Consiglio Nazionale delle Ricerche

On the Region Proximity in Metrics Spaces

Pavel Zezula, Paolo Ciaccia, Pasquale Savino, Fausto Rabitti

I.E.I. B4-15-06-98

On the Region Proximity in Metric Spaces

Pavel Zezula
IEI-CNR
Pisa, Italy
zezula@iei.pi.cnr.it

Pasquale Savino
IEI-CNR
Pisa, Italy
savino@iei.pi.cnr.it

Paolo Ciaccia
DEIS - CSITE-CNR
Bologna, Italy
pciaccia@deis.unibo.it

Fausto Rabitti
CNUCE-CNR
Pisa, Italy
F.Rabitti@cnuce.cnr.it

Abstract

The problem of defining a measure of *proximity* for regions of generic metric spaces, as needed for designing and implementing indexes for similarity retrieval, is investigated. Though the proposed probabilistic approach is valid for arbitrary regions, specific ready-to-use formulas are developed for the important case of *ball regions*, considering both the uniform and the actual distribution of objects' pair-wise distances. The theoretical results are verified by experiments. Possible applications of the approach to practical data and index management problems are discussed.

Categories and Subject Descriptors: **E.1 [Data Structures]:** *Trees*; **E.5 [Files]:** *Searching*; **H.2.2. [Database Management]:** Physical Design - *Access methods*

Other Keywords: information retrieval, distance-only data, metric regions, algorithms, performance evaluation

1 Introduction

As the volume of data processed by computers grows at an enormous speed, indexing of data files is becoming more and more important. Though numerous significantly different indexing designs exist, they are all based on a form of *partitioning* – a file of objects characterised by their keys is divided into parts constrained in their proper *regions*, and when a query is issued, only some parts are searched in order to find qualifying objects. A specific implementation depends on the type of keys, representing the objects' content, and on the type of queries which are used for searching. The traditional keys have a form of *attributes*, which can be ordered, thus regions of attribute-based files are typically decided as intervals on a sorted domain. Multiple attribute

keys are just more complex cases, because their objects can be seen as geometric objects in a multidimensional space.

In order to better capture objects' content, thus allowing to enlarge the set of data types for which efficient search is possible, more recent approaches to indexing of multimedia, genomic, and many other non-traditional databases have considered the case where keys are not restricted to stay in a vector space, and only pair-wise objects' distances are possible to compute. Conveniently, such data are sometimes called *distance-only data*. For instance, both the Hausdorff distance over sets of n-dimensional points [7] and the Levenshtein (*edit*) distance over strings [6] are relevant examples of distance *metrics* for which no effective vectorial representation is possible for the indexed key domains (sets and strings, respectively). Such distance-based approach to key comparison, which subsumes the case of multi-dimensional keys (typically compared using the Euclidean distance), has lead to generalise the notion of *similarity queries* and resulted in the design of so-called *metric index trees*, which organise objects from a generic *metric space* by only considering their relative distances. Although several specific designs have been proposed so far (see [3, 2, 1, 4]) the algorithms devised for partitioning and organising objects are only based on heuristic criteria, for which no theoretical justification is given. We believe that the basic reason for this unpleasant situation is the absence of a clear definition of the *region proximity* notion and its quantification for generic metric spaces. In other terms, the question is: *given two regions of a metric space, can we "measure" how "close" these regions are?*

The major aim of this article is to fill the above gaps by proposing a specific solution to the "proximity problem" for arbitrary metric spaces. In particular, we define the problem in Section 2 and develop formulas for estimating the proximity of metric *ball* regions in Section 3. The theoretical results are verified by experiments in Section 4. This article concludes in Section 5.

2 The problem

Suppose there is a *metric space* $\mathcal{M} = (\mathcal{D}, d)$, defined by a domain of objects, $\mathcal{D}$, (i.e. the *keys* or indexed *features*) and by a total (distance) function, d, which satisfies for each triple of objects $O_x, O_y, O_z \in \mathcal{D}$ the following properties:

(i) $d(O_x, O_y) = d(O_y, O_x)$ (*symmetry*)

(ii) $0 < d(O_x, O_y) < \infty, O_x \neq O_y$ and $d(O_x, O_x) = 0$ (*non negativity*)

(iii) $d(O_x, O_y) \leq d(O_x, O_z) + d(O_z, O_y)$ (*triangle inequality*)

Considering such metric space, partitions can be defined as *regions* satisfying specific constraints.

