# Applying Advanced Methods of Query Selectivity Estimation in Oracle DBMS

Dariusz R. Augustyn

**Abstract.** The paper shows the solution of the query selectivity estimation problem for certain types of database queries with a selection condition based on several table attributes. The selectivity parameter allows for estimating a size of data satisfying a query condition. An estimator of a multidimensional probability density function is required for an accurate selectivity calculation for conditions involving many attributes and correlated attribute values. Using multidimensional histogram as a non-parametric density function estimator is mostly too much storage-consuming. The implementation of the known unconventional storage-efficient approach based on Discrete Cosine Transform spectrum of a multidimensional histogram is presented. This solution extends functionality of the Oracle DBMS cost-based query optimizer. The method of experimental obtaining error-optimal parameters values of spectrum storage for typical attributes value distributions is considered.

**Keywords:** selectivity query estimation, multidimensional probability density function, Discrete Cosine Transform, database query optimizer extension, Oracle Data Catridge Interface Statistics.

## 1 Introduction

The paper presents the practical solution of a query selectivity estimation problem for range queries with a composed selection condition based on a few table attributes with a continuous domain. The selectivity factor is used by the database query optimizer for estimating the size of data satisfying query condition. Using a value of estimated selectivity the optimizer can choose the most efficient method of query executing (so-called the best query execution plan). For single-table queries the

Dariusz R. Augustyn
Institute of Informatics, Silesian University of Technology,
Akademicka 16, 44-100 Gliwice, Poland
e-mail: `draugustyn@polsl.pl`

K.A. Cyran et al. (Eds.): Man-Machine Interactions, AISC 59, pp. 585–593.
springerlink.com 

selectivity value is a number of rows satisfying query condition divided by a number of all table rows.

Most of Database Management System (DBMS) query optimizers are based on an independence attribute value assumption (AVI [7]) that selectivity for a composite condition is a product of simple component condition selectivities. The AVI assumption is based on the probability multiplication rule for independent events. Mostly, the AVI rule using results in an inaccuracy of obtained values of a query selectivity estimator for correlated data. An estimator of a multidimensional probability density function (PDF) is required for accurate selectivity calculations for query conditions involving many attributes. For continuous attribute domains the selectivity value of a range query is a value of definite integral of multivariate PDF.

The direct use of a multidimensional histogram as a non-parametric estimator of PDF is very space consuming for high dimensions. A space-efficient method of representation of attribute value joint distribution is needed. There are many techniques of a multidimensional distribution representation used for selectivity estimation e.g. kernel estimator [8, 4], PHASED [7], MHIST [7], GENHIST [4], STHoles [1], Bayesian Network [3], Discrete Cosine Transform [5], cosine series (with a triangular sampling zone) [9], Discrete Wavelets Transform [2], and many others.

This paper concentrates on the application of the approach using Discrete Cosine Transform (DCT [5]). The presented software solution extends the functionality of the Oracle DBMS query optimizer by using the Oracle Data Cartridge Interface Statistics module [6]. It is the implementation of a spectrum-based selectivity estimation method.

The method of an experimental obtaining of selectivity estimation error-optimal parameters for multidimensional DCT spectrum storage is also presented.

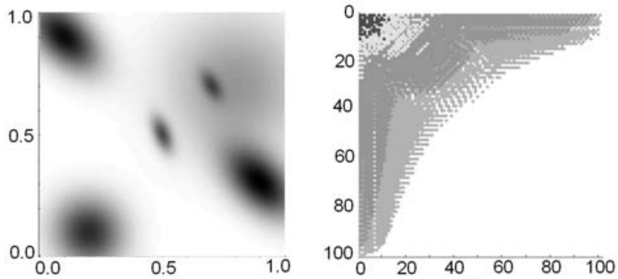
## 2 Discrete Cosine Transform Spectrum Representation of Multidimensional Attribute Value Distribution

The approach to the selectivity estimation method using DCT was proposed in [5], where a space-efficient DCT-spectrum representation of a database table attribute values distribution was considered. The spectrum-based selectivity calculation method was proposed in [5] as well.

