Separable Splits of Metric Data Sets

Vlastislav Dohnal[1], Claudio Gennaro[2], Pasquale Savino[2], and Pavel Zezula[1]

[1] Masaryk University, Brno, Czech Republic
xdohnal@fi.muni.cz, zezula@fi.muni.cz
[2] IEI-CNR, Pisa, Italy
gennaro@iei.pi.cnr.it, savino@iei.pi.cnr.it

Abstract. In order to speedup retrieval in large collections of data, index structures partition the data into subsets so that query requests can be evaluated without examining the entire collection. As the complexity of modern data types (such as image, video, or audio features) grows, the traditional partitioning techniques based on total ordering of data can not typically be applied. We consider the problem of partitioning data collections from generic metric spaces, where total ordering of objects does not exists, and where only distances between pairs of objects can be determined. We study the elementary type of partitioning that splits a given collection into two well-separated subsets, allowing some objects to be excluded from the partitioning process. Five implementation techniques of separable splits are proposed and proved for correctness. The first two are simple extensions of the known ball partitioning and the generalized hyperplane approaches, the third is an advanced hyperplane partitioning. The additional two techniques are completely original and are based on the elliptic and pseudo-elliptic geometric strategies. Effectiveness of all techniques is evaluated in terms of their ability to equalize the separable set sizes, and to minimize the number of excluded objects. Proposed techniques are evaluated on three large data files.

1 Introduction

Searching has always been one of the most prominent data processing operations because of its useful purpose of delivering required information efficiently. However, exact match retrieval, typical for traditional databases, is not sufficient or feasible for data types of present digital age. What seems to be more useful, if not necessary, is to base the search paradigm on a form of *proximity*, *similarity*, or *dissimilarity* of query and data objects. Roughly speaking, objects that are near to a given query object form the query response set. In this place, the notion of mathematical *metric space* [Ke55] provides a useful abstraction for *nearness*.

Similarity search is needed as a fundamental computational task in a variety of application areas, including multimedia information retrieval, data mining, pattern recognition, machine learning, computer vision, genome databases, data compression, and statistical data analysis. This problem, originally mostly studied within the theoretical area of computational geometry, is recently attracting more and more attention in the database community, because of the increasingly growing needs to deal with large, often distributed, volume of data.

Several storage structures, such as [Ch94,Br95,CPZ97,BO99,TTS$^+$00], have been designed to support efficient similarity search execution over large collections of metric data where only a distance measure for pairs of objects is possible to quantify. However, the performance is still not satisfactory. Though all of the indexes are trees, many tree branches have to be traversed to solve a query, because similarity queries are basically range queries that typically cover data from several tree nodes or some other content-specific partitions. In principle, if many data partitions need to be accessed for a query region, they are not *contrasted* enough and such partitioning is not useful from the search point of view.

Recently, the *excluded middle vantage point* strategy for partitioning of metric data have been proposed to develop two similarity search index structures: (1) a *forest of* one path search *trees*, [Yi99], and (2) a *similarity hash structure*, [Ze00]. The idea is to split a data set into two subsets that are separable up to a predefined value of a query search radius ρ. Such objective is achieved by excluding objects that would not allow such separation – excluded objects form the third part of the split and can recursively be processed again by another split.

Functions, supporting this kind of split, are called the *ρ-split functions*. Experimental results with structures based on this idea look to be very promising, but additional investigation of such strategy is needed. These ideas have the potential to make a very good contribution to the similarity search technology, given that good ρ-split functions can be designed. Thus it is imperative to investigate the problem and create clear guidelines for using such functions in specific systems and applications.

In this article, we define and analyze five different ρ-split functions. In order to compare, we consider the uniformity of split and the minimization of the number of excluded objects as two basic measures. All functions are analyzed and tested on one synthetic and two real-life data files. In the rest of the paper, we summarize the necessary background and specify our research objectives in Section 2. In Section 3, we extend and evaluate the traditional metric partitioning functions into the form of ρ-split functions. In Section 4, we investigate another two ρ-split functions. Conclusions and future research directions can be found in Section 5.

2 Background and Objectives

A *metric space* $\mathcal{M} = (\mathcal{D}, d)$ is defined by a domain of objects $\mathcal{D}$ (i.e. the *keys* or indexed features) and by a total (distance) function d – a *non negative* and *symmetric* function that satisfies the *triangle inequality* $d(O_x, O_y) \leq d(O_x, O_z) + d(O_z, O_y)$, $\forall O_x, O_y, O_z \in \mathcal{D}$. We assume that the maximum distance never exceeds d_{MAX}, thus a *bounded metric space* is considered. The similarity search problem, in general, can be sketched as follows.

Problem 1. Let $\mathcal{D}$ be a set, d a distance measure on $\mathcal{D}$, and $(\mathcal{D}, d)$ a metric space. Given a set $X \subseteq \mathcal{D}$ of n elements, preprocess or structure the data so that similarity queries can be answered in an efficient way.

Though more different types of similarity queries exist, we concentrate on the *similarity range* queries, i.e. the most fundamental type of queries. Such query is defined as a *region* $\mathcal{R}(x, r_x)$, where $x \in \mathcal{D}$ is called the region's *center* and r_x is the distance defining its *radius*. Objects *covered* by this region form the set $\{x_i \mid x_i \in \mathcal{D} \wedge d(x, x_i) \leq r_x\}$, and the *query response* is the set $\{x_i \mid x_i \in X \wedge d(x, x_i) \leq r_x\}$. Finally, we assume *similarity range queries* with radii up to a predefined value $\rho \leq d_{MAX}$.

Now, consider several sets of objects that are separable up to ρ.

Definition 1. *Sets of objects $\{X_1, X_2, \ldots, X_m \mid X_i \cap X_j = \emptyset\ \forall i \neq j\}$ are separable up to ρ, if $\forall x, y \mid x \in X_i, y \in X_j, i \neq j$, $d(x, y) > 2\rho$.* □

Figure 1(a) shows an example of four 2-dimensional partitions that cover object sets separable up to ρ. Notice that given a query region $\mathcal{R}(x, r_x)$, no more than one separable set can contain objects that are covered by this region. The proof is obvious. Suppose $x_i \in X_i$ is covered by $\mathcal{R}(x, r_x)$, i.e. $d(x, x_i) \leq r_x$. Then the maximum distance from x_i to any other object covered by this region, say x_j, is constrained by $d(x_i, x_j) \leq 2r_x$. Since $r_x \leq \rho$, x_j cannot be from any other subset but X_i, because the subsets are separable up to ρ – all objects from subsets different from X_i have distance $d(x_i, x_j) > 2\rho$, thus do not belong to the response set. In order to achieve such partitioning, objects from the gray area must be, from this partitioning process, excluded. Obviously, a query region can constrain data objects from exactly one partition, which is demonstrated in Figure 1(b) by the function $e^{\rho}(x, r_x) = true$.

Now consider the elementary case of partitioning, called the ρ-split.