Definition 2.1 *A region* $\mathcal{R} = \{O \in \mathcal{D} \mid C_{\mathcal{R}}(O)\}$ *is the set of objects of* $\mathcal{D}$ *which satisfy the constraint* $C_{\mathcal{R}}(\cdot)$. □

Obviously, two regions of the same metric space can have significantly different relative "positions". They can be quite far from each other, they can overlap, or one region can even be included into the other one. Since the quantification of such phenomenon is not trivial, it poses a not only theoretically interesting but mainly very practical *region proximity* problem.

Problem 2.1 *Given two regions* $\mathcal{R}_x$ *and* $\mathcal{R}_y$, *defined over the same metric space* $\mathcal{M}$, *determine the proximity* $X(\mathcal{R}_x, \mathcal{R}_y)$ *of these regions.* □

Notice that we are interested in proximity of regions of generic metric spaces, thus properties of special cases, such as the vector space co-ordinates, are not taken into account. Consequently, no volume of a region can be computed, because no formula for computing the volume of a region of a generic metric space exists.

Inspired by [8], where proximity measures for vector spaces are discussed, we propose to consider our proximity measure as a *chance* that a third region, called the *query region*, can share objects with both the compared regions, as the following definition formalises.

Definition 2.2 *The proximity $X(\mathcal{R}_x, \mathcal{R}_y)$ of regions $\mathcal{R}_x$ and $\mathcal{R}_y$ is the probability that a randomly chosen query region $\mathcal{Q}$ – where $\mathcal{R}_x, \mathcal{R}_y$, and $\mathcal{Q}$ are regions of the same metric space $\mathcal{M}$ – contains objects also found in both $\mathcal{R}_x$ and $\mathcal{R}_y$, i.e. $\exists O_i, O_j \mid O_i \in \mathcal{R}_x, O_j \in \mathcal{R}_y$ and $O_i, O_j \in \mathcal{Q}$.* □

To this purpose, the "classical" approach to probability theory suggests to consider proximity as the ratio of the number of cases in which a randomly chosen query region can find qualifying objects in both $\mathcal{R}_x$ and $\mathcal{R}_y$ regions, I, to the total number of possible query region occurrences, T. Specifically, the general form of the metric space proximity is

$$X(\mathcal{R}_x, \mathcal{R}_y) = \frac{I}{T} \tag{1}$$

2.1 Ball Regions

Up to now, we have not considered any specific type of regions. However, in order to come out with a solution which would satisfy the above requirements, let us concentrate on the *ball* regions.

Definition 2.3 *A ball $\mathcal{B}_x = \mathcal{B}_x(O_x, r_x) = \{O_i \in \mathcal{D} \mid C_{\mathcal{B}_x}(O_i) = d(O_x, O_i) \leq r_x\}$ is the region, determined by a centre $O_x \in \mathcal{D}$ and a radius $r_x \geq 0$, defined as the set of objects in $\mathcal{D}$ for which the distance to O_x is less than or equal to r_x.* □

Balls are the simplest region types which can be defined in a metric space, and as such, balls are more amenable to effective analysis. For instance, in order to see if two balls, $\mathcal{B}_x, \mathcal{B}_y \subseteq \mathcal{D}$, overlap, i.e. there can exists O_i which belongs to both $\mathcal{B}_x$ and $\mathcal{B}_y$, it is sufficient to check if the sum of their radii is greater than or equal to the distance between the balls' centres, specifically

$$\mathcal{B}_x \cap \mathcal{B}_y \neq \emptyset \iff r_x + r_y \geq d(O_x, O_y)$$

Note that, according to Definition 2.2, when $r_x + r_y < d(O_x, O_y)$ (thus $\mathcal{B}_x$ and $\mathcal{B}_y$ do not intersect each other) the value of $X(\mathcal{B}_x, \mathcal{B}_y)$ can still be positive, since it also depends on the query regions considered. On the other hand, it is quite intuitive that, for given radii values, the proximity of $\mathcal{B}_x$ and $\mathcal{B}_y$ should increase if $d(O_x, O_y)$ goes down (the two balls' centres get closer). Similarly, $X(\mathcal{B}_x, \mathcal{B}_y)$ should increase if, for a given $d(O_x, O_y)$, the sum $r_x + r_y$ grows. To summarise, we can say, with a slight abuse of terminology, that the proximity grows with the "size" of the regions' intersection.

