Multiple circle detection based on center-based clustering

Rudolf Scitovski[a], Tomislav Marošević[a,*]

[a]*Department of Mathematics, University of Osijek, Trg Lj. Gaja 6, HR – 31 000 Osijek, Croatia*

Abstract

The multiple circle detection problem has been considered in the paper on the basis of given data point set $\mathcal{A} \subset \mathbb{R}^2$. It is supposed that all data points from the set $\mathcal{A}$ come from k circles that should be reconstructed or detected. The problem has been solved by the application of center-based clustering of the set $\mathcal{A}$, i.e. an optimal k-partition is searched for, whose clusters are determined by corresponding circle-centers. Thereby, the algebraic distance from a point to the circle is used. First, an adaptation of the well-known k-means algorithm is given in the paper. Also, the incremental algorithm for searching for an approximate globally optimal k-partition is proposed. The algorithm locates either a globally optimal k-partition or a locally optimal k-partition close to the global one. Since optimal partitions with $2, 3, \ldots$ clusters are determined successively in the algorithm, several well-known indexes for determining an appropriate number of clusters in a partition are adopted for this case. Thereby, the Hausdorff distance between two circles is used and adopted. The proposed method and algorithm are illustrated and tested on several numerical examples.

Keywords: Multiple circle detection, Center-based clustering, Globally optimal partition, Approximate optimization, DIRECT

1. Introduction

Clustering or grouping a data set into conceptually meaningful clusters is a well-studied problem in recent literature, and it has practical importance in a wide variety of applications such as medicine, biology, pattern recognition, facility location problem, text classification, information retrieval, earthquake investigation, understanding the Earth's climate, psychology, ranking of municipalities for financial support, business, etc. (Liao et al., 2012; Morales-Esteban et al., 2010; Pintér, 1996; Sabo et al., 2011, 2013; Tan et al., 2006).

A multiple circle detection problem is considered in the paper based on given data set $\mathcal{A} \subset \mathbb{R}^2$. Let us assume that all data from the set $\mathcal{A} = \{a_i = (x_i, y_i) \in \mathbb{R}^2 \colon i = 1, \ldots, m\} \subset \mathbb{R}^2$ come from k circles that should be reconstructed or detected. There are many different approaches for solving this problem in literature, which are most often based on Hough transformation (Ballard, 1981), different heuristic methods (Chung et al., 2012; Cuevas et al., 2012; Kim and Kim, 2001; Qiao and Ong, 2004), `RANSAC` (Fischler and Bolles, 1981) or fuzzy clustering techniques, that search for the so-called soft or fuzzy partitions (Bezdek et al., 2005; Song et al., 2010; Theodoridis and Koutroumbas, 2009).

A center-based clustering method is applied to solving this problem (Kogan, 2007; Sabo et al., 2013; Teboulle, 2007). The set $\mathcal{A}$ will be grouped into k nonempty disjoint subsets $\pi_1, \ldots, \pi_k$, $1 \leq k \leq m$, such that

$$\bigcup_{i=1}^{k} \pi_i = \mathcal{A}, \quad \pi_r \cap \pi_s = \emptyset, \ r \neq s, \quad |\pi_j| \geq 1, \ j = 1, \ldots, k. \tag{1}$$

This partition will be denoted by $\Pi(\mathcal{A}) = \{\pi_1, \ldots, \pi_k\}$, and the elements $\pi_1, \ldots, \pi_k$ of such a partition are called *clusters* in $\mathbb{R}^2$.

To each cluster $\pi_j \in \Pi$ a corresponding circle-center $C_j^\star(S_j^\star, r_j^\star)$ with center $S_j^\star = (p_j^\star, q_j^\star)$ and radius $r_j^\star$ is associated by solving the following global optimization problem (GOP)

$$(p_j^\star, q_j^\star, r_j^\star) = \underset{p,q,r \in \mathbb{R}}{\operatorname{argmin}} \sum_{a_i \in \pi_j} D(C(p, q, r), a_i), \tag{2}$$

where $D(C(p, q, r), a_i)$ represents the distance from the point a_i to the circle C (see *Section 2.1*).

If the objective function $\mathcal{F} \colon \mathcal{P}(\mathcal{A}; m, k) \to \mathbb{R}_+$ is defined on the set of all partitions $\mathcal{P}(\mathcal{A}; m, k)$ of the set $\mathcal{A}$ containing k clusters, then the quality of a partition may be defined and one can search for the *globally optimal k-partition* by solving the following GOP

$$\underset{\Pi \in \mathcal{P}(\mathcal{A}; m,k)}{\operatorname{argmin}} \mathcal{F}(\Pi), \quad \mathcal{F}(\Pi) = \sum_{j=1}^{k} \sum_{a_i \in \pi_j} D(C_j(p_j, q_j, r_j), a_i). \tag{3}$$

Conversely, for a given set of different circles $C_1, \ldots, C_k \subset \mathbb{R}^2$, applying the minimal distance principle, the partition $\Pi = \{\pi_1, \ldots, \pi_k\}$ of the set $\mathcal{A}$ can be defined in the following way:

$$\pi_j = \{a \in \mathcal{A} \colon D(C_j, a) \leq D(C_s, a), \ \forall s = 1, \ldots, k, \ s \neq j\}, \quad j = 1, \ldots, k, \tag{4}$$

*Corresponding author: Tel.: ++385-31-224-800; fax: ++385-31-224-801;
Email address: `tmarosev@mathos.hr` (Tomislav Marošević)

where one has to pay attention to the fact that every element of the set $\mathcal{A}$ occurs in one and only one cluster. Therefore, the problem of finding an optimal partition of the set $\mathcal{A}$ can be reduced to the following optimization problem

$$\operatorname*{argmin}_{C_1,\dots,C_k\subset\mathbb{R}^2} F(C_1,\dots,C_k), \tag{5}$$

$$F(C_1,\dots,C_k) = \sum_{i=1}^{m} \min_{j=1,\dots,k} D(C_j,a_i).$$

The solution of (3) and (5) coincides (Späth, 1983). In general, the function F can have a great number of independent variables, it does not have to be either convex or differentiable and usually it has several local and global minima. Hence, this becomes a complex GOP.