For a 2-dimensional case and given definitions:

- $X, Y$ – attributes of relation $R$; both with continuous domain,
- $F$ – $\{f(m,n) : m = 0,\ldots,M-1 \wedge n = 0,\ldots,N-1\}$, $M \times N$ matrix of frequencies, estimator of PDF for joint distribution of $X$ and $Y$, values of a 2-dimensional equi-width histogram,
- $G$ – $\{g(u,v) : u = 0,\ldots,M-1 \wedge v = 0,\ldots,N-1\}$, $M \times N$ matrix of DCT coefficients,

The 2-dimensional Discrete Cosine Transform (DCT spectrum) is defined as follows:




a) b)

**Fig. 1** **(a)** Sample bivariate PDF of 6 Gaussian clusters. **(b)** Corresponding DCT spectrum with regions of comparable coefficient absolute values

$$g(u,v) = \sqrt{\tfrac{2}{M}} k_u \sum_{m=0}^{M-1} \left\{ \sqrt{\tfrac{2}{N}} k_v \sum_{n=0}^{N-1} f(m,n) \cos\left(\tfrac{(2n+1)v\pi}{2N}\right) \right\} \times \times \cos\left(\tfrac{(2m+1)u\pi}{2M}\right), \tag{1}$$

$$\text{where } k_r = \begin{cases} 1/\sqrt{2} & \text{for } r = 0, \\ 1 & \text{for } r \neq 0, \end{cases}$$

and the 2-dimensional Inverse Discrete Cosine Transform (IDCT) is defined below:

$$f(m,n) = \sqrt{\tfrac{2}{M}} \sum_{u=0}^{M-1} k_u \left\{ \sqrt{\tfrac{2}{N}} \sum_{v=0}^{N-1} k_v g(u,v) \cos\left(\tfrac{(2n+1)v\pi}{2N}\right) \right\} \times \times \cos\left(\tfrac{(2m+1)u\pi}{2M}\right). \tag{2}$$

DCT and IDCT can be easily extended for transforming a $k$-dimensional $F$ hyper-rectangle and $G$ hyper-rectangle.

The energy compaction property of DCT enables a storage-efficient $k$-dimensional joint distribution representation. For correlated data most of significant spectrum coefficients $g(u_{1i},\ldots,u_{ki})$ are concentrated near the coordinate system origin of the $U_1 \times \cdots \times U_k$ space. This property is presented in Fig. 1 for the 2-dimensional case. A sample PDF of distribution with 6 Gaussian clusters (Fig. 1a) and corresponding DCT-spectrum (Fig. 1b) were shown. Different grayscales of regions in Fig. 1b show grouped spectrum coefficients that absolute values are within intervals: $(+\infty,10],(10,1],(1,0.1],(0.1,0.01],(0.01,0]$.

For correlated data many small absolute value coefficients could be omitted (zeroed) without a significant loss of accuracy of the data reconstructed from such reduced spectrum. There are many methods of assigning a region in the $U_1 \times \cdots \times U_k$ space for zeroing coefficients values. These methods are known as zonal sampling techniques [5] (e.g., rectangular, spherical, triangular, reciprocal). The best accuracy experimental results are achieved by applying the reciprocal zonal sampling defined below (from [5]):

$$Z = \left\{ (u_1, \ldots, u_k) \colon \prod_{j=1}^{k} (u_j + 1) \le b \right\}. \tag{3}$$

Parameter $b$ determines a depth of cutting the spectrum region. Such lossy compressed spectrum will be denoted as $G^\wedge$. Shapes of sample spectrum regions shown in Fig.1b are approximately compliant with reciprocal zones.

## 3 DCT-Spectrum-Based Selectivity Estimation

The most important advantage of the approach in [5] is the capability of selectivity calculation directly from spectrum $G$ (without a reconstruction of the histogram $F$).

This technique will be presented for the 2-dimensional space. Using definitions:

- $X, Y$ – relation attributes with domains normalized to $[0,1]$,
- $Q$ – range query with selection condition: $a < X < b \wedge c < Y < d$,
- $X \times Y$ – $[0,1]^2$ space divided into $M \times N$ partitions by set of pairs $(x_i, y_j)$: $x_i = \frac{2i+1}{2M},\ y_i = \frac{2j+1}{2N},\ i = 0, \ldots, M-1,\ j = 0, \ldots, N-1$,

distribution can by expressed using $x_m, y_n$ in (2) instead of $m, n$:

$$\begin{aligned} f_{xy}(x_m, y_n) &= f(m,n) = \\ &= \sqrt{\tfrac{2}{M}} \textstyle\sum_{u=0}^{M-1} k_u \left\{ \sqrt{\tfrac{2}{N}} \textstyle\sum_{v=0}^{N-1} k_v g(u,v) \cos(y_n v \pi) \right\} \cos(x_m u \pi) \end{aligned} \tag{4}$$

and selectivity of query $Q$ can be obtained as follows (from [5]):

$$\begin{aligned} sel &= \textstyle\int_c^d \int_a^b f_{xy}(x,y)\ \mathrm{d}x\mathrm{d}y = \\ &= \textstyle\int_a^b \sqrt{\tfrac{2}{M}} \sum_{u=0}^{M-1} k_u \left\{ \int_c^d \sqrt{\tfrac{2}{N}} \sum_{v=0}^{N-1} k_v g(u,v) \cos(yv\pi)\ \mathrm{d}y \right\} \cos(xu\pi)\ \mathrm{d}x. \end{aligned} \tag{5}$$