In order to go beyond a purely qualitative analysis, we consider the relevant case where query regions are balls too. In this case, each query region $\mathcal{Q}$ is univocally identified by a *query key*, Q, and a query radius, r, thus $\mathcal{Q} = \mathcal{Q}(Q, r)$. A particular case arises when $r = 0$, which corresponds to *point* queries.

When only queries with a fixed radius value, r, are considered, we call this proximity as the *r-proximity* of $\mathcal{B}_x$ and $\mathcal{B}_y$, designated $X_r(\mathcal{B}_x, \mathcal{B}_y)$. Note that the 0-proximity, $X_0(\mathcal{B}_x, \mathcal{B}_y)$, means that proximity is evaluated by only considering point queries. As in the general case described by Equation 1, we can compute r-proximity as:

$$X_r(\mathcal{B}_x, \mathcal{B}_y) = \frac{I_r}{T_r} \tag{2}$$

When the ball queries under consideration have different radii, proximity should properly take all of them into account. There are basically two reasons why radii of query balls change:

Range queries leave the choice of a proper radius to the user. Though some radii values are, for a given metric, more likely than others – the response set should not typically be large - the radii are certainly not constant.

Nearest neighbour queries do not contain radii specifications at all. A radius for a given query object is changing dynamically, starting typically with a large radius and narrowing down its value according to the search space and the search strategy used. For details see e.g. [4].

In order to quantify proximity of two regions in such a situation, we simply view the radius of the query as a random variable, $\mathbf{r}$, and measure proximity by taking the expectation of r-proximities, that is:

$$X(\mathcal{B}_x, \mathcal{B}_y) = \int_r X_r(\mathcal{B}_x, \mathcal{B}_y) \cdot p_{\mathbf{r}}(r) dr \tag{3}$$

where $p_{\mathbf{r}}(r)$ is the probability that the $\mathbf{r}$ random variable takes the value r.

Before entering into technical details, concerning how proximity can effectively be evaluated under specific circumstances, we state some basic properties of our proximity measures.

Property 2.1 *For each pair of balls $\mathcal{B}_x(O_x, r_x)$ and $\mathcal{B}_y(O_y, r_y)$, the following properties hold:*

$$X_r(\mathcal{B}_x, \mathcal{B}_y) \leq X_{r'}(\mathcal{B}_x, \mathcal{B}_y) \iff r \leq r' \tag{4}$$

$$X_r(\mathcal{B}_x, \mathcal{B}_x) \geq X_r(\mathcal{B}_x, \mathcal{B}_y) \quad \forall r \geq 0 \tag{5}$$

$$X(\mathcal{B}_x, \mathcal{B}_x) \geq X(\mathcal{B}_x, \mathcal{B}_y) \quad \forall \{p_{\mathbf{r}}(r)\} \tag{6}$$

□

The first property asserts that the r-proximity of any two balls is not greater than their r'-proximity, if $r \leq r'$, whereas the second inequality states that the maximum r-proximity with respect to the $\mathcal{B}_x$ ball is obtained from $\mathcal{B}_x$ itself. This is also called the *self-r-proximity* of $\mathcal{B}_x$. Finally, 6 is an immediate consequence of 5, since r-proximity is not negative by definition.

3 Ball Proximity Measures

Consider a set of objects for which the distribution of distances between pairs of objects is uniform, and suppose that the maximum distance is $d_m < \infty$, thus consider a *bounded* metric space. Starting with the case of point queries, i.e. ball queries with $r = 0$, thus $\mathcal{Q} = \mathcal{Q}(Q, 0)$, we first show how proximity can be effectively computed for ball regions with identical (co-centric) and different centres. In Section 3.3, we generalise the approach to the case of range queries and non-uniform distance distributions.