The adaptation of the well-known k-means algorithm for searching for a locally optimal partition is given in *Section 2*, where clusters are determined by corresponding circle-centers. In *Section 3*, a new algorithm for searching for an optimal partition is proposed and an illustrative example is shown. In *Section 4*, the problem of determining an appropriate number of clusters in a partition is considered. The well-known Davies-Bouldin index, Calinski-Harabasz index, and Simplified Silhouette Width criterion are adopted and used for the case observed in the paper. Finally, in *Section 5*, several numerical examples are shown.

2. Adaptation of the *k*-means algorithm

The well-known k-means algorithm (Kogan, 2007; Leisch, 2006; Teboulle, 2007) will be adapted for searching for a locally optimal partition with circles as clusters-centers. The algorithm can be described in two steps which are repeated iteratively:

Algorithm 1. *(k closest circles algorithm (KCC))*

Step A: For each set of mutually different circles $C_1,\dots,C_k$ the set $\mathcal{A}$ should be divided into k disjoint unempty clusters $\pi_1,\dots,\pi_k$ by using the minimal distance principle (4);

Step B: Given a partition $\Pi = \{\pi_1,\dots,\pi_k\}$ of the set $\mathcal{A}$, one can define the corresponding circle-centers $C_1^\star(p_1^\star,q_1^\star,r_1^\star),\dots,C_k^\star(p_k^\star,q_k^\star,r_k^\star)$ by solving the following GOPs $(j=1,\dots,k)$

$$(p_j^\star,q_j^\star,r_j^\star) = \operatorname*{argmin}_{p,q,r\in\mathbb{R}} \sum_{a_i\in\pi_j} D(C(p,q,r),a_i). \tag{6}$$

Knowing a good initial approximation, this algorithm can provide an acceptable solution, but in case we do not have a good initial approximation, the KCC-algorithm can be restarted several times with various random initializations (Leisch, 2006).

2.1. The distance from the point to the circle

In order to solve the multiple circle detection problem, it is very important for the distance between a point and a circle to be well defined. This distance will be used by applying the minimal distance principle (4), by determining circle-centers of clusters (6), and by defining the most appropriate number of clusters in a partition in *Section 4*. Several approaches to determining the distance from the point $a_i = (x_i,y_i) \in \mathbb{R}^2$ to the circle $C(S,r)$ with center $S = (p,q)$ and radius r have been proposed in literature (Ahn et al., 2001; Chernov, 2010; Drezner et al., 2002; Nievergelt, 2002; Song et al., 2010):

$$D_1(C(S,r),a_i) = |\|S-a_i\| - r|, \tag{7}$$

$$D_2(C(S,r),a_i) = (\|S-a_i\| - r)^2, \tag{8}$$

$$D(C(S,r),a_i) = (\|S-a_i\|^2 - r^2)^2. \tag{9}$$

The last possibility (9) is called the *algebraic distance*. It is very often used in practical applications (see e.g. Nievergelt (2002); Song et al. (2010)) and for that reason that possibility is also used in our paper.

2.2. Searching for the circle-center of a cluster

The GOPs (6) can have several local and global minima and the corresponding minimizing function is continuous. Note that, if (7) or (8) were used as the distance from the point a_i to the circle C, then the corresponding minimizing function would be a Lipschitz continuous function (see e.g. Sergeyev and Kvasov (2011)). Therefore, for solving GOPs (6) in these cases, some global optimization methods (see e.g. (Jones et al., 1993; Neumaier, 2004; Pintér, 1996; Sergeyev et al., 2001) and corresponding software `http://www.pinterconsulting.com`) can be used. One of the most popular algorithms for solving a GOP for the Lipschitz continuous function is the `DIRECT` (DIvidingRECTangles) algorithm (Finkel, 2003; Jones et al., 1993; Grbić et al., 2013).

In the case of using the algebraic distance (9), exact solutions of GOPs (6) can be obtained (Chernov, 2010). In the case of using the distance (7) or (8), special methods can also be applied for solving GOPs (6) (Nievergelt, 2002). The random circle consensus technique - RANSAC (Fischler and Bolles, 1981) can also be used for circle fitting.

3. A method for searching for a globally optimal partition

Let $\mathcal{A} \subset [a,b]\times[c,d] \subset \mathbb{R}^2$ be a data points set. A globally optimal k-partition $\Pi^\star = \{\pi_1^\star,\dots,\pi_k^\star\}$ with circle-centers $C_1^\star,\dots,C_k^\star$ should be determined as a solution of GOP (3), or equivalently (5).

Since objective function (5) is a Lipschitz continuous function, the aforementioned `DIRECT` algorithm for global optimization can be used. Because of the symmetry property of the function F defined by (5), there are at least $k!$ solutions of this problem. A very efficient special version of the DIRECT algorithm for symmetric functions is developed in (Grbić et al., 2013). However, complexity of this problem is particularly emphasized if the number of clusters or data points m is large. Instead of solving the GOP (5), motivated by (Bagirov and Ugon, 2005; Bagirov et al., 2011; Scitovski and Scitovski, 2013), an efficient method for searching for a very acceptable approximate of an optimal k-partition is proposed.

First, a sequence of objective functions $F_k\colon \underbrace{\mathbb{R}^3 \times \cdots \times \mathbb{R}^3}_{k} \to \mathbb{R}_+$,

$$F_k(C_1,\ldots,C_k) = \sum_{i=1}^{m} \min\{D(C_1,a_i),\ldots,D(C_k,a_i)\}, \tag{10}$$

is defined, where $C_j := (S_j, r_j)$, $S_j = (p_j, q_j)$.