Definition 2. *The ρ-split of X, $S^{\rho}(X) = \{X_0, X_1, X_2\}$, is a disjoint separation of all objects from X into subsets $X_0 \cup X_1 \cup X_2 = X, X_i \cap X_j = \emptyset\ \forall i \neq j$, such that X_0, X_1 are subsets separable up to ρ.* □

In fact, the set X_2 contains elements that cannot be accommodated in X_0 or in X_1 so that these two sets remain separable up to ρ. The *non-separable* elements are logically *excluded*, and for practical purposes stored in X_2. In order to achieve ρ-split, we need a function to decide which subset of the ρ-split a specific object, say x, belongs to.

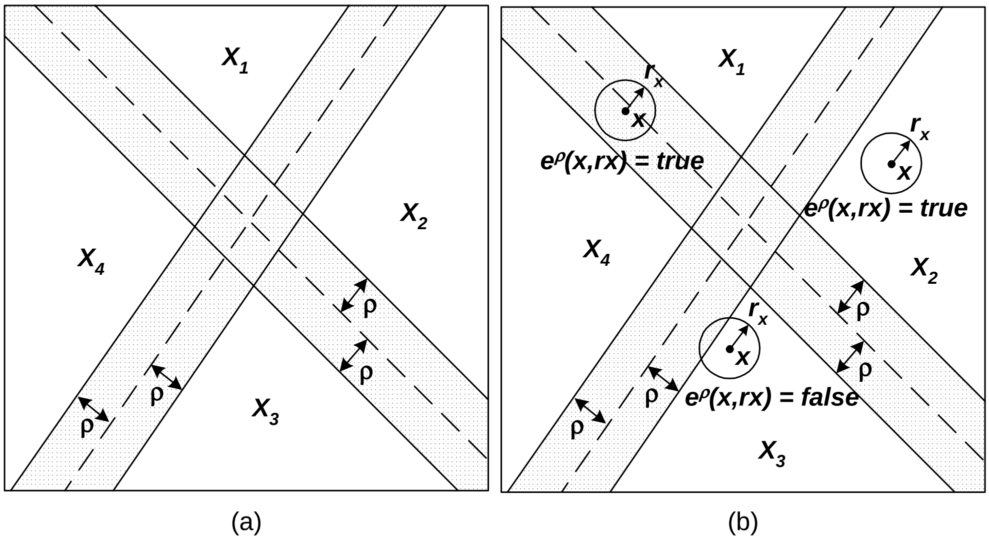

Fig. 1. Example of 2-dimensional partitions separable up to ρ (a), and exclusive containment function behavior (b).

Definition 3. *Given a ρ-split of X $S^\rho(X) = \{X_0, X_1, X_2\}$, we define the ρ-split function, $s^\rho(x) = \{0, 1, 2\}$, such that $\forall x$ $s^\rho(x) = i \Leftrightarrow x \in X_i$.* □

By definition, $s^0(x) = \{0, 1\}$ becomes a binary function, since $s^0(x)$ can never evaluate to the value 2. Notice that $s^\rho(x)$ only depends on specific object x. It does not assume the knowledge of the entire indexed set X, thus allows to build $S^\rho(X)$ in incremental way.

The function is symmetric such that given two arbitrary objects $x_0 \in X_0, x_1 \in X_1$ from $S^\rho(X)$ with $\rho > 0$, $s^0(x) = 0 \Rightarrow d(x, x_1) > \rho$ and $s^0(x) = 1 \Rightarrow d(x, x_0) > \rho$. In other words, the symmetry property helps to decide, which of the separable sets in $S^\rho(X)$, i.e. X_0 or X_1, needs not be accessed to solve a query defined by region $\mathcal{R}(x, r_x)$. If $s^0(x) = 0$, the distance from x to an object in X_1 of $S^\rho(X)$ is greater than ρ, (Definition 3). That means, X_1 cannot contain any qualifying object for $\mathcal{R}(x, r_x)$, because the distance from x to a qualifying object cannot exceed $r_x \leq \rho$. Due to the symmetry property of ρ-split functions, similar arguments apply in case $s^0(x) = 1$. In summary, if x is the $\mathcal{R}$'s center, the sets that might contain objects belonging to $\mathcal{R}$ are $X_{s^0(x)}$ and X_2.

2.1 Research Objectives and Evaluation Strategies

It is intuitively clear that many ρ-split functions can be defined, and that some of the functions can be more convenient than the others, which might also depend on the data. On the extreme basis, a ρ-split function can allocate all objects in one of the subsets (partitions), keeping the others empty. Of course, there is no meaning in such split, whatever subset would contain the result. Considering the fact that objects from partitions produced by a ρ-split function can recursively become the subject of another ρ-splits to obtain even smaller partitions, we can say that a good ρ-split function should aim at maximizing the following two objective measures:

balancing – Separable sets should be of (approximately) equal size to avoid a skew in object allocation for separable sets. In principle, there are two reasons to achieve such objective. First, since one of the separable subsets can always be neglected for any kind of query, balanced partitions should guarantee good performance on the average – large sets have high probability of access, provided the distribution of query objects follows the distribution of searched data, which is usually true. Second, recursive partitioning of unbalanced sets can only increase the data allocation skew.

minimization – The exclusion set should be small, because it is the set that cannot typically be eliminated from the search point of view.

In order to verify the validity of our analytic work, we use three data sets: one synthetic file of 45-dimensional uniformly distributed vectors and two real-life files of color-image features. In the first feature file, designated as HV1, color features are represented as 9-dimensional vectors containing the *average*, *standard deviation*, and *skewness* of pixel values for each of the red, green, and blue channels. An image is divided into five overlapping regions, each one represented by a 9-dimensional color feature vector. That results in a 45-dimensional vector as a descriptor of one image. The second feature file, called HV2, contains color histograms represented in 32-dimensions. This data set was obtained from the UCI Knowledge Discovery in Databases Archive ([Bay99]). The color histograms are extracted from the Corel image collection as follows: the HSV space is divided into 32 subspaces (32 colors: 8 ranges of hue and 4 ranges of saturation). The value in each dimension of such vector is the density of the corresponding color in the entire image.

The distance function, used to compare two feature vectors, is always the Euclidean (L_2) distance. Moreover, the synthetic file, designated UD, contains 100 000 objects, the HV1 file 11 000 objects, and the HV2 file 65 000 objects.

