as show in Fig. 6(b), the probability should be the minimum of the height and width of the overlapping rectangle region between its the outer rectangle and the “$qr$”. In any other case, the probability is 1.

(2) Within probability between bucket $i$ and a query range. When both the outer rectangle and the inner rectangle of bucket $i$ are not within the “$qr$”, as shown in Fig. 6(c), its probability of within the “$qr$” should be 0. When the outer rectangle of bucket $i$ is not within the “$qr$” but its inner rectangle within the “$qr$”, as shown in Fig. 6(d), the probability should be the minimal gap distance from the four edges of the outer rectangle of bucket $i$ to the four edges of the “$qr$”. In any other case, the probability is 1.

(3) Probability between bucket $i$ and a query range. When both the outer rectangle and the inner rectangle of bucket $i$ do not contain the “$qr$”, as shown in Fig. 6(e), its probability that contains the “$qr$” should be 0. When the outer rectangle of bucket $i$ contains the “$qr$” but its inner rectangle does not contain the “$qr$”, as shown in Fig. 6(e), the probability should be the minimal gap distance from the four edges of the outer rectangle of bucket $i$ to the four edges of “$qr$”. Under the other spatial relationships,
(4) **Disjoint probability between bucket i and a query range.** Since “disjoint” and “intersect” are complementary, the disjoint probability of bucket i can be calculated by Eq.(1):

\[
P_{g_{\text{disjoint}}} (i, qr) = 1 - P_{g_{\text{intersect}}} (i, qr)
\]

(5) **Equal probability between bucket i and a query range.** The probability that two spatial objects are equal is so small that it can be ignored. In this paper, the probability that the object in bucket i equals the “qr” is regarded as 0.

(6) **Overlap probability between bucket i and a query range.** Fig. 1 shows the intersect relation can be divided into four sub-relations: “contains”, “within”, “overlap”, and “equal”. The probability of equal is 0. Thus, the probability of overlap can be calculated by Eq. (2).

\[
P_{g_{\text{overlap}}} (i, qr) = P_{g_{\text{intersect}}} (i, qr) - P_{g_{\text{within}}} (i, qr) - P_{g_{\text{contains}}} (i, qr)
\]

Figs. 7(a) to 7(d) graphically give intersection, within contains, and overlap probability histograms of Fig. 4(a). These probability histograms are derived according to the logic given above.

### 2.4 Probability histograms about more spatial selection operators

The paper assumes the restriction with more spatial selection operators is written (or can be converted to) conjunctive normal form

\[
re: \quad bf(0) \text{ and } bf(1) \text{ and } bf(2) \text{ and} \ldots
\]

where each conjunct, \(bf(i)\), is called a “Boolean factor”. Note that all Boolean factors must be true in order for the qualification to be true. A Boolean factor has the form

\[
\text{bf: } \quad sp(1) \text{ or } sp(2) \text{ or } sp(3) \text{ or} \ldots
\]

where the form of \(sp(i)\) is call a spatial predicate, which has been given in the first paragraph in Section 2.3.

The probability of complex selection could be synthesized by the probability of intersection or union of two events. The probability of spatial predicates is obtained by the logic in the previous section. The value of bucket i in the probability histogram after the application of a Boolean factor with several predicates can be better approximated using Eq. (3) because it connects two predicates with an “or”. Moreover, the value of bucket i in the probability histogram after the application of a restriction with several Boolean factors can be better approximated using Eq. (4) because it connects the two Boolean factors with an “and”.

\[
P_{g_{\text{or}}} (i) = \sum_{j=1}^{n} P_{g_{\text{or}}} (i, \text{cons} \tan t) + \prod_{j=1}^{n} P_{g_{\text{or}}} (i, \text{cons} \tan t)
\]

\[
P_{g_{\text{and}}} (i) = \prod_{j=1}^{n} P_{g_{\text{and}}} (i)
\]

where \(P_{g_{\text{or}}} (i)\) is the probability that bucket i satisfies the \(opr(j)\) operator, \(P_{g_{\text{and}}} (i)\) is the probability that bucket i satisfies a Boolean factor, and \(P_{g_{\text{and}}} (i)\) is the probability that bucket i satisfies a spatial selection restriction.
2.5 Selectivity estimation of spatial selection and its updated AB histogram