For $k = 1$, the problem is reduced to a simple problem of locating a circle on the basis of a given set of points in the plane. The corresponding function $F_1\colon \mathbb{R}^3 \to \mathbb{R}_+$ is of the form

$$\begin{aligned} F_1(p_1,q_1,r_1) &= \sum_{i=1}^{m} D(C_1(p_1,q_1,r_1),a_i) \\ &= \sum_{i=1}^{m} (\|S_1 - a_i\|^2 - r_1^2)^2, \end{aligned} \tag{11}$$

where $S_1 = (p_1, q_1)$. There exists a number of algorithms and methods for solving this problem (see e.g. (Chernov, 2010; Drezner et al., 2002; Nievergelt, 2002)).

For $k > 1$, an optimal k-partition with circle-centers $C_1^{\star(k)},\ldots,C_k^{\star(k)}$ is determined by the following incremental algorithm.

Algorithm 2. (Searching for an optimal k-partition)

Step 1: *Let $\hat{C}_1,\ldots,\hat{C}_{k-1}$ be the solution to the $k-1$-circle detecting problem and let*

$$F_{k-1}(\hat{C}_1,\ldots,\hat{C}_{k-1}) = \sum_{i=1}^{m} \delta_{k-1}^{i},$$
$$\delta_{k-1}^{i} = \min\{D(\hat{C}_1,a_i),\ldots,D(\hat{C}_{k-1},a_i)\}, \tag{12}$$

$$\begin{aligned} \Phi_k(p,q,r) &:= F_k(\hat{C}_1,\ldots,\hat{C}_{k-1},C(p,q,r)) = \\ &= \sum_{i=1}^{m} \min\{\delta_{k-1}^{i}, D(C(p,q,r),a_i)\}. \end{aligned} \tag{13}$$

Step 2: *By using the DIRECT algorithm for global optimization determine*

$$\hat{C}_k \in \operatorname*{argmin}_{\substack{(p,q)\in[a,b]\times[c,d] \\ r\in[0,r_{max}]}} \Phi_k(p,q,r), \tag{14}$$

where r_{max} is the maximum radius of circle C, which can be expected in rectangle $[a,b]\times[c,d]$.

Step 3: *By using the KCC-algorithm with initial approximations $\hat{C}_1,\ldots,\hat{C}_k$ determine new centers $C_1^{\star(k)},\ldots,C_k^{\star(k)}$.*

An important assumption of the algorithm is usage of an efficient algorithm for solving a GOP in Step 2. The DIRECT algorithm for global optimization of a Lipschitz continuous function was proposed by (Jones et al., 1993). The algorithm operates by systematically dividing the box domain of the objective function into hyperrectangles, and evaluating its values in their centers. There are two phases to an iteration of DIRECT; first, hyperrectangles are identified as potentially optimal, i.e., they have potential to contain a global solution. The second phase of an iteration is to divide potentially optimal hyperrectangles into smaller hyperrectangles. The objective function is evaluated in the centers of new hyperrectangles.

By increasing the number of circle-centers (Step 2) and by applying the KCC algorithm (Step 3), the objective function value does not increase. Therefore,

$$\begin{aligned} &F_1(C_1^{\star(1)}) \geq F_2(C_1^{\star(1)},\hat{C}_2) \geq F_2(C_1^{\star(2)},C_2^{\star(2)}) \geq \cdots \\ &\geq F_k(C_1^{\star(k-1)},\ldots,\hat{C}_k) \geq F_k(C_1^{\star(k)},\ldots,C_k^{\star(k)}) = F_k^{\star}, \end{aligned}$$

and, according to (Bagirov and Ugon, 2005), the maximum number of clusters k_{max} that makes sense to be calculated using Algorithm 2 is determined by

$$\frac{F_{k_{max}-1}^{\star} - F_{k_{max}}^{\star}}{F_1^{\star}} < \epsilon, \tag{15}$$

for some small $\epsilon > 0$. Namely, in that case the relative reduction of the objective function value for $k \geq k_{max}$ is less than ϵ.

Remark 1. *Note that the partitions obtained in this way can unfortunately not be said to be globally optimal, but numerous calculations in* Section 5 *show that the partitions obtained in this way are either globally optimal or locally optimal partitions close to globally optimal ones and therefore acceptable in practical applications. In what follows, the partition obtained by Algorithm 2 will simply be called an* optimal partition*.*

Note also that by using Algorithm 2 an optimal partition for each $k \leq k_{max}$ is obtained, which makes it possible to decide on the appropriate number of clusters in a partition by adaptation of various well-known indexes (see Section 4).

In the following example the proposed Algorithm 2 will be illustrated on synthetic data.

Example 1. *The set of 5 circles $C = \{C_i = S_i + r_i(\cos t, \sin t)\colon t \in [0,2\pi],\ i = 1,\ldots,5\}$ is given in the plane. In the neighborhood of the i-th circle $n_i \sim \mathcal{U}(30,50)$ random points are generated by using binormal random additive errors with mean vector $\mathbf{0} \in \mathbb{R}^2$ and the covariance matrix $.05\,\mathbf{I}$, where $\mathbf{I} \in \mathbb{R}^{2\times 2}$ is the identity matrix. In this way, a data point set $\mathcal{A} = \{a_i \in \mathbb{R}^2\colon i = 1,\ldots,m\}$, with $m = 47+34+43+34+43$ random points is obtained.*

The algorithm starts with initial circle $\hat{C}_1 = ((3,3),1)$ (see Fig. 1a left). By using the DIRECT algorithm the circle $\hat{C}_2$ is determined in *Step 2* (see also Fig. 1a left) and