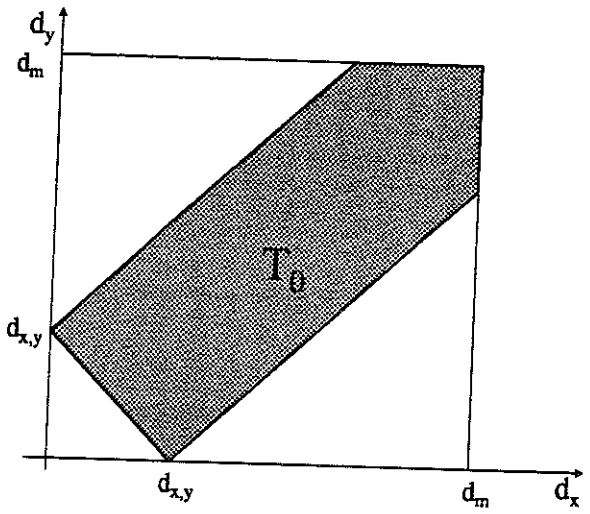

Figure 1: Query distance space

3.1 The proximity of co-centric balls

Given two balls $\mathcal{B}_x$ and $\mathcal{B}_y$, with $O_x \equiv O_y$ and $r_x \leq r_y$, in a metric space bounded by d_m, the distance of any possible query point Q from a generic object is in the range $[0, d_m]$. In fact, the quantification of proximity of co-centric balls can be regarded as a one-dimensional geometric problem where the total number of distinct query points is measured by the length of the line segment $[0, d_m]$, thus $T_0 = d_m$, and the number of query points intersecting both $\mathcal{B}_x$ and $\mathcal{B}_y$ is the length of the segment $[0, r_x]$, that is, $I_0 = r_x$.[1] Specifically, we can compute the 0-proximity as

$$X_0(\mathcal{B}_x, \mathcal{B}_y) = \frac{r_x}{d_m} \tag{7}$$

3.2 The proximity of balls with different centres

Provided $\mathcal{B}_x$ and $\mathcal{B}_y$ have different centres, i.e. $d(O_x, O_y) > 0$, the situation is a bit more complex. In particular, let Q be a randomly chosen object from $\mathcal{D}$. In order to simplify notation, we denote the distance between the O_x and O_y balls' centres as $d_{x,y}$, the distance $d(Q, O_x)$ as d_x, and the distance $d(Q, O_y)$ as d_y. Since $d_{x,y}$ is, for given $\mathcal{B}_x$ and $\mathcal{B}_y$, constant, thus independent of Q, a specific Q can be represented as a point in the two dimensional distance space (d_x, d_y), and the whole range of possible Q values is fully specified (restricted) by an area which satisfies the following space constraints.

Restriction 3.1 *Provided the maximum distance in $\mathcal{M}$ is d_m, the distance space (d_x, d_y), relative to given objects $O_x, O_y \in \mathcal{D}$, is restricted by: 1. $d_{x,y} \leq d_x + d_y \leq d_m$, 2. $d_x \leq d_{x,y} + d_y \leq d_m$, and 3. $d_y \leq d_{x,y} + d_x \leq d_m$.* □

This situation is illustrated in Figure 1 to indicate that the value space for the possible distances between a randomly chosen object Q and two ball centres O_x and O_y having distance $d_{x,y}$, is restricted by a polygon, the area of which corresponds to T_0, i.e. the total number of

[1] Notice that our assumption is that r_y is never smaller than r_x.

possible object Q values, which is evaluated by the following formula where, for the sake of brevity, we further simplify notation, and use x instead of d_x.

$$\begin{aligned} T_0 &= \int_0^{d_m - d_{x,y}} (d_{x,y} + x)dx + \int_{d_m - d_{x,y}}^{d_m} (d_m)dx - \int_0^{d_{x,y}} (d_{x,y} - x)dx - \int_{d_{x,y}}^{d_m} (x - d_{x,y})dx \\ &= d_{x,y}(2d_m - 1.5d_{x,y}) \end{aligned} \tag{8}$$

From Equation 8 it can be derived that T_0 increases with the distance between the balls' centres, $d_{x,y}$, up to a maximum value given by $2/3d_m^2$, which occurs when $d_{x,y} = 2/3d_m$. The fact that T_0, the measure of the number of (point) queries, depends on data, that is the distance between the compared ball centres $d_{x,y}$, is somewhat counter-intuitive, and requires some explanation. The key point is that T_0 is not an "absolute" measure, which is meaningless in a generic metric space, rather, it is a "subjective" measure which depends on the "observers", that is the ball centres O_x and O_y. When the distance between such objects changes, accordingly, the number of cases (queries) which the two points can distinguish has to change as well.