According to the AB histogram and its probability histogram, the tuples of selection results (selectivity) could be estimated by the sum of the production between each bucket-count and its probability satisfying a given spatial selection restriction. Taking a “re” restriction will be applied to a column \( (g, g') \) of original data as an example; its selectivity could be estimated by Eq. (5).

\[
S_{g}^{\text{re}} = \sum_{i=1}^{n} (C_{g}^{i}(i) \times P_{g}(i))
\]

where \( C_{g}^{i}(i) \) is the count of bucket \( i \), and \( P_{g}(i) \) is the probability that the objects in bucket \( i \) satisfying the “re” restriction. According to the information AB histogram in Fig. 4(b) and the probability histograms in Figs. 7(a) and (d), the number of objects that intersect the query range is \( 2 \times 1.0 + 3 \times 0.3 + 1 \times 0.0 + 2 \times 1 = 4.9 \), and the number of objects that overlap the query range is \( 2 \times 0.5 + 3 \times 0.3 + 1 \times 0.0 + 2 \times 0.5 = 2.9 \).

If the spatial selection described in the last paragraph corresponds to a leaf node in query execution plan, its updated AB histogram, which indicates the data distribution after the selection restriction has been applied to the original data, is a very important input parameter to estimate the selectivity of its parent node. The objects in updated histogram’s bucket \( i \) should be the objects satisfying that selection restriction. Thus, the range-value of bucket \( i \) in \( DH(j, g) \) is the same with the corresponding bucket’s range-value in \( DH(j-1, g) \), that is, \( R_{g}(i) \), and the count of bucket \( i \) in \( DH(j, g) \) could be calculated by Eq. (6).

\[
C_{g}^{i}(i) = C_{g}^{i-1}(i) \times P_{g}(i)
\]

where \( C_{g}^{i-1}(i) \) and \( C_{g}^{i}(i) \) are the bucket-count of bucket \( i \) in \( DH(j-1, g) \) and \( DH(j, g) \), respectively; \( P_{g}(i) \) is the probability that bucket \( i \) in \( DH(j-1, g) \), satisfying “re” restriction. Fig. 8 gives the updated AB histogram after applying the overlap selection in Fig. 4(a).

![AB histogram after “overlap” has been applied](image)

2.6 Selectivity estimation of spatial join and its updated AB histogram

The spatial join does find pairs of objects from different datasets that satisfy a given spatial operator, for example, finding the pairs in a column \( (g) \) of one dataset that intersect with a column \( (g') \) of the other dataset. The pair is represented as \( (g, g') \) in this paper. Thus, the selectivity of a spatial join should be the tuples of pairs \( (g, g') \); there should be two updated histograms: one for \( g \) and the other for \( g' \).

As proposed in [11,22], one of the two input data histograms of the spatial join is considered as underlying data, while the other is considered as the source for query windows. The query ranges in the source AB histogram can be easily obtained according to uniform distribution in Section 2.2. Taking \( DH(0, g) \) as underlying data and \( DH(0, g') \) as source for the query range, for bucket \( i \) in \( DH(0, g) \), there are serial probabilities \( (P_{g}^{\text{opr}}(i, q_{g}(k)), k=1 \cdots C_{g}^{0}(j)) \) to indicate the degree to satisfy the spatial join operator between it and every query range in bucket \( j \) of \( DH(0, g') \). Then, their average \( \left( \sum_{k=1}^{C_{g}^{0}(j)} p_{g}^{\text{opr}}(i, q_{g}(k)) \right) / C_{g}^{0}(j) \) represents the degree that the bucket \( i \) in \( DH(0, g) \) and bucket \( j \) of \( DH(0, g') \) satisfy that spatial join operator. Therefore, the probabilities of bucket \( i \) in \( DH(0, g) \) appearing in join results is the sum of those average values between bucket \( i \) in \( DH(0, g) \) and every bucket in \( DH(0, g') \). Thus, Eq. (7) can obtain the every bucket-count of the updated histogram of \( g' \) can be obtained by Eq. (8).