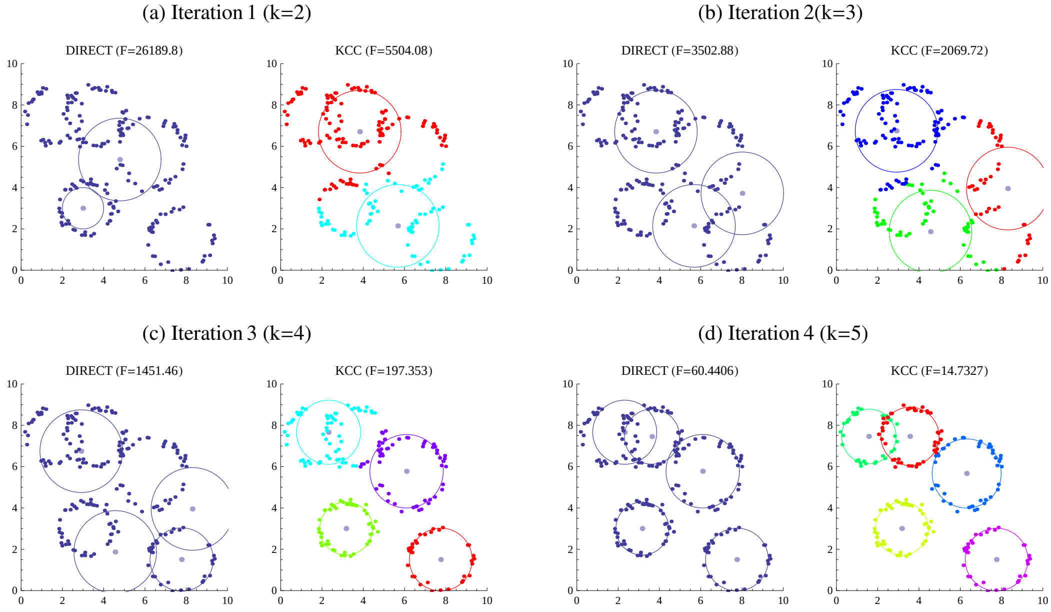

Figure 1: The flow of the iterative process of Algorithm 2

after that, in *Step 3*, circles $C_1^{\star(2)}, C_2^{\star(2)}$ are obtained by using the KCC algorithm (see Fig. 1a right). A further flow of the iterative process provided by Algorithm 2 is illustrated in Fig. 1b-d. In the end of the iterative process the set of recognized circles $C^\star = \{C_i^{\star(5)} : i = 1, \ldots, 5\}$ is obtained. Objective function values during the iterative process of Algorithm 2 are also shown in Fig. 1.

The quality of recognition can be measured by the Hausdorff distance between sets C and $C^\star$

$$d_H(C, C^\star) = \max\{\max_{C_r \in C} \min_{C_s^\star \in C^\star} \delta(C_r, C_s^\star), \max_{C_s^\star \in C^\star} \min_{C_r \in C} \delta(C_r, C_s^\star)\}. \quad (16)$$

In this example, $d_H(C, C^\star) = 0.05465$.

4. The most appropriate number of clusters

Determining the number of clusters automatically has been one of the most difficult problems in data clustering processes (Gan et al., 2007; Kaufman and Rousseeuw, 2005; Kogan, 2007; Vendramin et al., 2009). In simple cases, the number of clusters in a partition is determined by the nature of the problem itself. If the number of clusters in a partition is not given in advance, then it is natural to search for an optimal partition which consists of clusters that are as compact and relatively strongly separated as possible. For determining the most appropriate number of clusters in a partition it is possible to find numerous indexes that point to that number. Some of these indexes are particularly analyzed and adapted in this paper. For this purpose, in addition to the distance measure between a point and a circle, it is important to have a good measure for the distance between two circles.

4.1. The distance between two circles

The distance between two circles $C_1 = (S_1, r_1)$, $C_2 = (S_2, r_2)$ in the plane can be defined as the Hausdorff distance (Chen et al., 2010; Jüttler, 2001)

$$d_H(C_1, C_2) = \max\{\max_{Y \in C_2} \min_{X \in C_1} \|X - Y\|, \max_{X \in C_1} \min_{Y \in C_2} \|X - Y\|\}, \quad (17)$$

where $\|\cdot\|$ denotes the Euclidean distance.

Proposition 1. *For any two circles $C_1 = (S_1, r_1)$, $C_2 = (S_2, r_2)$ in the plane the Hausdorff distance is given by*

$$d_H(C_1, C_2) = \|S_1 - S_2\| + |r_2 - r_1| =: \delta(C_1, C_2). \quad (18)$$

4.2. Adaptation of some of the known indexes

Let $\mathcal{A} \subset \mathbb{R}^2$ be the set and $\Pi^\star = \{\pi_1^\star, \ldots, \pi_k^\star\}$ an optimal k-partition with circle-centers $C_1^\star, \ldots, C_k^\star \in \mathbb{R}^2$. Thereby $C_j^\star = (S_j^\star, r_j^\star)$, where $S_j^\star = (p_j^\star, q_j^\star)$ is the center and $r_j^\star$ the radius of the circle $C_j^\star$.

(a) Davies - Bouldin index in the standard case for the optimal partition $\Pi^\star = \{\pi_1^\star, \ldots, \pi_k^\star\}$ of the set $\mathcal{A}$ with k clusters $\pi_1^\star, \ldots, \pi_k^\star$ and the corresponding centers $c_1^\star, \ldots, c_k^\star \in \mathbb{R}^2$ is defined by (see e.g. (Vendramin et al., 2009))

$$\mathtt{DB}(k) = \frac{1}{k} \sum_{j=1}^{k} \max_{\substack{s=1,\ldots,k \\ s \neq j}} \frac{V(\pi_j^\star) + V(\pi_s^\star)}{\|c_j^\star - c_s^\star\|}, \quad (19)$$

where

$$V(\pi_j^\star) = \frac{1}{|\pi_j^\star|} \sum_{a_s \in \pi_j^\star} \|c_j^\star - a_s\|.$$

That is why the *Davies - Bouldin index for circle-centers* will be defined analogously to (19) by

$$\mathtt{DBC}(k) = \frac{1}{k} \sum_{j=1}^{k} \max_{\substack{s=1,\ldots,k \\ s \neq j}} \frac{\hat{V}(\pi_j^\star) + \hat{V}(\pi_s^\star)}{\delta^2(C_j^\star, C_s^\star)}, \quad (20)$$

where

$$\hat{V}(\pi_j^\star) = \frac{1}{|\pi_j^\star|} \sum_{a_s \in \pi_j^\star} \sqrt{D(C_j^\star, a_s)},$$

and $\delta(C_j^\star, C_s^\star)$ is the distance between circles $C_j^\star, C_s^\star$ defined by (18) in *Proposition 1*. Let us notice that in the definition of the index (20) the property of non-dimensionality has been retained, since nominators and denominators have the same dimension (of squared length).