The proximity of $\mathcal{B}_x$ and $\mathcal{B}_y$ also depends on the balls' "volumes", as determined by the radii r_x and r_y, respectively. Indeed, to measure the number of query objects which intersect balls $\mathcal{B}_x$ and $\mathcal{B}_y$, we need to take the following additional constraints into account.

Restriction 3.2 *The distance space of objects belonging to both balls $\mathcal{B}_x$ and $\mathcal{B}_y$ is obtained by adding to Restriction 3.1 the conditions: 1. $d_x \leq r_x$ and 2. $d_y \leq r_y$.* □

In general, the number of objects which appear in the intersection, I_0, of the balls $\mathcal{B}_x$ and $\mathcal{B}_y$ is proportional to the area of a polygon defined by Restrictions 3.1 and 3.2. For a specific case, refer to Figure 2.

3.2.1 Numeric evaluation

In order to determine I_0, that is, the number of possible query objects appearing in the intersection of two balls with radii r_x and r_y, we suggest to use the following systematic approach:

$$I_0 = I_0^t - I_0^{x,y} - I_0^x - I_0^y \tag{9}$$

where: I_0^t is the total area constrained only by the radii r_x and r_y; $I_0^{x,y}$ is the part of I_0^t discarded by the $d_{x,y} \leq d_x + d_y$ restriction; I_0^x is the area of I_0^t discarded by the $d_x \leq d_{x,y} + d_y$ restriction; and I_0^y is the area of I_0^t discarded by the $d_y \leq d_{x,y} + d_x$ restriction. The specific formulas are the following:

$$I_0^t = r_x \cdot r_y \tag{10}$$

$$I_0^{x,y} = \frac{d_{x,y}^2}{2} - \frac{(\max\{0, d_{x,y} - r_x\})^2}{2} - \frac{(\max\{0, d_{x,y} - r_y\})^2}{2} \tag{11}$$

$$I_0^x = \frac{(\max\{0, r_x - d_{x,y}\})^2}{2} - \frac{(\max\{0, r_x - r_y - d_{x,y}\})^2}{2} \tag{12}$$

$$I_0^y = \frac{(\max\{0, r_y - d_{x,y}\})^2}{2} - \frac{(\max\{0, r_y - r_x - d_{x,y}\})^2}{2} \tag{13}$$

Notice that Equations 12 and 13 are symmetric in the balls' radii r_x and r_y.

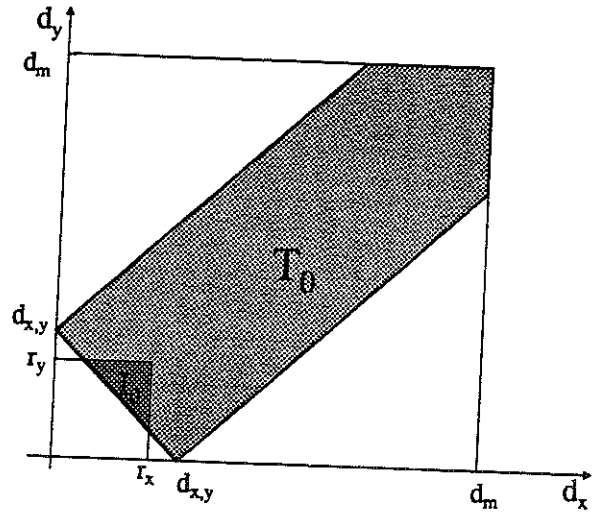

Figure 2: The intersection of ball regions when $r_x + r_y \geq d_{x,y}$, $r_x < d_{x,y}$, and $r_y < d_{x,y}$

3.3 Range Queries and non-uniform distance distributions

Till now, we have assumed that distances produced by the function $d(\cdot,\cdot) \in [0, d_m]$ follow the uniform distribution, i.e. the probability of any value to occur is constant. However, such assumption is not realistic, because experience says that distances between objects, for example in high-dimensional spaces, tend to have very skewed distributions.