\[
C_{g}^{i}(i) = C_{g}^{0}(i) \times \sum_{j=1}^{n} \left( \sum_{k=1}^{C_{g}^{0}(j)} P_{g}^{\text{opr}}(i, q_{g}(k)) \right) / C_{g}^{0}(j)
\]

(7)

\[
C_{g'}^{i}(i) = C_{g'}^{0}(i) \times \sum_{j=1}^{n} \left( \sum_{k=1}^{C_{g}^{0}(j)} P_{g}^{\text{opr}}(i, q_{g}(k)) \right) / C_{g}^{0}(j)
\]

(8)
join operator with the object \( k \) in bucket \( j \) of \( DH(0, g') \); \( P_{g^{'}}^{opr}(i, qr_g(k)) \) is the probability that bucket \( i \) in \( DH(0, g') \) satisfies the “opr” join operator with the object \( k \) in bucket \( j \) of \( DH(0, g) \); and \( m \) and \( n \) are the numbers of buckets in \( DH(0, g) \) and \( DH(0, g') \), respectively.

Spatial join selectivity should be the pairs’ tuples after spatial join of two datasets. Based on the every bucket count in updated histograms, the selectivity of pairs \((g, g')\) could be estimated by summing all bucket counts in one of updated histograms, as shown in Eq. (9).

\[
S_{g,g'}^{opr} = \sum_{i=1}^{m} C_g(i) \quad \text{or} \quad S_{g,g'}^{opr} = \sum_{i=1}^{n} C_g'(i) \quad (9)
\]

In addition, according to the serial probabilities of every bucket, here the tuples can be chosen from two datasets. For every bucket in \( DH(0, g) \), its probability to be fetched out to take part in join operation will be the maximum of those serial probabilities \( P_{g}^{opr}(i, qr_g(l)) \). Therefore, the tuples to be chosen from columns \((g)\) and \((g')\) can be estimated by Eqs. (10) and (11).

\[
S_{g, g'}^{opr} = \sum_{i=1}^{m} C_g(i) \times \max \{ P_{g}^{opr}(i, qr_g(l)) \}; \quad 1 \leq l \leq \sum_{j=1}^{n} C_g'(j) \quad (10)
\]

\[
S_{g, g'}^{opr} = \sum_{i=1}^{n} C_g'(i) \times \max \{ P_{g'}^{opr}(i, qr_{g'}(l)) \}; \quad 1 \leq l \leq \sum_{j=1}^{m} C_g'(j) \quad (11)
\]

\( P_{g'}^{opr}(i, qr_{g'}(l)) \) is the probability that bucket \( i \) in \( DH(0, g') \) satisfies a “opr” join operator with the object \( l \) in \( DH(0, g') \).  

2.7 Discussion

The AB histogram is more generic than the existing spatial histograms discussed in Section 1. First, it supports not only the selectivity estimation for spatial selection but also that for spatial join. Secondly, it can be easily integrated with traditional attribute histograms in spatial database management, since most of the concepts of the AB histogram (e.g., its graphical representation, the method for processing with more operators, and the transformation from original histogram to the updated histogram) are the same as those for traditional attribute histograms, except for the logic to calculate different probabilities for different operators.

The AB histogram is more practical than existing spatial histograms. Besides supporting selectivity estimation for both spatial selection and spatial join, it provides a histogram transformation approach from the original histogram to the updated histogram. Therefore, the transformation makes it possible to estimate the selectivity of parent nodes in a query plan tree and the selectivity of entire query trees.

However, there are still some defects in this version of the AB histogram. The selectivity estimation based on the AB histogram is not as accurate as the CD histogram to estimate intersection selectivity, because the even-distribution assumption in Section 2.2 differs slightly from the real distribution. However, a smaller grid cell size will result in a more accurate estimation. In addition, the AB histogram contains more buckets than a traditional attribute histogram. The primary reason for this is that the probability is too small that the MBRs fall into an annular bucket due to the limitation from several dimensions. A second reason is the AB histogram is an exact histogram and not a range histogram. Thus, it is impossible to try and merge neighboring buckets.