Finally, (4) and (5) enable to obtain the estimator of selectivity (from [5]):

$$sel^\wedge = \sqrt{\tfrac{2}{M}} \sqrt{\tfrac{2}{N}} \textstyle\sum_{(u,v) \in Z} k_u k_v g(u,v) \int_c^d \cos(v\pi y)\ \mathrm{d}y \int_a^b \cos(u\pi x)\ \mathrm{d}x \tag{6}$$

by using not all spectrum coefficients $g(u,v)$, but only these ones where $(u,v) \in Z$, according to assumed reciprocal zonal sampling:

$$Z = \{(u,v) \colon (u+1)(v+1) \le b\}.$$

For validation of estimation accuracy after zoning, the formula of the relative error was assumed as follows:

$$ERR = \frac{|sel - sel^\wedge|}{sel} \times 100\%. \tag{7}$$

## 4 Implementing Selectivity Estimation in Oracle DBMS

ODCIStats (**O**racle **D**ata **C**artridge **I**nterface **Stat**istics) is a mechanism for extending the functionality of standard Oracle DBMS statistics [6]. It supports creating domain-specific user-defined extensions for the query optimizer module. Those extensions can be easily maintained by administrators using standard Oracle commands (e.g., ANALYZE TABLE, EXEC DBMS_STATS.gather_table_stats).

The use of DCT-statistics implementation is presented for a simple domain of points in a 2-dimensional space $[0,1]^2$. Relevant software elements for this sample solution are shown below. The main functionality of DCT-statistics is implemented in a Java package, which is registered in the database catalog.

*CreateZonedSpectrum* Java method creates a 2-dimensional DCT-spectrum for *schema_name.table_name.col_name* attribute for given $b$ (depth of spectrum cutting) and $N$ (size of the $N \times N$ histogram). The spectrum representation is created in the operating memory and then persisted in a newly inserted table row (in a BLOB-type column). The identifier of the spectrum representation is returned.

```
int CreateZonedSpectrum (String schema_name,
                         String table_name,
                         String col_name, int b, int N)
```

*CountSelectivity* Java method retrieves the DCT-spectrum from database into operating memory using *id_stat* identifier and then returns a calculated selectivity for the query $Q(a < X < b \wedge c < Y < d)$ according to (6).

```
double CountSelectivity (int id_stat, double a, double b,
                         double c, double d)
```

*PointType* defined below is a domain composed type.

```
CREATE TYPE PointType AS OBJECT (x NUMBER, y NUMBER)
```

*SomeTable* is a domain table with a standard-type attribute (*id*) and a user-type attribute (*atr*). Values of *atr.x* and *atr.y* are correlated.

```
CREATE TABLE SomeTab (id NUMBER, atr PointType)
```

*IncludePointChckFnc* is a domain PL/SQL function which acts on user-type objects. It returns the value 1 if a point (*arg*) is placed inside a rectangle defined by a top-left corner (*tl*) and a bottom-right one (*br*).

```
CREATE FUNCTION IncludePointChkFnc
  (arg PointType, tl PointType, br PointType) RETURN NUMBER ...
```

*PointDCTStatsType* is a type which implements the ODCIStats interface. It is responsible for plugging the user-defined statistics functionality into DBMS.

```
CREATE TYPE PointDCTStatsType AS OBJECT (
   STATIC FUNCTION ODCIStatsCollect
   (col sys.ODCIColInfo, ...) ...
```

```
STATIC FUNCTION ODCIStatsSelectivity(..., sel OUT NUMBER,
   ..., arg PointType, top_left PointType,
   bottom_right PointType, ...) ...)
```

*ODCIStatsCollect* function creates a statistics for values of *col* table column by invoking *CreateZonedSpectrum*. *ODCIStatsSelectivity* function calculates the selectivity by invoking *CountSelectivity* and returns the selectivity value in *sel*.