3 Basic Metric Partitioning Principles as ρ-Split Functions

Partitioning, in general, is one of the most fundamental principles of any storage structure, aiming at dividing the search space into sub-groups, so that once a query is given, only some of these groups are searched. Given a set $X \subseteq \mathcal{D}$ of objects in metric space $\mathcal{M} = (\mathcal{D}, d)$, Uhlmann [Uh91] has defined the *ball partitioning* and the *generalized hyperplane partitioning* techniques that can briefly be characterized as follows:

ball – partitioning brakes the set X into subsets X_0 and X_1 using spherical cuts with respect to *vantage point* $o_v \in \mathcal{D}$, chosen arbitrarily. Let $d_m \leftarrow$ median of $\{d(o_i, o_v) \mid \forall o_i \in X\}$. Then all $o_j \in X$ are distributed in X_0 and X_1 according to the following rules:

- $X_0 \leftarrow \{o_j \mid d(o_j, o_v) \leq d_m\}$
- $X_1 \leftarrow \{o_j \mid d(o_j, o_v) \geq d_m\}$

The redundant conditions $\leq$ and $\geq$ assure balance when the median value is not unique. This is accomplished by assigning each element on the partition to one of the subsets in arbitrary, but balanced, fashion.

generalized hyperplane – partitioning brakes the set X into subsets X_0 and X_1 using two, arbitrarily chosen, reference points (objects) $o_0, o_1 \in \mathcal{D}$. Then all $o_j \in X$ are distributed in X_0 and X_1 according to the following rules:

- $X_0 \leftarrow \{o_j \mid d(o_0, o_j) \leq d(o_1, o_j)\}$
- $X_1 \leftarrow \{o_j \mid d(o_1, o_j) \leq d(o_0, o_j)\}$

In case $d(o_0, o_j) = d(o_1, o_j)$, o_j is placed in smaller from the X_0 and X_1 sets to reduce a possible skew, though a balanced split cannot be guaranteed, in general.

In the following, we define two ρ-split functions as an extension of the basic partitioning principles and prove their correctness in the sense of Definition 1. Specifically, we define the vantage point (ball) ρ-split and the hyperbolic ρ-split as a modification of the generalized hyperplane. We also define an extension of the hyperbolic approach, that is the modified hyperbolic ρ-split function. Finally, we experimentally evaluate these functions and compare their properties.

3.1 Vantage point ρ-split

Vantage point ρ-split function $vps^\rho(x, o_0)$ uses one routing (reference) object o_0 and a distance δ for partitioning the data file into three subsets X_0, X_1, and X_2. The result of the function gives the index of the set, which the point x belongs to (in sense of Definition 3). The function is defined as follows:

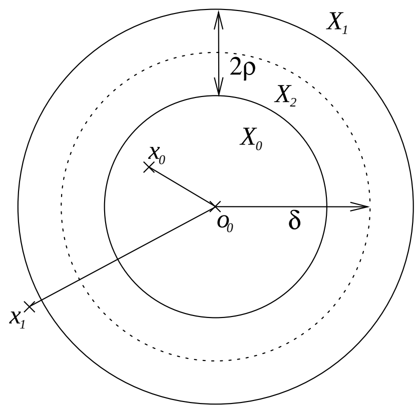

Fig. 2. The vantage point ρ-split function

$$vps^{\rho}(x, o_0) = \begin{cases} 0 \text{ if } d(x, o_0) < \delta - \rho, \\ 1 \text{ if } d(x, o_0) > \delta + \rho, \\ 2 \text{ otherwise.} \end{cases} \tag{1}$$

For convenience, the situation is illustrated in Figure 2.

The following proof shows that the vps^{ρ}-split function behaves in the desired way, i.e. the distance between arbitrarily picked objects x_0 and x_1 from subsets X_0 and X_1 is greater than 2ρ.

Proof. We want to show, that: $\forall x_0 \in X_0, \forall x_1 \in X_1 : d(x_0, x_1) > 2\rho$. From the definition of the vantage point ρ-split function, we have

$$d(x_0, o_0) < \delta - \rho \tag{2}$$

$$d(x_1, o_0) > \delta + \rho \tag{3}$$

If we use the triangle inequality, we get

$$d(x_0, x_1) + d(x_0, o_0) \geq d(x_1, o_0)$$

Using 2 and 3, we obtain

$$d(x_0, x_1) + (\delta - \rho) > (\delta + \rho)$$

$$d(x_0, x_1) > 2\rho$$

□

The parameter $\rho \geq r_x$ of the vantage point split function depends on the application and limits the size of the query search radius r_x. The parameter $\delta > \rho$ is used for balancing separable sets as follows.

Balancing: First notice that the balance property doesn't depend on a choice of the routing point, so balanced separable sets can always be achieved. If we choose the parameter $\rho = 0$, the parameter δ can be set to a median of all distances from the routing point o_0 to the other points in a data set. Then the subsets X_0 and X_1 are balanced (the ratio of their cardinalities is approximately equal to one), because it is the case of ball partitioning. However, when $\rho > 0$ is used, which is our objective, the median does not guarantee the balanced split, because the split depends on distance distribution of objects with respect to o_0. But also in this case, the proper value of δ can experimentally be found using e.g. the *interval splitting* method.

This method is based on iterative bisecting an interval of possible values of δ. Initially, boundaries of the interval are set to the minimum and the maximum values of δ parameter[1]. Then the ρ-split function is iteratively

[1] For the VPS function, the minimum value of δ is zero and the maximum is d_{MAX}.

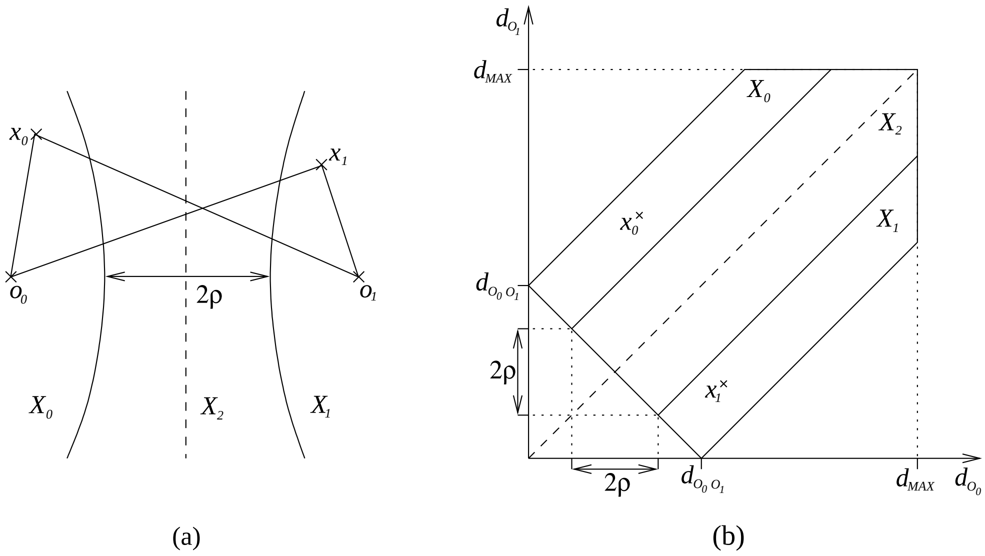

Fig. 3. The hyperbolic ρ-split function in the data space point of view (a) and in the distance point of view (b).

computed with δ as an average of limits of the current interval. This is repeated until balanced subsets are achieved. In every iteration, the interval's boundaries are updated – the lower or the higher boundary is set to the current value of δ depending on the cardinalities of the subsets X_0 and X_1. This approach is also used in the other ρ-split functions defined later.