3 Experiments

3.1 Test data and environment

The purpose of our experiments was to test the selectivity estimation accuracy for spatial selection and spatial join based on five spatial relations and to test the accuracy of its updated histogram. The test used 1:10000 scale land-use data of a Chinese county; containing 17207 polygons, the MBR distributions of which are shown in Fig. 9(a). Nine rectangles were defined as shown in Fig. 9(b), called the query range data. In the spatial selection test, the nine rectangles were used as different query ranges to estimate the number of land-use objects that satisfy a certain spatial operator. In the spatial join test, the nine rectangles are taken as the source dataset participating in various spatial joins.

Fig. 9 Map of land-use data and query range data
These data are stored in a PostgreSQL, while the functions to build the AB histogram, to estimate the selectivity of the spatial selection and spatial join, and to generate the subsequent histograms are programmed in C and based on PostGIS. These functions is similar to that functions (i.e., build_histogram2d and estimate_histogram2d) have existed in PostGIS. This paper only provides some selectivity estimation interfaces for internal module of PostgreSQL. It would efficiently work in the query optimizer after some further work in PostgreSQL had been done. However, now the test can use these functions to get the selectivity estimation values for spatial selection and join after preparing corresponding parameters. In order to evaluate the veracity of estimation values, the test used the SQL queries given in Sections 3.2 and 3.3 to simulate the filter step process to arrive at the corresponding actual results. The statistical values for the actual results will be presented as actual values to evaluate those estimation values.

### 3.2 Selectivity test of spatial selection

As described in Section 2.1, the test entire space of land-use data was divided into 60×60 grid cells and built as an AB histogram with 3283 buckets. The test used the selectivity estimation functions programmed using the logic in Section 2.3 to estimate the number of objects that satisfy “disjoint”, “intersect”, “within”, “contains”, and “overlap” relations with the nine query ranges in Fig. 9(b). The estimation results are shown in the first column of each spatial relation group in Table 1. The actual number of selected objects can be obtained by executing a certain GSQL code corresponding spatially. For example, given a query range BOX (507137.087 285226.243,507989.318 285852.359), the actual number of objects that satisfy “disjoint”, “intersect”, “within”, “contains”, and “overlap” relations for the given range can be obtained by the following GSQL:

```sql
select count(*) from landuse
where box2d_intersects('BOX(507137 285226, 507989 285852)'::box2d, box2d(geom))=FALSE

select count(*) from landuse
where box2d_intersects('BOX(507137 285226, 507989 285852)'::box2d, box2d(geom))=TRUE

select count(*) from landuse
where box2d_contain('BOX(507137 285226,507989 285852)'::box2d, box2d(geom))=TRUE

select count(*) from landuse
where box2d_contained('BOX(507137 285226,507989 285852)'::box2d, box2d(geom))=TRUE

select count(*) from landuse_mbr
where ST_overlaps('BOX(507137 285226, 507989 285852)'::box2d, geom)=TRUE
```

Note that the actual selectivity of overlap is obtained by executing the finer filter predication (ST_overlaps) on the more coarse layer of the land-use data’s MBR, because the execution results of box2d_Overlap are the same as box2d_intersects and do not meet the requirements of overlap. The actual selectivity values are given in the second column of each spatial relation group in Table 1. Thus, the relative error ratio ((Est. value-Actual value)/Actual value) is obtained as shown in the third column of each group in Table 1.

#### Table 1 Estimated selectivity, actual selectivity, and their relative error ratios for different spatial relations and nine query ranges