After an association created by the command listed below, *PointDCTStatsType.-ODCIStatsSelectivity* function will be invoked for a selectivity estimation for any query with a selection condition based on *IncludePointFnc*.

```
ASSOCIATE STATISTICS WITH FUNCTIONS IncludePointFnc
   USING PointDCTStatsType
```

The command listed below associates *atr* attribute of *SomeTab* table with *Point-DCTStatsType*. When ANALYZE TABLE command is executed *PointDCTStat-sType.ODCIStatsCollect* function will be invoked for *atr*.

```
ASSOCIATE STATISTICS WITH TYPES PointType USING
   PointDCTStatsType
```

The next command creates a so-called database statistics for all attributes of *SomeTab*.

```
ANALYZE TABLE SomeTab COMPUTE STATISTICS
```

The following are created: a standard 1-dimensional equi-depth histogram for *id* attribute and a non-standard 2-dimensional DCT-spectrum for *atr* attribute.

Finally, for a sample range query $Q(0 < atr.x < 5 \wedge 1 < atr.y < 4)$

```
SELECT  *  FROM  SomeTab  WHERE
   IncludePointCheckFnc(atr, PointType(0,1), PointType(5,4))=1
```

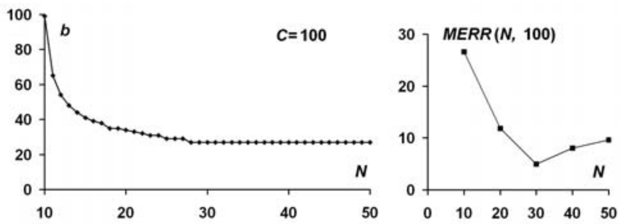
the user-defined DCT-spectrum-based selectivity estimation function is used in the process of obtaining query execution plan.

The DCT-spectrum-based method will be used by the optimizer for any query with a selection condition based on *IncludePointCheckFnc*.

## 5 Tuning of DCT-Statistics Storage Parameters

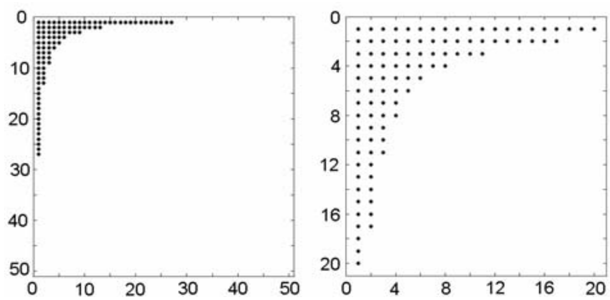
The previous section shows the use of the implemented DCT-statistics. However, there were no hints for the database administrator what values of $b$ (depth of the spectrum reciprocal zone) and $N$ (size of the spectrum hypercube) should be used.

$C$ shall denote a given total number of coefficients which are planned to store in a database catalog. The value of $C$ is the cardinality of set $Z$ (3). $C$ determinates the space size required for storing the DCT-spectrum-based statistics. This section explains a proposed method for obtaining values of parameters $b$ and $N$ for given $C$ using the criterion of the least mean selectivity estimation error (based on (6)). The idea behind this simple method will be explained for the 2-dimensional case.




(a) (b)

**Fig. 2** **(a)** $b(N)$ – depth of spectrum cropping as a function of resolution for $C = 100$ – given the total number of coefficients. **(b)** Experimentally obtained $MERR(N, 100)$ – mean relative selectivity estimation error



(a) (b)

**Fig. 3** Sampling zones for resolutions $N = 50$ (**a**) and $N = 20$ (**b**) where $C = 100$

Figure 2a shows $b(N)$ function for $C = 100$. This dependency between $b$ and $N$ is obtained from (3).

The distribution representation becomes more accurate when $N$ (histogram resolution) becomes larger. Then histogram $F$ better approximates PDF resulting in a better accuracy of the selectivity estimation based on corresponding full spectrum $G$. However, because of the low bandwidth filter property of DCT this rule may not be true for the cropped spectrum $G$^(with the reciprocal zone and the predeterminated value of $C$). For example, which spectrum shown in Fig. 3 will give less selectivity estimation error? The problem can by generally formulated as higher histogram resolution versus stronger low bandwidth filter of zoning (the effect of concentration of the reciprocal zone near the spectrum-space origin).

The set of tests was implemented in Matlab for experimentally finding the best spectrum representation for the predefined storage size. Test returns optimal value of $N$ for a given value of $C$ and the least $MERR(N,C)$ – approximate mean selectivity estimation error.

The pseudocode of the algorithm for calculation $MERR(N,C)$ is presented in Fig. 4.