3.2 Hyperbolic ρ-split

The hyperbolic ρ-split function $hps^\rho(x, o_0, o_1)$ uses two objects o_0 and o_1 for partitioning the data file into three subsets X_0, X_1, and X_2, as follows:

$$hps^\rho(x, o_0, o_1) = \begin{cases} 0 \text{ if } d(x, o_1) - d(x, o_0) > 2\rho, \\ 1 \text{ if } d(x, o_0) - d(x, o_1) > 2\rho, \\ 2 \text{ otherwise.} \end{cases} \tag{4}$$

The situation is illustrated from the data space point of view in Figure 3(a). We can also represent the situation from the distance point of view considering areas of a two-dimensional distance space, as available in Figure 3(b). However, the second alternative may require more explanation, which in principle is easy. On the x axis, we have a distance, of an arbitrary object of the metric space, to the reference object o_0; while on the y axis, we have distance to the reference object o_1. Having two fixed reference objects o_0 and o_1, a distance to any other possible metric object is not only constrained by the maximum distance d_{MAX}, but it must also satisfy the triangle inequality, provided the distance $d(o_0, o_1)$ is given. This in fact constrains possible combinations of distances and results in the figure in the largest polygon of five sides.

The following proof shows that the hps^ρ-split function behaves in the desired way, i.e. the distance between arbitrarily picked objects x_0 and x_1 from subsets X_0 and X_1 is greater than 2ρ.

Proof. To be proved: $\forall x_0 \in X_0, \forall x_1 \in X_1 : d(x_0, x_1) > 2\rho$.
By the triangle inequality, we have

$$d(x_0, o_0) + d(x_0, x_1) \geq d(x_1, o_0)$$

Using Definition 4, we write

$$d(x_0, o_1) - 2\rho > d(x_0, o_0)$$

From the previous two equations, we obtain

$$d(x_0, o_1) - 2\rho + d(x_0, x_1) > d(x_1, o_0) \tag{5}$$

Definition 4 also implies

$$d(x_1, o_0) > 2\rho + d(x_1, o_1) \tag{6}$$

Substituting 5 and 6, we can write

$$d(x_0, o_1) - 2\rho + d(x_0, x_1) > 2\rho + d(x_1, o_1) \tag{7}$$

Using the second triangle inequality, we have

$$d(x_1, o_1) + d(x_0, x_1) \geq d(x_0, o_1) \tag{8}$$

Summing 7 and 8, we get

$$d(x_0, o_1) + d(x_1, o_1) - 2\rho + 2d(x_0, x_1) > 2\rho + d(x_0, o_1) + d(x_1, o_1)$$

That gives

$$2d(x_0, x_1) > 4\rho$$
$$d(x_0, x_1) > 2\rho$$

□

The correct functionality of the hyperbolic ρ-split assumes $d(o_0, o_1) > 2\rho$. Otherwise, the split function allocates all points from the data file, except the routing points, in the exclusion set X_2, which is certainly not desirable.

Balancing: Equal sizes of subsets X_0 and X_1 can be achieved only by an appropriate choice of routing points, i.e. this ρ-split function has no tuning parameter that would affect balancing. Therefore, we define a modified hyperbolic split function, possessing such property in the following.

3.3 Modified hyperbolic ρ-split

The modified hyperbolic ρ-split function $mhs^\rho(x, o_0, o_1)$ uses two routing objects o_0 and o_1 and a distance δ for partitioning the data file into three subsets X_0, X_1 and X_2.

$$mhs^\rho(x, o_0, o_1) = \begin{cases} 0 \text{ if } d(x, o_1) - d(x, o_0) > 2\rho - 2\delta \\ 1 \text{ if } d(x, o_0) - d(x, o_1) > 2\rho + 2\delta \\ 2 \text{ otherwise} \end{cases} \tag{9}$$

For convenience, the situation is illustrated in Figure 4 from the data space (a) and the distance space (b) points of view, respectively.

The following proof shows that the mhs^ρ-split function is a correct ρ-split function, i.e. the distance between arbitrarily picked couple of objects x_0 and x_1 from subsets X_0 and X_1 is greater than 2ρ.

Proof. We want to show, that $\forall x_0 \in X_0, \forall x_1 \in X_1 : d(x_0, x_1) > 2\rho$.
Firstly, we use the triangle inequality

$$d(x_0, o_0) + d(x_0, x_1) \geq d(x_1, o_0)$$

and Definition 9 to get

$$d(x_0, o_1) - 2\rho + 2\delta > d(x_0, o_0)$$

Summing the previous equations, we obtain

$$d(x_0, o_1) - 2\rho + 2\delta + d(x_0, x_1) > d(x_1, o_0) \tag{10}$$

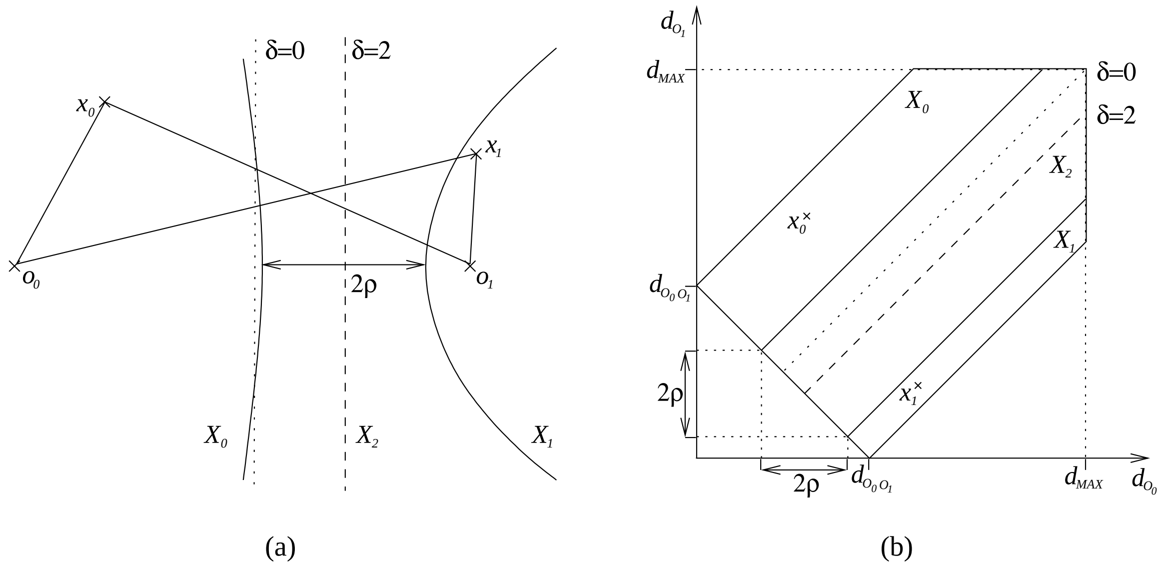

Fig. 4. The modified hyperbolic ρ-split function in the data space point of view (a) and in the distance point of view (b).