More compact and better separated clusters in an optimal partition will result in a lower DBC index.

(b) Calinski-Harabasz index in the standard case for the optimal partition $\Pi^\star = \{\pi_1^\star, \dots, \pi_k^\star\}$ of the set A with k clusters $\pi_1^\star, \dots, \pi_k^\star$ and the corresponding centers $c_1^\star, \dots, c_k^\star \in \mathbb{R}^2$ is defined by (see e.g. (Vendramin et al., 2009))

$$\mathtt{CH}(k) = \frac{\mathcal{G}(c_1^\star, \dots, c_k^\star)/(k-1)}{\mathcal{F}(c_1^\star, \dots, c_k^\star)/(m-k)}, \qquad (21)$$

where

$$\mathcal{F}(c_1^\star, \dots, c_k^\star) = \sum_{j=1}^{k} \sum_{a_i \in \pi_j^\star} \|c_j^\star - a_i\|^2,$$

$$\mathcal{G}(c_1^\star, \dots, c_k^\star) = \sum_{j=1}^{k} |\pi_j^\star| \, \|c^\star - c_j^\star\|^2,$$

where $c_j^\star = (1/|\pi_j|) \cdot \sum_{a_i \in \pi_j} a_i$ are centroids of clusters π_j, and $c^\star = (1/|\mathcal{A}|) \cdot \sum_{a_i \in \mathcal{A}} a_i$ is a centroid of the entire set $\mathcal{A}$.

That is why the *Calinski-Harabasz index for circle-centers* will be defined analogously to (21) by

$$\mathtt{CHC}(k) = \frac{\hat{\mathcal{G}}(C_1^\star, \dots, C_k^\star)/(k-1)}{\hat{\mathcal{F}}(C_1^\star, \dots, C_k^\star)/(m-k)}, \qquad (22)$$

where

$$\hat{\mathcal{F}}(C_1^\star, \dots, C_k^\star) = \sum_{j=1}^{k} \sum_{a_i \in \pi_j^\star} \sqrt{D(C_j^\star(S_j^\star, r_j^\star), a_i)},$$

$$\hat{\mathcal{G}}(C_1^\star, \dots, C_k^\star) = \sum_{j=1}^{k} |\pi_j^\star| \delta^2(C^\star(S^\star, r^\star), C_j^\star(S_j^\star, r_j^\star)) = \sum_{j=1}^{k} |\pi_j^\star| \left(\|S^\star - S_j^\star\| + |r^\star - r_j^\star| \right)^2$$

where $C_j^\star = (S_j^\star, r_j^\star)$ are center-circles of clusters $\pi_j^\star$, and $C^\star = (S^\star, r^\star)$ is a center-circle of the entire set $\mathcal{A}$. In that way, by definition (22), the property of non-dimensionality for CHC index has also been retained.

More compact and better separated clusters in an optimal partition will result in a greater CHC index.

(c) Simplified Silhouette Width Criterion in the standard case for the optimal partition $\Pi^\star = \{\pi_1^\star, \dots, \pi_k^\star\}$ of the set $\mathcal{A}$ with k clusters $\pi_1^\star, \dots, \pi_k^\star$ and the corresponding centers $c_1^\star, \dots, c_k^\star \in \mathbb{R}^2$ is defined in the following way (see e.g. (Kaufman and Rousseeuw, 2005; Vendramin et al., 2009)). For each $a_i \in \mathcal{A} \cap \pi_r^\star$ the numbers

$$\alpha_{ir} = \|c_r^\star - a_i\|, \quad \beta_{ir} = \min_{\substack{s=1,\dots,k \\ s \neq r}} \|c_s^\star - a_i\|,$$

$$s_i = \frac{\beta_{ir} - \alpha_{ir}}{\max\{\alpha_{ir}, \beta_{ir}\}},$$

are calculated and the Simplified Silhouette Width Criterion is defined as the average of s_i: $\mathtt{SSW}(k) = (1/|\mathcal{A}|) \cdot \sum_{a_i \in \mathcal{A}} s_i$.

Analogously, the *Simplified Silhouette Width criterion for circle-centers* will be defined. For each $a_i \in \mathcal{A} \cap \pi_r^\star$ the numbers

$$\hat{\alpha}_{ir} = D(C_r^\star, a_i), \quad \hat{\beta}_{ir} = \min_{\substack{s=1,\dots,k \\ s \neq r}} D(C_s^\star, a_i),$$

$$\hat{s}_i = \frac{\hat{\beta}_{ir} - \hat{\alpha}_{ir}}{\max\{\hat{\alpha}_{ir}, \hat{\beta}_{ir}\}},$$

are calculated and the *Simplified Silhouette Width Criterion for circle-centers* is defined as the average of $\hat{s}_i$: $\mathtt{SSWC}(k) = (1/|\mathcal{A}|) \cdot \sum_{a_i \in \mathcal{A}} \hat{s}_i$.

More compact and better separated clusters in an optimal partition will result in a greater SSWC number.

5. Numerical examples

Example 2. *The iterative process of Algorithm 2 for k ≤ 8 has been implemented for the data from Example 1.*

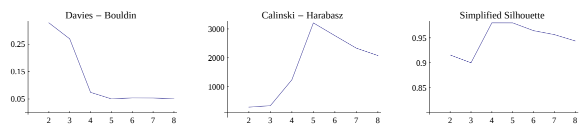

Figure 2: Indexes for the data from Example 2

The configuration of the obtained circles in iterations after Step 2 (application of the DIRECT algorithm) and Step 3 (application of the KCC algorithm) for the first four iterations is shown previously in Fig 1. The graphs of the used indexes are shown in Fig. 2. The DBC index gives a certain priority to the partition with 5 clusters, the CHC index gives a clear priority to the same partition, whereas the SSWC criterion gives an insignificant priority to the partition with 4 clusters compared to the partition with 5 clusters.