40 distributions with random numbers of different Gaussian cluster were used in line 03. 40 different query range pairs were used in line 06. The left range query

```
Set values for N and C
Obtain value of b for given N and C (see Fig. 2a for C=100)
For each distribution D from set of sample distributions:
  Obtain frequencies matrix F for distribution D
  Obtain compressed spectrum G^ for F
  For randomly generated query range bounds (a, b, c, d):
    Calculate estimated sel^ from G^
    Calculate accurate sel directly from PDF of D
    Obtain ERR value using sel and sel^ (see (7))
Calculate MERR by averaging all over ERR values
```

**Fig. 4** Pseudocode of the algorithm for calculation $MERR(N,C)$

bound $a$ was generated using uniform distribution $[0,1)$. The right query range bound $b$ was generated from $(a,1]$ using a truncated exponent distribution to prefer smaller query ranges. The same method was used for $c$ and $d$ range bounds generation for the second dimension.

Figure 2b shows an experimental result – the numerically obtained $MERR(N,100)$ function. The optimal value of $N$ for the least $MERR(N,100)$ is about 30.

The method of finding the error-optimal value of $N$ for given $C$ (based on the described algorithm) can be used for more than 2-dimensional case. Such obtained set of values ($b$ and $N$ for a predeterminated $C$ and number of dimensions) may be stored in the database catalog and used implicitly by the DBMS extension module (presented in the previous section) when the DCT-spectrum will be created.

The method is similar to the one presented in [5], but this method only concerns the reciprocal zone using.

## 6 Conclusions

The paper affects the problem of an accurate selectivity calculation for multi-attribute query selection conditions. For accuracy reason a representation of multidimensional PDF of attribute values is required.

The approach based on storage-efficient Discrete Cosine Transform representation [5] was implemented. The presented practical solution of selectivity estimation is fully integrated with Oracle DBMS. It is based on the Oracle ODCIStat interface for extending functionality of the cost-based query optimizer. Using user-defined DCT-based statistics extension is transparent for formulated queries. The paper shows the simplicity of using developed software elements.

The problem of obtaining an optimal resolution of the distribution representation for a predeterminated number of DCT spectrum coefficients was discussed. The method of experimental finding error-optimal size of cropped spectrum was shown and applied.

Future work may concentrate on extending the method and implementation by distinguishing the approach for each distribution dimension (using frequency hyper-rectangle instead of hypercube). Each edge size of a multidimensional histogram hyper-rectangle can be calculated separately using the criterion of the least AMISE (approximated mean integrated standard error [8]) for each equi-width 1-dimensional histogram of marginal distribution.

## References


1. Bruno, N., Chaudhuri, S., Gravano, L.: STHoles: a multidimensional workload-aware histogram. In: Proceedings of ACM SIGMOD International Conference on Management of Data, New York, US, pp. 211–222 (2001)
2. Chakrabarti, K., Garofalakis, M., Rastogi, R., Shim, K.: Approximate query processing using wavelets. The Very Large DataBases Journal 10(2-3), 199–223 (2001)
3. Getoor, L., Taskar, B., Koller, D.: Selectivity estimation using probabilistic models. In: Proceedings of ACM SIGMOD International Conference on Management of Data, New York, US, pp. 461–472 (2001)
4. Gunopulos, D., Kollios, G., Tsortas, V.J., Domeniconi, C.: Selectivity estimator for multidimensional range queries over real attributes. The Very Large DataBases Journal 14(2), 137–154 (2005)
5. Lee, L., Deok-Hwan, K., Chin-Wan, C.: Multi-dimensional selectivity estimation using compressed histogram estimation information. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Philadelphia, US, pp. 205–214 (1999)
6. Oracle: Oracle 10g. using extensible optimizer, `http://download.oracle.com/docs/cd/B14117_01/appdev.101/b10800/dciextopt.htm`
7. Possala, V., Ioannidis, Y.E.: Selectivity estimation without the attribute value independence assumption. In: Proceedings of the 23rd International Conference on Very Large Databases, Athens, Greece, pp. 486–495 (1997)
8. Scott, D.W., Sain, S.R.: Multi-dimensional Density Estimator. Handbook of Statistics, vol. 24. North-Holland Publishing Company, Amsterdam (2004)
9. Yan, F., Hou, W.C., Jiang, Z., Luo, C., Zhu, Q.: Selectivity estimation of range queries based on data density approximation via cosine series. Data & Knowledge Engineering 63(3), 855–878 (2007)