Using Definition 9 again, we also have

$$d(x_1, o_0) > 2\rho + 2\delta + d(x_1, o_1) \tag{11}$$

Putting together Equations 10 and 11, we can write

$$d(x_0, o_1) - 2\rho + 2\delta + d(x_0, x_1) > 2\rho + 2\delta + d(x_1, o_1)$$

$$d(x_0, o_1) - 2\rho + d(x_0, x_1) > 2\rho + d(x_1, o_1) \tag{12}$$

Using the second triangle inequality

$$d(x_1, o_1) + d(x_0, x_1) \geq d(x_0, o_1) \tag{13}$$

Summing 12 and 13, we get

$$d(x_0, o_1) + d(x_1, o_1) - 2\rho + 2d(x_0, x_1) > 2\rho + d(x_0, o_1) + d(x_1, o_1)$$

This results in

$$2d(x_0, x_1) > 4\rho$$
$$d(x_0, x_1) > 2\rho$$

□

The constraints for the parameter ρ are the same as for the hyperbolic ρ-split function, that is $2\rho < d(o_0, o_1)$. In addition, the δ parameter must satisfy the following condition.

$$\delta \in \left] \frac{-d(o_0, o_1)}{2} - \rho, \frac{d(o_0, o_1)}{2} - \rho \right[$$

Balancing: Contrary to the hyperbolic split, this method has a special parameter to balance separable sets. The advantage is that the ratio between cardinalities of subsets X_0, X_1 is independent of a choice of the routing points and can always be adjusted. The parameter δ can be decided by using an interval splitting method.

3.4 Comparison

Experiments were conducted on all three data files described in Section 2. In all experiments, we picked 100 reference objects (or pairs of objects) to define ρ-split functions. Then, we considered three different values of ρ, specifically, $\rho_1 = 0.1d_{MAX}$, $\rho_2 = 0.01d_{MAX}$, and $\rho_3 = 0.001d_{MAX}$.

In Table 1, results of our experiments are presented. All numbers are average values from one hundred split function computations, splitting always the entire file. Sizes of sets are in percents with respect to the size of whole data file to avoid problems with different sizes of individual files. In the table, we use $|X|$ to designate the number of elements in X, i.e. the cardinality of the set.

		ρ_1			ρ_2			ρ_3		
		$\lvert X_0\rvert$	$\lvert X_1\rvert$	$\lvert X_2\rvert$	$\lvert X_0\rvert$	$\lvert X_1\rvert$	$\lvert X_2\rvert$	$\lvert X_0\rvert$	$\lvert X_1\rvert$	$\lvert X_2\rvert$
	VPS	32.75%	32.70%	34.55%	48.21%	48.23%	3.56%	49.75%	49.89%	0.36%
UD	HPS	27.00%	25.69%	47.31%	44.42%	50.57%	5.01%	49.88%	49.61%	0.51%
	MHS	24.24%	24.15%	51.61%	47.20%	47.20%	5.60%	49.68%	49.76%	0.56%
	VPS	21.71%	21.65%	56.64%	46.61%	46.58%	6.81%	49.61%	49.68%	0.71%
HV2	HPS	16.79%	15.42%	67.79%	48.94%	42.95%	8.11%	46.93%	52.11%	0.96%
	MHS	5.91%	5.96%	88.13%	39.83%	39.82%	20.35%	48.81%	48.82%	2.37%
	VPS	33.75%	33.77%	32.48%	48.42%	48.37%	3.21%	49.77%	49.89%	0.34%
HV1	HPS	24.32%	28.34%	47.34%	50.64%	44.64%	4.72%	47.44%	52.00%	0.56%
	MHS	20.61%	20.72%	58.67%	46.15%	46.10%	7.75%	49.59%	49.59%	0.82%

Table 1. Relative sizes of individual sets of the vantage point (VPS), the hyperbolic (HPS) and the modified hyperbolic (MHS) ρ-split functions

Vantage point and modified hyperbolic split functions always produce (practically) balanced subsets X_0 and X_1, but the hyperbolic function does not. In order to make this fact even more obvious, Table 2 shows averages of differences between cardinalities of sets X_0 and X_1. Once again, Table 2 demonstrates that the hyperbolic ρ-split function produces very unbalanced sets, therefore this function is not recommended for usage.

		ρ_1	ρ_2	ρ_3
	VPS	0.48%	0.45%	0.37%
UD	HPS	22.52%	28.59%	27.17%
	MHS	0.45%	0.42%	0.42%
	VPS	0.43%	0.45%	0.50%
HV2	HPS	25.78%	64.66%	57.75%
	MHS	0.40%	0.42%	0.41%
	VPS	0.44%	0.43%	0.40%
HV1	HPS	29.30%	35.58%	39.16%
	MHS	0.43%	0.47%	0.44%

Table 2. Differences between sizes of sets S_0 and S_1, in percents

Another interesting observation is that the hyperbolic function produces a greater exclusion set compared to the vantage point approach. This is mainly due to the shape of the hyperbolic curves that divide a metric space into three sets. As Figure 3(a) demonstrates, the curves are departing with growing distance from the routing point, so the amount of data in the exclusion set is essentially larger than necessary. The size of exclusion set of the modified hyperbolic ρ-split function is even greater than size of exclusion set of the hyperbolic ρ-split because of a greater distance between the curves. In order to understand, consider Figure 4(a) again and see that in case the parameter δ is different from zero, the distance between the curves grows. Notice that a distance between curves (metric space balls) defined by the vantage point function is constantly equal to 2ρ. That's the reason for the smallest size of the exclusion set of the vantage point ρ-split function. We are aware of the fact that the distribution of objects in partitions depends on the distance distribution, which can be different for different data files. However, such influence has not been very significant in our experiments.

We have also experimentally studied the problem of choosing reference objects in relation to our performance objectives, that is the balancing and minimization. Our observations can be summarized as follows:

- *VPS* (the vantage point function), all experiments we have made show that the size of the exclusion set is lower if a routing object is chosen outside the center of given space, and the larger the distance from the center the better.
- *HPS* (the hyperbolic ρ-split), a bad choice of routing objects implies unbalanced sets.
- *HPS* and *MHS* (the modified hyperbolic split), generally, we observe that an exclusion set is smaller if routing objects are more distant. In theory, if the distance between routing points is infinite then the distance between curves is constantly equal to 2ρ.

These observations are demonstrated in Figure 5. All values in this figure are from fifty computations of ρ-split functions on UD data file where routing objects were chosen from a diagonal of given space. Routing objects were picked uniformly starting from the center of diagonal to the ends i.e. to the corners of given space. So the distance between routing objects is growing from the center to the corners, from zero to d_{MAX}. The zero value on the x axis represents the center of space and d_{MAX} the corners of space. Because the vantage point ρ-split function uses only one routing object, all computations of *VPS* were made using the first routing object of a picked pair.