|     | Disjoint | Intersect | Within | Contain | Overlap |
|-----|----------|-----------|--------|---------|--------|
|     | Est. value | Actual value | Error ratio(%) | Est. value | Actual value | Error ratio(%) | Est. value | Actual value | Error ratio(%) | Est. value | Actual value | Error ratio(%) |
| Q1  | 16523.24 | 16534 | -0.07 | 683.76 | 673 | 1.60 | 506.86 | 536 | -5.44 | 0.00 | 0 | # | 176.90 | 137 | 29.12 |
| Q2  | 16317.83 | 16315 | 0.02 | 899.17 | 892 | -0.32 | 758.34 | 789 | -3.89 | 0.00 | 0 | # | 130.83 | 103 | 27.02 |
| Q3  | 17163.16 | 17164 | 0.00 | 43.84 | 43 | 1.95 | 7.94 | 13 | -38.92 | 1.00 | 1 | 0.00 | 34.9 | 29 | 20.34 |
| Q4  | 15954.99 | 15950 | 0.03 | 1252.01 | 1257 | -0.40 | 996.27 | 1060 | -6.01 | 0.00 | 0 | # | 255.74 | 197 | 29.82 |
| Q5  | 16963.35 | 16954 | 0.06 | 243.65 | 253 | -3.70 | 163.20 | 188 | -13.19 | 0.00 | 0 | # | 80.45 | 65 | 23.77 |
| Q6  | 17100.55 | 17101 | 0.00 | 106.45 | 106 | 0.42 | 73.58 | 79 | -6.86 | 0.00 | 0 | # | 32.87 | 27 | 21.74 |
| Q7  | 17145.94 | 17142 | 0.02 | 61.07 | 65 | -6.05 | 36.03 | 42 | -14.21 | 0.95 | 1 | -5.00 | 24.09 | 22 | 9.50 |
| Q8  | 17192.21 | 17195 | -0.02 | 14.80 | 12 | 23.33 | 2.36 | 2 | 18.00 | 3.20 | 3 | 6.67 | 9.24 | 7 | 32.00 |
| Q9  | 17202.85 | 17205 | -0.01 | 3.15 | 2 | 57.50 | 0.19 | 0 | # | 1.00 | 1 | 0.00 | 1.56 | 1 | 56.00 |
3.3 Selectivity and subsequent histogram test of spatial join

As described in Section 2.6, the histogram of the data in Fig. 9(b) was built after dividing the entire space into 40×40 grids. Taking the land-use data \((D_1)\) as underlying data and the query-range data \((D_2)\) as source for query windows, the test used the selectivity estimation functions programmed using Eq. (10) to estimate the number of object from \(D_1\), which satisfies the “disjoint”, “intersect”, “within”, “contains”, and “overlap”, respectively. The first row in Table 2 shows those values. The number of pairs \((D_1,D_2)\) that respectively satisfies that five operators mentioned above could be obtained by the functions using the former formula in Eq. (9). The third row in Table 3 shows those values. On the contrary, the test could obtain the number of object from \(D_2\) by (11) after taking \(D_2\) as underlying data and \(D_1\) as source for query range. The second row in Table 2 shows those values.

Taking the intersect operator as an example, the actual number of objects from the land-use and query-range data is the number of tuples of the data-sets obtained by the following GSQLs, respectively:

\[
\text{select distinct (landuse.ogc_fid) from landuse, Queryrange}
\]

\[
\text{where box2d_intersects (Queryrange.the_geom, landuse.the_geom)=TRUE}
\]

\[
\text{select distinct (Queryrange.ogc_fid) from landuse, Queryrange}
\]

\[
\text{where box2d_intersects (Queryrange.the_geom, landuse.the_geom)=TRUE.}
\]

The actual number of pairs of objects composed from the land-use data and query-range data is the number of tuples of the datasets obtained by the following GSQL:

\[
\text{select landuse.ogc_fid, Queryrange.ogc_fid}
\]

from landuse, Queryrange

\[
\text{where box2d_intersects(Queryrange.the_geom, landuse.the_geom)= TRUE.}
\]

The actual number of objects and pairs that satisfy other predicates can be obtained by the appropriate GSQL query after substituting the corresponding predication or data. The actual selectivity values are given in the second column of each spatial relation group in Table 2. Thus, the relative error ratio is calculated as shown in the third column of each group in Table 2.

| Disjoint | Intersect | Within | Contains | Overlap |
|----------|-----------|--------|----------|---------|
| \((D_1)\) | 17207.00  | 3225.75 | 2467.69  | 794.36  |
| \((D_2)\) | 9.00      | 8.78   | 8.21     | 7.96    |

| \((D_1)\) | 15156.97 | 3006.03 | 2238.63 | 784.30  |
| \((D_2)\) | 151560.00| 3303.00 | 2179.63 | 588.00  |

* \(D_1\) represents land-use data and \(D_2\) represents query-range data.