Given two ball regions $\mathcal{B}_x$ and $\mathcal{B}_y$, let $p_{\mathbf{x}}(x)$ and $p_{\mathbf{y}}(y)$ denote the probabilities that the distances from a random query object to the centres of balls $\mathcal{B}_x$ and $\mathcal{B}_y$ are x and y, respectively. Then, T_r ($r \geq 0$) can be computed by integrating over x from 0 to d_m and over y with variable limits $y_1(x)$ and $y_2(x)$, which are determined for each value of x by Restriction 3.1, that is:

$$T_r = \int_{x=0}^{d_m} \int_{y=y_1(x)}^{y_2(x)} p_{\mathbf{y}}(y) p_{\mathbf{x}}(x)\, dy\, dx \tag{14}$$

Similarly, I_r can be computed by integrating over x from 0 to $r_x + r$ and over y with variable limits $y'_1(x)$ and $y'_2(x)$, as determined by Restrictions 3.1 and 3.2.

$$I_r = \int_{x=0}^{r_x+r} \int_{y=y'_1(x)}^{y'_2(x)} p_{\mathbf{y}}(y) p_{\mathbf{x}}(x)\, dy\, dx \tag{15}$$

It is a fact that range (rather than exact match) queries are the typical case when dealing with data from generic metric spaces, which implies that point queries are rarely used in practice. However, as shown in [10], range queries can easily be transformed into point queries just by modifying the r_x and r_y radii. Specifically,

$$X_r(\mathcal{B}_x(O_x, r_x), \mathcal{B}_y(O_y, r_y)) = X_0(\mathcal{B}_x(O_x, r_x + r), \mathcal{B}_y(O_y, r_y + r)) \tag{16}$$

4 Experimental evaluation

The results of the analytical formulas derived in previous sections have been compared with actual proximity of regions as observed on real data for point queries. In order to evaluate and

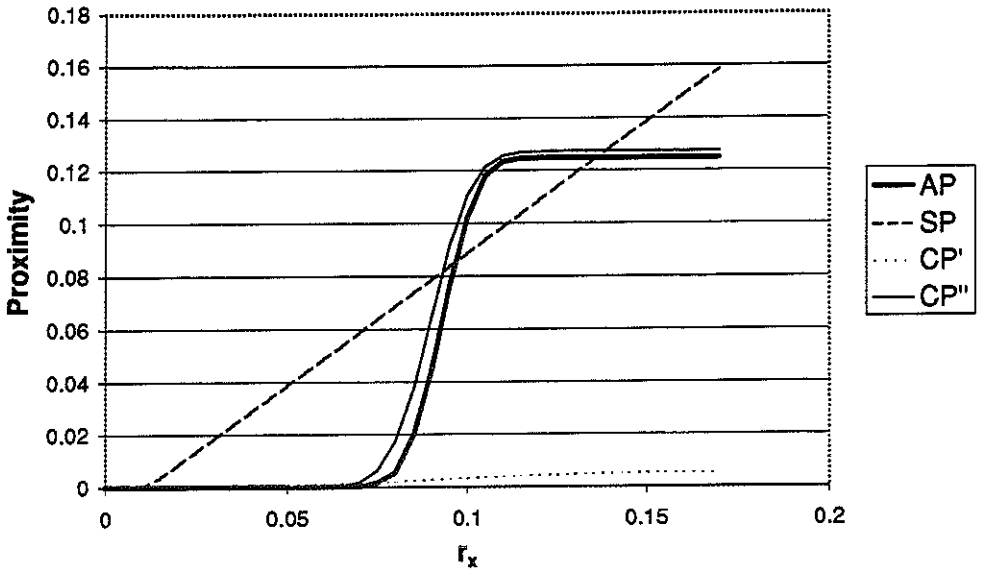

Figure 3: Proximity vs r_x. UV file. $r_y = 0.08$, $d_{x,y} = 0.091$

compare the proposed proximity measures, we have used two qualitatively different files of 45-dimensional vectors, each of them of size $n = 10,000$. The first file, designated as UV, contains synthetic data, that is, vectors uniformly distributed on the 45-dimensional unit hyper-cube. The second file, designated as HV, represents colour histograms of images and has been chosen as a representative of real-life files.