Therefore, it can be said that all the indexes used in this example have pointed to the partition with 5 clusters as the most appropriate partition, whereby the Hausdorff distance between original and recognized circles is sufficiently small ($d_H(C, C^\star) = 0.05$).

Example 3. *Set C of* 4 *concentric circles with common center $S = (5, 5)$ and radii $r_i = .75, 1.5, 2.5, 3.5$ is given in the plane. Data point set $\mathcal{A}$ is constructed similarly to Example 1 and it consists of $m = 40+40+43+43$ random points around circles.*

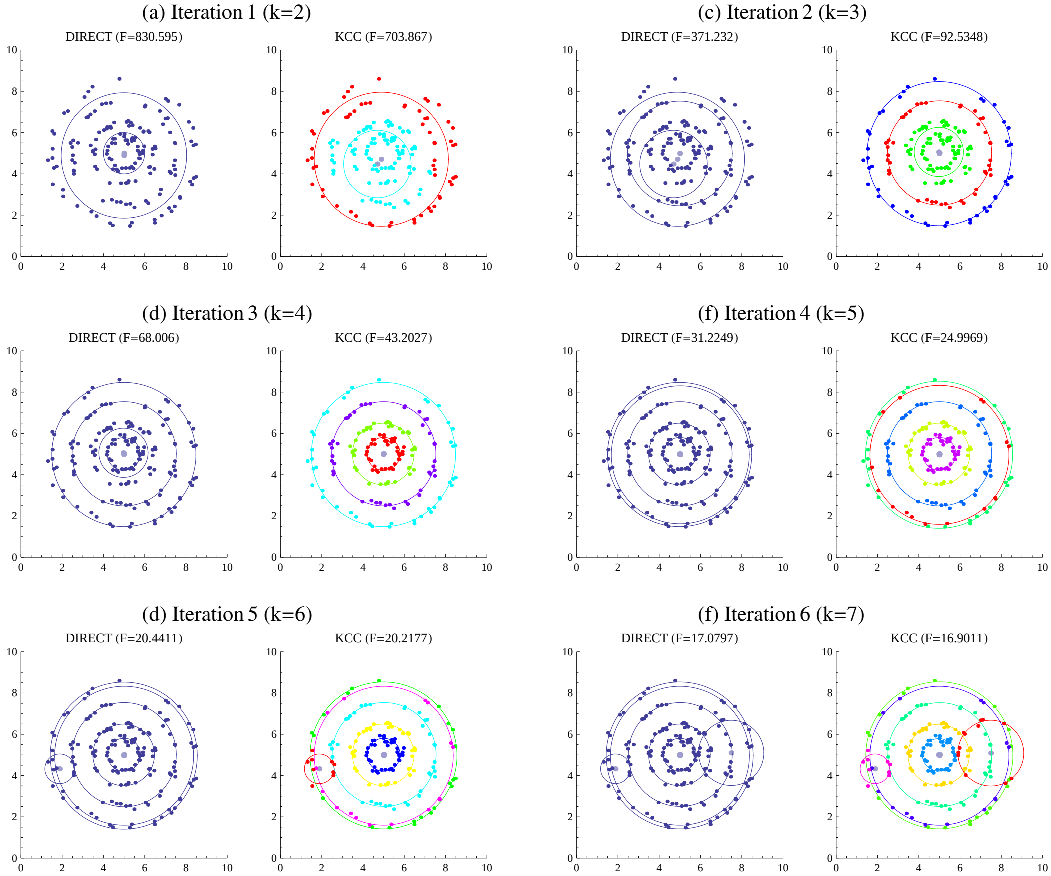

Figure 3: The flow of the iterative process of Algorithm 2 for the data from Example 3

The configuration of the obtained circles in iterations after Step 2 (application of the `DIRECT` algorithm) and Step 3 (application of the `KCC` algorithm) is shown in Fig. 3. The graphs of the used indexes are visible in Fig. 4. As can be seen, the `DBC` index and the `SSWC` criterion clearly point to the partition with 4 clusters as the most appropriate partition. However, as can be seen in (22) and in Fig. 4, in the case of concentric circles the `CHC` index is not usable.

The Hausdorff distance between original and recognized circles is also sufficiently small ($H_d(C, C^\star) = 0.05$).

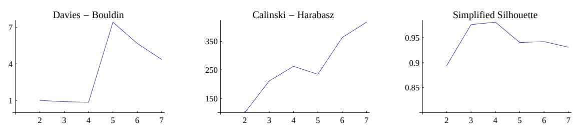

Figure 4: Indexes for the data from Example 3

The numerical characteristics of the proposed algorithms for circle detection were tested on the example of data originating from three different and well separated circles and on the example of data originating from three different mutually intersecting circles, whereby the number of circle data and their dissipation around circles was varied.

Remark 2. *For the purpose of comparison, the Hough Transform for circle detection was also used. The principle of the well-known circular Hough Transform is that for circles with equation* $(x-p)^2+(y-q)^2 = r^2$*, every point* (x, y) *corresponds to the surface of the cone in the* (p, q, r) *Hough circle space (the three-dimensional accumulator array) (Ballard, 1981). Finite detected values are found at the intersection of all cones. Hence, the Hough Transform for circle detection is based on recognizing the most intensive points in the Hough circle space. The Hough Transform is a method not sensitive to outliers.*

Example 4. *Let* $C = \{C_1 = ((2.5, 4), 1.8),\ C_2 = ((7, 3), 2.2),\ C_3 = ((5, 8), 1.2)\}$ *be the set of three well separated circles in the plane (see Fig. 5a). In the neighborhood of each circle* μ *random points (*$\mu \in \{20, 40, 60\}$*) were generated by using Gaussian distribution with covariance matrix* $Cov[\sigma] = \{\{\sigma^2, \sigma^2/2\}, \{\sigma^2/2, \sigma^2\}\}$*, (*$\sigma \in \{0.1, 0.2, 0.4\}$*). In this way, a data point set* $\mathcal{A} = \{a_i \in \mathbb{R}^2 : i = 1, \ldots, m\}$ *with* $m = 3\mu$ *random points is obtained. Based on these data, we reconstruct circles by applying Algorithm 2 with initial circle* $\hat{C} = ((3, 3), 1)$ *and by applying Matlab-function "imfindcircles" which finds circles by using circular Hough Transforms. The experiment was repeated* 100 *times.*