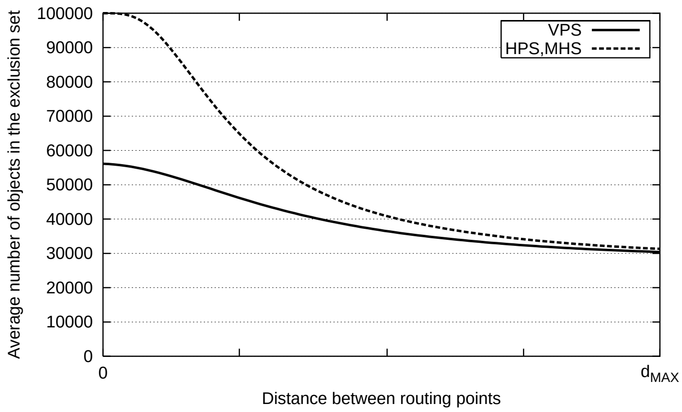

Fig. 5. Decreasing size of the excluded partition with growing distance between routing points

Since the hyperbolic ρ-split functions suffer from producing relatively large excluded sets, we have proposed additional two functions, which are described in the following section.

4 New ρ-split Partitioning Strategies

In this section we define two new ρ-split functions and investigate their properties in sense of the minimization and balancing. The first function is called the elliptic ρ-split function, which is a geometric alternative of the hyperbolic split function. The second, called the pseudo-elliptic ρ-split function, is a combination of the vantage point and the elliptic ρ-split functions.

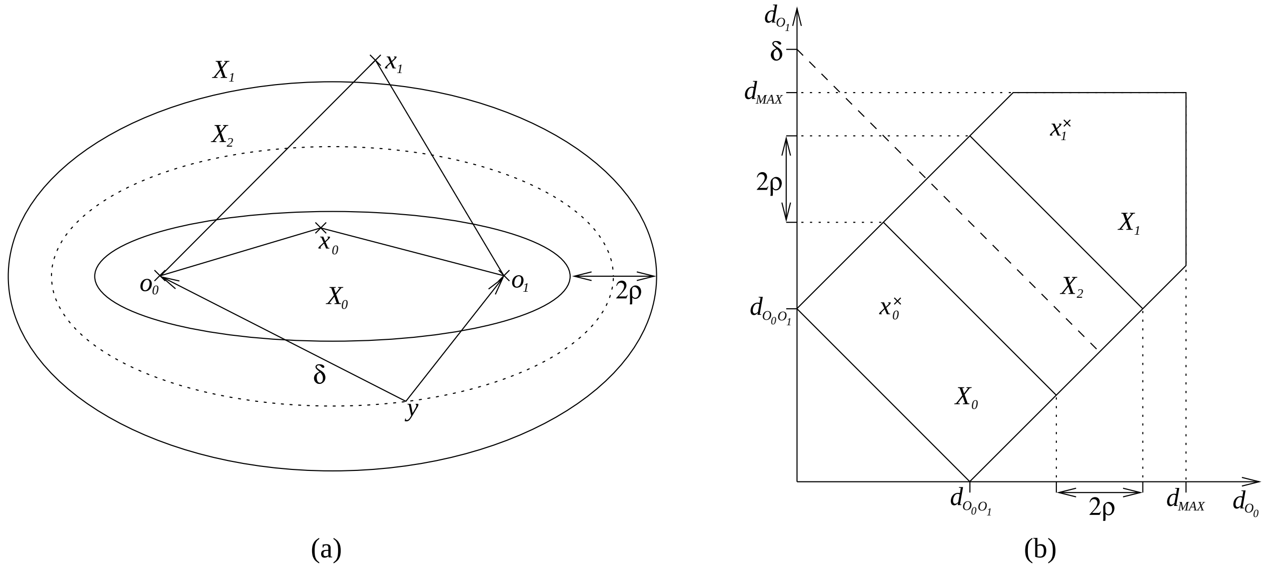

Fig. 6. The elliptic ρ-split function in the data space (a) and the distance (b) points of view.

4.1 Elliptic ρ-split

Elliptic ρ-split function $els^{\rho}(x, o_0, o_1)$ uses two reference objects o_0 and o_1 and a distance $\delta = d(o_0, y) + d(y, o_1)$, see Figure 6(a), for partitioning the data file into three subsets X_0, X_1 and X_2.

$$els^{\rho}(x, o_0, o_1) = \begin{cases} 0 \text{ if } d(x, o_0) + d(x, o_1) < \delta - 2\rho \\ 1 \text{ if } d(x, o_0) + d(x, o_1) > \delta + 2\rho \\ 2 \text{ otherwise} \end{cases} \tag{14}$$

For convenience, the distance space of elliptic split is illustrated in Figure 6(b).

The following proof shows that the els^{ρ}-split function satisfies the necessary properties of ρ-split functions, i.e. the distance between arbitrarily picked objects x_0 and x_1 from subsets X_0 and X_1, respectively, is greater than 2ρ.

Proof. To be proved: $\forall x_0 \in X_0, \forall x_1 \in X_1 : d(x_0, x_1) > 2\rho$.
From the triangle inequalities, we have

$$d(x_0, o_0) + d(x_0, x_1) \geq d(x_1, o_0)$$
$$d(x_0, o_1) + d(x_0, x_1) \geq d(x_1, o_1)$$

By summing the previous inequalities, we have

$$d(x_0, o_0) + d(x_0, o_1) + 2d(x_0, x_1) \geq d(x_1, o_0) + d(x_1, o_1)$$

If we use the expression 14 twice, step by step, we get

$$2d(x_0, x_1) + \delta - 2\rho > d(x_1, o_0) + d(x_1, o_1)$$
$$2d(x_0, x_1) + \delta - 2\rho > \delta + 2\rho$$

That is

$$d(x_0, x_1) > 2\rho$$

□

As for all the other methods, the parameter ρ must not be negative. The parameter δ must be greater than 2ρ, otherwise the ρ-split function never assigns an object in the set X_0.

Balancing: The balancing principle is the same as for the modified hyperbolic split, and the balance property is independent of the choice of routing points. Once again, the parameter δ is computed using an interval splitting method. However, if the distance between routing points is approximately the same as the maximum distance d_{MAX} of a space, the elliptic split function cannot produce balanced subsets because all points come into a set X_0, this fact is also demonstrated in Figure 7 where values close to d_{MAX} are missing.

Comparison: Elliptic ρ-split function produces balanced subsets X_0 and X_1 in a similar way as the modified hyperbolic function (MHS) does. But, as Table 3 demonstrates, the elliptic function has smaller excluded partition (set X_2) than *MHS*. That is because the excluded partition is bounded by curves defined by the elliptic ρ-split function. However, the size of this exclusion set is still greater than the excluded partition's size of the vantage point ρ-split function. The reason is that the distance between curves in the elliptic split is not constant and can become greater than 2ρ, which can easily be verified in Figure 6(a).