Given two balls, $\mathcal{B}_x(O_x, r_x)$ and $\mathcal{B}_y(O_y, r_y)$, the purpose of the experiments is to verify the proposed formulas for computing proximity with the *actual proximity* AP. The value of AP is determined as the fraction of objects in the file that satisfy the condition $d_x \leq r_x$ and $d_y \leq r_y$, where $d_x = d(O_x, O_i)$ and $d_y = d(O_y, O_i)$. Specifically, we compare AP with the following three estimates of the proximity:

- the *simple proximity* SP; this is an easy to compute heuristic estimate, often used in current implementations, defined as:

$$SP(\mathcal{B}_x, \mathcal{B}_y) = \begin{cases} 0 & \text{if } r_x + r_y < d(O_x, O_y) \\ 2\min\{r_x, r_y\} & \text{if } r_x > r_y + d(O_x, O_y) \\ r_x + r_y - d(O_x, O_y) & \text{otherwise} \end{cases} \quad (17)$$

  Obviously, the measure depends on the distance of the balls' centres and on the size of their radii – the closer the centres, for given radii values, are, the higher SP is.

- two types of *complex proximity*, CP' and CP'', the first obtained by assuming that the distribution of distances between objects is *uniform* and the second calculated by considering that distances follow a *normal distribution*. Formulas introduced in Section 3 have been used to compute CP' and CP''.

It should be observed that the values of AP, CP' and CP'' are in the range $[0, 1]$ and that their absolute values can be directly compared; this is not the case for SP, which implies that only the trends of SP and AP can be compared.

Figure 3 shows the proximity (AP, SP, CP', and CP'') as a function of r_x, with constant $r_y = 0.08$, for the UV file. The considered mean μ and the variance σ of the objects' distribution,

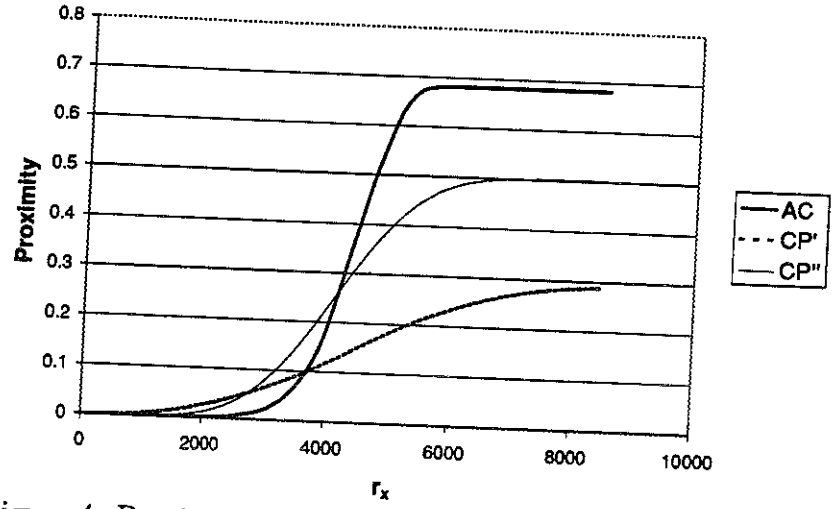

Figure 4: Proximity vs r_x. HV file. $r_y = 4122$, $d_{x,y} = 4422$

obtained by random sampling from UV, are $\mu = 0.0935$ and $\sigma = 0.01$, respectively. The actual proximity is zero up to $r_x \approx 0.06$ and it grows up to its maximum, obtained for $r_x \approx 0.11$. The simple proximity SP has a behaviour not comparable to that of AP, since it grows linearly from 0 to $2\min\{r_x, r_y\}$ for $d_{x,y} - r_y \geq r_x \geq r_y + d_{x,y}$. The estimates obtained from CP' are quite different from those of AP, whereas CP'' demonstrates a behaviour as well as absolute proximity values well in accordance with the values of AP. This simply demonstrates how specific distance distributions are important to consider.