The test used histogram transformation functions programmed by Eqs. (7) and (8) to generate two AB histograms and stored them in the database; one histogram is for the geometry column of \(D_1\) in pairs \((D_1,D_2)\), while the other is for the geometry of \(D_2\) in pairs \((D_1,D_2)\). Taking “intersect” as an example, the pairs of objects after the spatial join can be expressed as an “intersect_result” table by the following GSQL:

\[
\text{select distinct (landuse.ogc_fid) from landuse, Queryrange.the_geom}
\]

\[
\text{where box2d_intersects (Queryrange.the_geom, landuse.the_geom)=TRUE}
\]

Then, the actual histograms of the spatial join can be built on the two geometry columns of the material table. According to the estimated histogram and actual histogram of the geometry of land use, it is easy to calculate serial relative estimation error ra-
tions for every two corresponding buckets. Then, the average and variance can be used to reflect the difference between the estimated histogram and actual histogram. These averages and variances are given in Table 3.

### Table 3  Average and variance of the relative estimation error ratios between two corresponding buckets

| Land-use  | Query-range | Avg. Relative Est. Error | Var. Relative Est. Error | Avg. Relative Est. Error | Var. Relative Est. Error |
|-----------|-------------|--------------------------|-------------------------|--------------------------|-------------------------|
| Disjoint  |             | 0.0379% 0.014            | 4.0157% 0.0014          |                         |                         |
| Intersect |             | 0.5878% 0.114            | 2.8966% 0.3751          |                         |                         |
| Within    |             | 1.3122% 0.112            | 7.0942% 0.4753          |                         |                         |
| Contains  |             | 0.0037% 0.005            | 44.2616% 9.5196         |                         |                         |
| Overlap   |             | 0.0185% 0.017            | 21.2037% 3.2248         |                         |                         |

### 3.4 Summary

The tests described here show that the AB histogram can be used to estimate the selectivity of spatial selection and spatial join operations with “disjoint”, “intersect”, “within”, “contains”, and “overlap” relations. Therefore, the AB histogram is a more generic approach to estimate selectivity of the spatial selection and spatial join.

The differences between the estimated and actual values in Tables 1 and 2 show that the estimated selectivity based on the AB histogram has high accuracy. In Tables 1 and 2, most of the relative error ratios are below 15%, although some ratios are relatively higher. However, some of the higher ratios are caused by a smaller base number. The other higher ratios are from the overlap predicate. Correspondingly, the estimation errors for the “disjoint”, “intersect”, “within”, and “contains” relations are smaller than that for overlap. The reason for this may be accumulated errors. In this paper, the probability of overlap is excluded by the probabilities of “intersect”, “within”, and “contains” in Eq. (2), because the situation of overlap is too complex to obtain its probability. A better approach for estimating selectivity of overlap needs to be developed.

Table 3 shows that the updated AB histograms for the spatial join are close to the distribution of the AB histograms built from the actual query results. Therefore, the histogram generated from two AB histograms can be used reliably to estimate the selectivity of higher nodes in the plan tree.

Tests show that the AB histogram requires more buckets than traditional histograms. One reason for this is that the probability that the MBR of an object falls into an annular bucket is not very high in most spatial data. The other reason is that the AB histogram is an exact histogram rather than a range histogram and it is impossible to try and combine buckets. Therefore, how to reduce the number of buckets is another problem for further investigation.

### 4 Conclusion

The paper proposed a more generic and practical spatial histogram (AB histogram), presented selectivity estimation methods for complex spatial selection and spatial join operations, and discussed the histogram transformation from the original data to the query result data. The selectivity of almost all spatial operations can be estimated from the AB histogram and corresponding probability model, since the spatial data distribution is better represented. In addition, it is easy to integrate with traditional attribute histograms in spatial database management, since most of the concepts of the AB histogram (e.g., its graphical representation, a method for processing with more restrictions, and the histogram transformation) are the same as those for traditional attribute histograms, except for the logic used to calculate different probabilities for different operators. Finally, the update histogram transformation makes it possible to estimate the selectivity of parent nodes in a query plan tree.