After two iterations of Algorithm 2 a set of circles $C^\star = \{C_1^\star, C_2^\star, C_3^\star\}$ is obtained. The quality of reconstruction is measured by using the Hausdorff distance between sets C and $C^\star$ according to Proposition 1. Basic numbers

(a) Well separated circles (b) Mutually intersecting circles

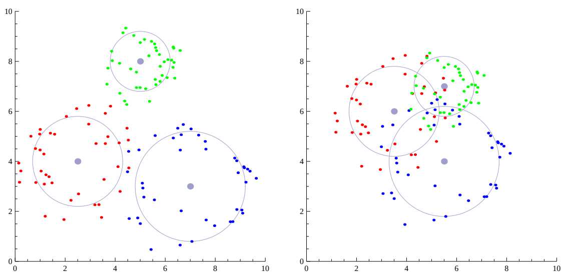

Figure 5: Data points from Example 4 with $\mu = 40$ and $\sigma = 0.4$

Table 1: Percentage of recognition of Algorithm 2 (basic numbers) and the Hough Transform (numbers in parentheses)

	Well separated circles			Mutually intersecting circles		
	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.4$	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.4$
$\mu = 20$	85 (1)	72 (1)	52 (0)	68 (5)	52 (3)	41 (2)
$\mu = 40$	97 (4)	94 (4)	64 (2)	89 (3)	84 (5)	60 (2)
$\mu = 60$	99 (1)	96 (1)	86 (2)	87 (4)	86 (2)	77 (2)

in Table 1 show the number of experiments in which Algorithm 2 recognized the set of circles ($d_H(C, C^\star) < 0.1$) well. The number of such cases is large and it depends on the number of data μ along the circle and parameter σ in the covariance matrix. This proves a very good performance of the proposed algorithm. The numbers in parentheses show the number of experiments in which the Hough Transform for circle detection recognized the circles well. The number of such cases is very small. It turned out that the Hough Transform is not designed for such case of a relatively small number of data points. Only in special cases, when it was possible to adjust certain values of a Matlab option of “imfindcircles” (e.g., sensitivity factor), a good reconstruction of circles was obtained.

A similar experiment is repeated for three mutually intersecting circles (see Fig. 5b), and the obtained results are similar.

6. Conclusions

The method and the algorithm for solving the multiple circle detection problem on the basis of the given data point set $\mathcal{A} \subset \mathbb{R}^2$ gives either a globally optimal k-partition or a locally optimal k-partition close to the global one. It can be seen that center-circles obtained in this way present a good approximation of the original circles. The method has been tested on several numerical examples with synthetic data on the basis of our own software done by *Mathematica*. The Hausdorff distance between original and recognized circles shows an acceptable quality of reconstruction. It is important to note that minimizing functions (12)-(13) can have several local and global minima and that, precisely for this reason, it is necessary to use some global optimization method. Since minimizing functions are Lipschitz continuous functions, the `DIRECT` algorithm was found to be very useful.

Assuming that we have a good approximation of the optimal partition, indexes adapted to the multiple circle detection problem recognized the appropriate number of clusters in a partition well. Thereby, algebraic distance (9) from a point to a circle and the Hausdorff distance between two circles have been used in the paper. If any other distance measure from a point to a circle and the distance between two circles were used, then the corresponding indexes would have to be redefined. Also, some other well-known indexes could be considered.

Finally, the considered problem can be generalized such that ellipses in the plane $E_j = \{x \in \mathbb{R}^2 \colon \|x - C_j\|_{A_i} = 1\}$ are considered instead of circles, where C_j is the center of ellipse E_j and $A_i > 0$ is a positive definite symmetric matrix which determines the major and the minor axes lengths as well as the orientation of the ellipse. In that way, the ellipse is determined by its 5 parameters. The problem can be further generalized to a hyperellipse in $\mathbb{R}^n$ (Bezdek et al., 2005).

Acknowledgements

The authors would like to thank anonymous referees and journal editors for their careful reading of the paper and insightful comments that helped us improve the paper. This work was supported by the Ministry of Science, Education and Sports, Republic of Croatia.

References

Ahn, S.J., Rauh, W., Warnecke, H.J., 2001. Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola. Pattern Recognition 34, 2283–2303.

Ballard, D.H., 1981. Generalizing the Hough transform to detect arbitrary shapes. Pattern Recognition 13, 111–122.

Bagirov, A.M., Ugon, J., 2005. An algorithm for minimizing clustering functions. Optimization 54, 351–368.

Bagirov, A.M., Ugon, J., Webb, D., 2011. Fast modified global k-means algorithm for incremental cluster construction. Pattern Recognition 44, 866876.

Bezdek, J. C, Keller, J., Krisnapuram, R., Pal, N. R, 2005. Fuzzy models and algorithms for pattern recognition and image processing. Springer, New York.

Chen, X.D., Ma, W., Xu, G., Paul, J.C., 2010. Computing the hausdorff distance between two B-spline curves. Computer-Aided Design 42, 11971206.

Chernov, N., 2010. Circular and linear regression: Fitting circles and lines by least squares. Volume 117 of *Monographs on Statistics and Applied Probability*. Chapman & Hall/CRC.

Chung, K.L., Huang, Y.H., Shen, S.M., Yurin, A.S.K.D.V., Semeikina, E.V., 2012. Efficient sampling strategy and refinement strategy for randomized circle detection. Pattern Recognition 45, 252-263.

Cuevas, E., Osuna-Enciso, V., Wario, F., Zaldívar, D., Pérez-Cisneros, M., 2012. Automatic multiple circle detection based on artificial immune systems. Expert Systems with Applications 39, 713-722.

Drezner, Z., Steiner, S., Wesolowsky, G.O., 2002. On the circle closest to a set of points. Computers & Operations Research 29, 637–650.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Graphics Image Process 24, 381-395.