		ρ_1			ρ_2			ρ_3		
		$\|X_0\|$	$\|X_1\|$	$\|X_2\|$	$\|X_0\|$	$\|X_1\|$	$\|X_2\|$	$\|X_0\|$	$\|X_1\|$	$\|X_2\|$
UD	VPS	32.75%	32.70%	34.55%	48.21%	48.23%	3.56%	49.75%	49.89%	0.36%
UD	MHS	24.24%	24.15%	51.61%	47.20%	47.20%	5.60%	49.68%	49.76%	0.56%
UD	ELS	28.04%	28.10%	43.86%	47.67%	47.71%	4.62%	49.81%	49.73%	0.46%
HV2	VPS	21.71%	21.65%	56.64%	46.61%	46.58%	6.81%	49.61%	49.68%	0.71%
HV2	MHS	5.91%	5.96%	88.13%	39.83%	39.82%	20.35%	48.81%	48.82%	2.37%
HV2	ELS	19.57%	19.53%	60.90%	46.39%	46.42%	7.19%	49.66%	49.61%	0.73%
HV1	VPS	33.75%	33.77%	32.48%	48.42%	48.37%	3.21%	49.77%	49.89%	0.34%
HV1	MHS	20.61%	20.72%	58.67%	46.15%	46.10%	7.75%	49.59%	49.59%	0.82%
HV1	ELS	28.99%	29.05%	41.96%	47.73%	47.84%	4.43%	49.82%	49.74%	0.44%

Table 3. Percentage sizes of sets of the vantage point (VPS), the modified hyperbolic (MHS) and the elliptic (ELS) ρ-split functions

A recommendation for a suitable choice of routing objects is just the opposite of the advice we have made for the hyperbolic and the modified hyperbolic ρ-split functions. Since the size of the exclusion set is increasing with a distance between routing points, the better choice is to use points that are close to each other. The fact is illustrated in Figure 7, where the values were obtained through elliptic split functions with generated pair of routing objects in fifty iterations and using UD data file. Pairs of objects were chosen uniformly from a diagonal from the center to the corners of the UD metric space.

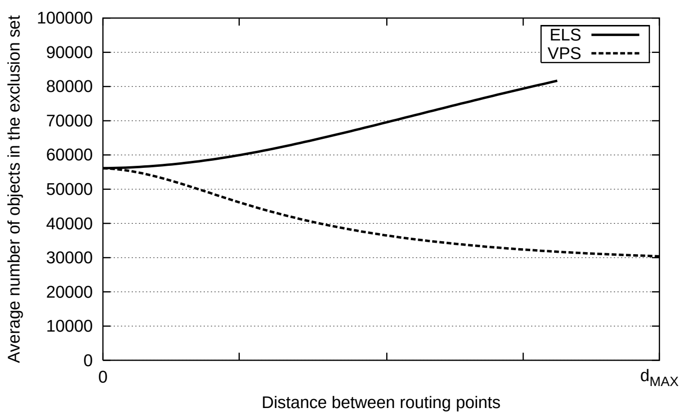

Fig. 7. Increasing size of the excluded partition with growing distance between routing points

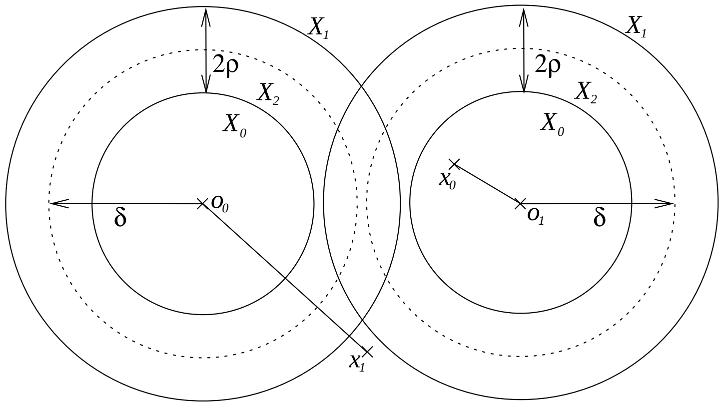

Fig. 8. The pseudo-elliptic ρ-split function

In summary, we can say that the results of the vantage point and the elliptic ρ-split functions are similar, so up to now, both of them can be considered as suitable candidates for application.

4.2 Pseudo-elliptic ρ-split

The second new function, the pseudo-elliptic ρ-split function $pes^{\rho}(x, o_0, o_1)$, uses two objects o_0 and o_1 for partitioning a data file into three subsets X_0, X_1 and X_2.

$$pes^{\rho}(x, o_0, o_1) = \begin{cases} 0 \text{ if } d(x, o_0) < \delta - \rho \;\text{ or }\; d(x, o_1) < \delta - \rho \\ 1 \text{ if } d(x, o_0) > \delta + \rho \text{ and } d(x, o_1) > \delta + \rho \\ 2 \text{ otherwise} \end{cases} \tag{15}$$

The situation is illustrated in Figure 8. The following proof shows that the pes^{ρ}-split function behaves in the desired way, i.e. the distance between arbitrarily picked objects x_0 and x_1 from subsets X_0 and X_1 is greater than 2ρ.

Proof. We want to show that: $\forall x_0 \in X_0, \forall x_1 \in X_1 : d(x_0, x_1) > 2\rho$.
Obviously, the following triangle inequality holds

$$d(x_0, o_0) + d(x_0, x_1) \geq d(x_1, o_0)$$

From the previous equation and Equation 15, we have

$$d(x_0, x_1) + d(x_0, o_0) > \delta + \rho \tag{16}$$

By analogy

$$d(x_0, x_1) + d(x_0, o_1) > \delta + \rho \tag{17}$$

According to 15, one of the two following assertions is true

$$\delta - \rho > d(x_0, o_0) \tag{18}$$
$$\delta - \rho > d(x_0, o_1) \tag{19}$$

That implies that also one of the following two expressions is true

$$d(x_0, x_1) + \delta - \rho > \delta + \rho \qquad (\textit{combining } 16 \textit{ and } 18)$$
$$d(x_0, x_1) + \delta - \rho > \delta + \rho \qquad (\textit{combining } 17 \textit{ and } 19)$$

So we get

$$d(x_0, x_1) > 2\rho$$

□

Balancing: Obviously, the assertion $\delta > \rho$ must be satisfied. An appropriate choice of δ results in sets X_0 and X_1 of equal sizes. The important thing is that this possibility is independent of a specific choice of routing points, so balancing is always possible.

Comparison: The size of the excluded partition of this ρ-split function is upper-bounded by curves which have constant distance 2ρ. Therefore the size of the exclusion set is smaller than the exclusion set of the elliptic ρ-split function, that can easily be verified in Table 4. The results showed in this table for the pseudo-elliptic method are similar to the vantage point split function's results. That implies this two methods are interchangeable.