There is some further work that needs to be done. Still based on the idea of the annular bucket, it is necessary to develop a more accurate selectivity estimation method for overlap. In addition, it is necessary to investigate a mutation AB histogram representation that could more accurately describe the distribution of spatial data with fewer buckets.

## References

[1] Jiang Songtao, Lee B S, Zhen He (2007) Cost modeling of spatial operators using non-parametric regression [J].
Information Sciences, 177: 607-631

[2] Bjorke J T, Nilsen S (2003) Wavelets applied to simplification of digital terrain models [J]. International Journal of Geographical Information Science, 17: 601-621

[3] Yang Bisheng, Purves R S, Weibel R (2008) Variable-resolution compression of vector data [J]. Geoinformatic, 12: 357-376

[4] Van Oosterom P, Schenkelaar V (1995) The development of an interactive multiscale GIS [J]. International Journal of Geographical Information Systems, 9: 489-507

[5] Prasher S, Zhou Xiaofang (2003) Efficient update and retrieval of objects in a multiresolution geospatial database [C]. Proceedings of SSDBM 2003, Cambridge, MA, USA

[6] Cheng Changxiu, Niu Fangqu, Cai Jun, et al. (2008) Extensions of GAP-tree and its implementation based on a non-topological data model [J]. International Journal of Geographical Information Science, 22, 657-673

[7] Buttenfield B (1999) Progressive transmission of vector data on the Internet: A cartographic solution [C]. The 18th International Cartographic Association Conference, Stockholm, Sweden

[8] Bertolotto M, Egenhofer M J (2001) Progressive transmission of vector map data over the World Wide Web [J]. GeoInformatica, 5(4): 345-373

[9] Cheng Changxiu, Lu Feng, Cai Jun (2009) A quantitative scale-setting approach for building multi-scale spatial databases [J]. Computer & GeoScience, 35: 2004-2209

[10] Aboulnaga A, Chaudhuri S (1999) Self-tuning histograms: building histograms without looking at data [C]. 1999 ACM SIGMOD, Philadelphia, PA, USA

[11] An Ning, Yang Zhenyu, Sivasubramaniam A (2001) Selectivity estimation for spatial joins [C]. The 17th International Conference on Data Engineering, Heidelberg, Germany

[12] Jin Ji, An Ning, Sivasubramaniam A (2000) Analyzing range queries on spatial data [C]. The 16th International Conference on Data Engineering, San Diego, California, USA

[13] Acharya S, Poosala V, Ramaswamy S (1999) Selectivity estimation in spatial database [C]. 1999 ACM SIGMOD, Philadelphia, PA, USA

[14] Liu Qing, Yuan Yidong, Lin Xuemin (2003) Multi-resolution algorithms for building spatial histograms [C]. The 14th Australasian Database Conference, Adelaide, South Australia

[15] Sun Chengyu, Bandi N, Agrawal D, et al. (2006) Exploring spatial datasets with histograms [J]. Distributed and Parallel Databases, 20(1): 57-88

[16] Haas P J, Swami A N (1995) Sampling-based selectivity estimation for joins using augmented frequent value statistics [C]. The 11th International Conference on Data Engineering, Taipei

[17] Faloutsos C, Seeger B, Traina A, et al. (2000) Spatial join selectivity using power laws [C]. 2000 ACM SIGMOD International Conference on Management of Data, Dallas, Texas, USA

[18] Sun Chengyu, Agrawal D, Abbbi A E (2002) Selectivity estimation for spatial joins with geometric selections [J]. Lecture Notes in Computer Science, 2287: 359-360

[19] Beigel R, Tanin E (1998) The geometry of browsing [C]. The Latin American Symposium on Theoretical Informatics, Brazil

[20] Mamoulis N, Papadias D (2001) Selectivity estimation of complex spatial queries [J]. Lecture Notes in Computer Science, 2121: 155-174

[21] Aboulnaga A, Naughton J F (2000) Accurate estimation of the cost of spatial selections [C]. The 16th International Conference on Data Engineering, San Diego, California, USA

[22] Aref W G, Samet H (1994) A cost model for query optimization using R-trees [C]. The 2nd ACM Conference on Geographical Information Systems (GIS), Gaithersburg, Maryland, USA