Provided the relative error with respect to AP, obtained when the proximity is estimated through CP (where CP is either CP' or CP''), is defined as

$$\epsilon = \frac{|\, AP - CP \,|}{AP},$$

the observed values for CP'' were approximately 0.02 for most of the values of r_x, while for CP', the value of ϵ was around 0.9. Indeed, vectors of 45 dimensions have a distribution of distances which is far from being uniform (even if the objects are uniformly distributed), rather it is approximated quite well by a normal distribution.

As Figure 4 demonstrates, a similar behaviour is also observed for experiments performed on the HV file. Figure 4 does not report on SP which takes values too large to be shown in the graph: SP is equal to 0 for $r_x < d_{x,y} - r_y$, then it grows linearly up to $SP = 2 \times r_y = 8244$ obtained for $r_x \geq r_y + d_{x,y}$. In general, the correlation between the actual and the estimated values of the proximity is not as good as for file UV. Such result is mainly attributed to the fact that the proper distance distribution parameters of this (real life) file are more difficult to obtain by sampling. However, the relative error is still limited to 0.25, so that CP'' can be used as a good measure of the proximity also in this case.

5 Conclusions

Motivated by the urgent needs of theoretical foundations for partitioning and allocating metric data files, the problem of proximity for metric space regions has been elaborated. The proximity of two regions has been defined as the probability that a randomly chosen query region finds

qualifying objects in both the regions. This problem has been studied in depth for the ball regions, and formulas for estimating the proximity for point and range queries have been outlined, respecting both the uniform and actual distance distributions.

The proposed machinery has been tested on two different data sets, and obtained results have confirmed the validity of this approach. More experimental evaluations can be found in [10], where the proximity measures are also considered as suitable tools able to compute the selectivity and region similarity features, which have got important applications in storage structure designs. However, what seems to be central is the necessary knowledge of the actual distance distribution - the better the knowledge about the distance distribution is, the higher the precision of our region proximity estimate can be.

As a first step towards applications, our proximity measure has been applied to devise declustering algorithms for the parallel M-tree index [11]. Preliminary tests show that the proposed approach can be indeed valuable, mainly considering the stability it can achieve when considering the *speedup* and *scaleup* obtained from the parallelisation of the index.

Future research should concentrate on proximity problems of non-ball regions as well as on proximity of multiple regions. More effort is also to be spent on other applications.

References

[1] T. Bozkaya and M. Ozsoyoglu. Distance-based indexing for high-dimensional metric spaces. *ACM SIGMOD*, pp.357-368, Tucson, AZ, May1997.

[2] S. Brin. Near neighbour search in large metric spaces. In *Proceedings of the 21st VLDB International Conference*, pp. 574–584, Zurich, Switzerland, September 1995.

[3] T. Chiueh. Content-based image indexing. In *Proceedings of the 20th VLDB International Conference*, pages 582–593, Santiago, Chile, September 1994.

[4] P. Ciaccia, M. Patella, and P. Zezula. M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. Proceedings of the *23rd VLDB Conference*, Athens, Greece, 1997, pp. 426-435.

[5] A. Guttman. R-trees: A dynamic index structure for spatial searching. In *Proceedings of the 1984 ACM SIGMOD International Conference on Management of Data*, pages 47–57, Boston, MA, June 1984.

[6] P.A.V. Hall and G.R. Dowling. Approximate String Matching. *ACM Computing Surveys* 12(4):381-402, December 1980.

[7] D.P. Huttenlocker, G.A. Klanderman, and W.J. Rucklidge. Comparing Images Using the Hausdorff Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850-863, September 1993.

[8] I. Kamel and C. Faloutsos. Parallel R-trees. *Proc. of the ACM SIGMOD Conf.*, June 1992, pp. 195-204.

[9] J.K. Uhlmann. Satisfying general proximity/similarity queries with metric trees. *Information Processing Letters*, 40(4):175–179, November 1991.

[10] P. Zezula, P. Savino, P. Ciaccia, and F. Rabitti. On the Region Proximity in Metric Spaces. Technical Report of the ESPRIT LTR HERMES Project, No. 9141.

[11] P. Zezula, P. Savino, F. Rabitti, G. Amato, and P. Ciaccia. Processing M-trees with Parallel Resources. In *Proceedings of the Eighth International Workshop on Research Issues in Data Engineering: Continuous-Media Databases and Applications - RIDE'98*, February, 1998, Orlando, Florida, pp. 147-154.