		ρ_1			ρ_2			ρ_3		
		$\lvert X_0\rvert$	$\lvert X_1\rvert$	$\lvert X_2\rvert$	$\lvert X_0\rvert$	$\lvert X_1\rvert$	$\lvert X_2\rvert$	$\lvert X_0\rvert$	$\lvert X_1\rvert$	$\lvert X_2\rvert$
	VPS	32.75%	32.70%	34.55%	48.21%	48.23%	3.56%	49.75%	49.89%	0.36%
UD	ELS	28.04%	28.10%	43.86%	47.67%	47.71%	4.62%	49.81%	49.73%	0.46%
	PES	30.58%	30.69%	38.73%	47.97%	47.99%	4.04%	49.81%	49.79%	0.40%
	VPS	21.71%	21.65%	56.64%	46.61%	46.58%	6.81%	49.61%	49.68%	0.71%
HV2	ELS	19.57%	19.53%	60.90%	46.39%	46.42%	7.19%	49.66%	49.61%	0.73%
	PES	22.43%	22.54%	55.03%	46.87%	46.90%	6.23%	49.69%	49.68%	0.63%
	VPS	33.75%	33.77%	32.48%	48.42%	48.37%	3.21%	49.77%	49.89%	0.34%
HV1	ELS	28.99%	29.05%	41.96%	47.73%	47.84%	4.43%	49.82%	49.74%	0.44%
	PES	32.44%	32.46%	35.10%	48.21%	48.22%	3.57%	49.79%	49.86%	0.35%

Table 4. Percentage sizes of sets in *VPS*, *ELS* and *PES*

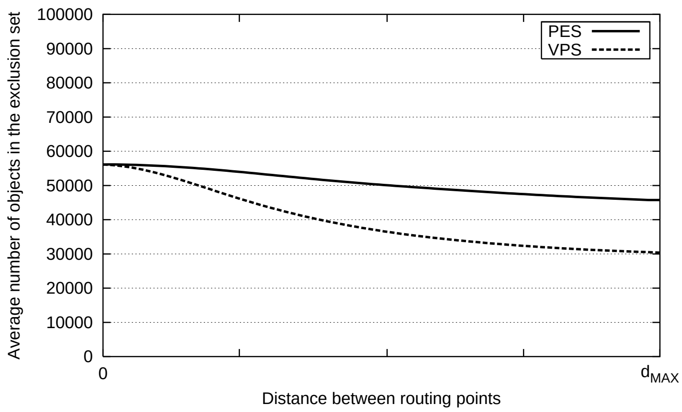

Fig. 9. Decreasing size of the exclusion partition with growing distance between routing points

Observations on the suitability of selectivity the routing points are summarized in Figure 9 – all values in this figure were computed in the same way as in previous Figures 5 and 7. As this graph suggests, the size of the exclusion set is falling slowly with growing distance, but not significantly. The exclusion set is the smallest for reference points chosen from the corners of given space.

5 Conclusion

In this paper, we have proposed, analyzed, and compared several implementation approaches that are able to achieve separable partitioning of metric data sets. These principles are arising from the ball and the generalized hyperplane decomposition of metric spaces, introduced in [Uh91]. However, they are modified to the case of

ρ-split functions, that divide (with possible exclusion) a metric space into two sets, which are separable for similarity range queries with distance radii up to ρ. Besides, we have also proposed two completely new partitioning principles. Properties of the proposed techniques were verified on one synthetic and two real-life data files.

The first considered technique was the vantage point ρ-split function (VPS) and the second one was the hyperbolic ρ-split function (HPS). We have compared them in sense of size *balancing* of separable sets and *minimization* of the exclusion set size. Experimentally, we have found out that the hyperbolic function produces unbalanced separable sets. Therefore, we have proposed the modified hyperbolic ρ-split function (MHS), which can be tuned to achieve balancing. In terms of the minimization of exclusion set, VPS has smaller exclusion set than HPS and MHS. Unfortunately, MHS generates larger exclusion set than HPS.

For this reasons, we have proposed new partitioning principles, that do not have any analogy in previous technical literature. Specifically, the elliptic ρ-split function (ELS) and the pseudo-elliptic ρ-split (PES) function. The elliptic ρ-split function has approximately the same qualities as the vantage point method though the VPS is still a bit better. On the other hand, the results produced by PES are better than those achieved by ELS, but comparable to VPS. So the PES method is interchangeable with VPS.

We have also addressed the problem of choosing appropriate reference objects. It seems that the routing objects for VPS are better to be chosen from corners, rather than from an inner area of given space. Such findings are quite consistent with similar investigations, e.g. in [BO99]. For HPS and MHS functions, it is more desirable to choose pairs of routing objects with the largest possible distance. This is in contrast to the ELS function where close reference objects are preferable. Finally, the recommended distance between pairs of reference objects for PES function is d_{MAX} – the reference objects are corners of given space.

Currently, we are investigating possibilities to develop other ρ-split functions, that would perform even better. We are also considering the case of separable partitioning of higher order than binary. All these functions are tested in the similarity hashing structure [Ze00] and we will report on the results in another paper.

References

[Bay99] Bay, S. D. The UCI KDD Archive [http://kdd.ics.uci.edu]. Irvine, CA: University of California, Department of Information and Computer Science.

[BO99] T. Bozkaya and Ozsoyoglu. Indexing Large Metric Spaces for Similarity Search Queries. *ACM TODS*, 24(3):361-404, 1999.

[Br95] S. Brin. Near Neighbor Search in Large Metric Spaces. *Proceedings of the 21st VLDB Conference*, pp. 574-584, 1995.

[Ch94] T. Chiueh. Content-Based Image Indexing. *Proceedings of the 20th VLDB Conference*, pp. 582-593, 1994.

[CPZ97] 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*, pp. 426-435, 1997.

[Ke55] J.L. Kelly. *General Topology*, D. Van Nostrand, New York, 1955.

[TTS+00] C. Traina Jr, A. Traina, B. Seeger, and C. Faloutsos. Slim-trees: High Performance Metric Trees Minimizing Overlap Between Nodes. In *Proceedings of the 7th EDBT International Conference*, Konstanz, Germany, March 2000, pp. 51-56.

[Uh91] J.K. Uhlmann. Satisfying general proximity/similarity queries with metric trees. *Information Processing Letters*, 40(4):175–179, November 1991.

[Yi99] P.N. Yianilos. Excluded Middle Vantage Point Forests for Nearest Neighbor Search. Tech. rep., NEC Research Institute, 1999, Presented at Sixth DIMACS Implementation Challenge: Nearest Neighbor Searches workshop, January 15, 1999.

[Ze00] C. Gennaro, P. Savino, and P. Zezula. A Hashed Schema for Similarity Search in Metric Spaces (invited talk). *Proceedings of the First DELOS Network of Excellence Workshop on "Information Seeking, Searching and Querying in Digital Libraries"*, Zurich, Switzerland, 2000, pp. 83-88.