Finkel, D.E., 2003. DIRECT Optimization Algorithm User Guide. Center for Research in Scientific Computation. North Carolina State University. `http://www4.ncsu.edu/definkel/research/index.html`.

Gan, G., Ma, C., Wu, J., 2007. Data Clustering: Theory, Algorithms, and Applications. SIAM, Philadelphia.

Grbić, R., Nyarko, E.K., Scitovski, R., 2012. A modification of the DIRECT method for Lipschitz global optimization for a symmetric function. Journal of Global Optimization 57, 1193–1212.

Jones, D.R., Perttunen, C.D., Stuckman, B.E., 1993. Lipschitzian optimization without the Lipschitz constant. Journal of Optimization Theory and Applications 79, 157–181.
Jüttler, B., 2001. Bounding the hausdorff distance between implicitly defined and/or parametric curves, in: Lyche, T., Schumaker, L.L. (Eds.), Mathematical Methods for Curves and Surfaces: Oslo 2000, Vanderbilt University Nashville. 223–232.
Kaufman, L., Rousseeuw, P.J., 2005. Finding groups in data: An introduction to cluster analysis. Jonh Wiley & Sons, Hoboken.
Kim, H.S., Kim, J.H., 2001. A two-step circle detection algorithm from the intersecting chords. Pattern Recognition Letters 22, 787-798.
Kogan, J., 2007. Introduction to Clustering Large and High-dimensional Data. Cambridge University Press.
Leisch, F., 2006. A toolbox for k-centroids cluster analysis. Computational Statistics & Data Analysis 51, 526–544.
Liao, S.H., Chu, P.H., Hsiao, P.Y., 2012. Data mining techniques and applications a decade review from 2000 to 2011. Expert Systems with Applications 39, 11303-11311.
Likas, A., Vlassis, N., Verbeek, J.J., 2003. The global k-means clustering algorithm. Pattern Recognition 36, 451-461.
Morales-Esteban, A., Martínez-Álvarez, F., Troncoso, A., Justo, J., Rubio-Escudero, C., 2010. Pattern recognition to forecast seismic time series. Expert Systems with Applications 37, 8333-8342.
Neumaier, A., 2004. Complete search in continuous global optimization and constraint satisfaction. Acta Numerica 13, 271–369.
Nievergelt, Y., 2002. A finite algorithm to fit geometrically all midrange lines, circles, planes, spheres, hyperplanes, and hyperspheres. Numerische Mathematic 91, 257-303.
Pintér, J.D., 1996. Global Optimization in Action (Continuous and Lipschitz Optimization: Algorithms, Implementations and Applications). Kluwer Academic Publishers, Dordrecht.
Qiao, Y., Ong, S.H., 2004. Connectivity-based multiple-circle ftting. Pattern Recognition 37, 755-765.
Sabo, K., Scitovski, R., Vazler, I., 2013. One-dimensional center-based l_1-clustering method. Optimization Letters 7, 5–22.
Sabo, K., Scitovski, R., Vazler, I., Zekić-Sušac, M., 2011. Mathematical models of natural gas consumption. Energy Conversion and Management 52, 1721–1727.
Scitovski, R., Scitovski, S., 2013. A fast partitioning algorithm and its application to earthquake investigation. Computers and Geosciences 59, 124–131.
Sergeyev, Y.D., Famularo, D., Pugliese, P., 2001. Index branch-and-bound algorithm for lipschitz univariate global optimization with multiextremal constraints. Journal of Global Optimization 21, 317–341.
Sergeyev, Y.D., Kvasov, D.E., 2011. Lipschitz global optimization, in: Cochran, J. (Ed.), Wiley Encyclopedia of Operations Research and Management Science. Wiley, New York. volume 4, 2812–2828.
Song, Q., Yang, X., Soh, Y.C., Wang, Z.M., 2010. An information-theoretic fuzzy c-spherical shells clustering algorithm. Fuzzy Sets and Systems 161, 1755-1773.
Späth, H., 1983. Cluster-Formation und Analyse. R. Oldenburg Verlag, München.
Tan, P.N., Steinbach, M., Kumar, V., 2006. Introduction to Data Mining. Wesley.
Teboulle, M., 2007. A unified continuous optimization framework for center-based clustering methods. Journal of Machine Learning Research 8, 65–102.
Theodoridis, S., Koutroumbas, K., 2009. Pattern Recognition. Academic Press, Burlington. 4th edition.
Vendramin, L., Campello, R.J.G.B., Hruschka, E.R., 2009. On the comparison of relative clustering validity criteria, in: Proceedings of the SIAM International Conference on Data Mining, SDM 2009, April 30 - May 2, 2009, Sparks, Nevada, USA, SIAM. 733–744.

Appendix

Proof of Proposition 1

Without loss of generality, it can be supposed that

$$S_1 = (0,0), \quad S_2 = (d,0), \quad d \geq 0, \quad r_2 \geq r_1 > 0. \quad (23)$$

It should be proved that

$$d_H(C_1, C_2) = d + r_2 - r_1. \quad (24)$$

Let us notice the following points on circle C_1, i.e. C_2

$$X_1 = (-r_1, 0),\ X_2 = (r_1, 0) \in C_1,$$
$$Y_1 = (d - r_2, 0),\ Y_2 = (d + r_2, 0) \in C_2.$$

The Hausdorff distance between circles C_1, C_2 is attained precisely on one of the pairs of points (X_i, Y_j), $i, j = 1, 2$. That is why (17) becomes

$$d_H(C_1, C_2) = \max\{ \quad (25)$$
$$\max\{\min\{\|X_1 - Y_1\|, \|X_2 - Y_1\|\}, \min\{\|X_1 - Y_2\|, \|X_2 - Y_2\|)\}\},$$
$$\max\{\min\{\|(X_1 - Y_1\|, \|X_1 - Y_2\|\}, \min\{\|X_2 - Y_1\|, \|X_2 - Y_2\|\}\}\ \}.$$

In particular, if $S_1 = S_2$, i.e. $d = 0$, the Proposition

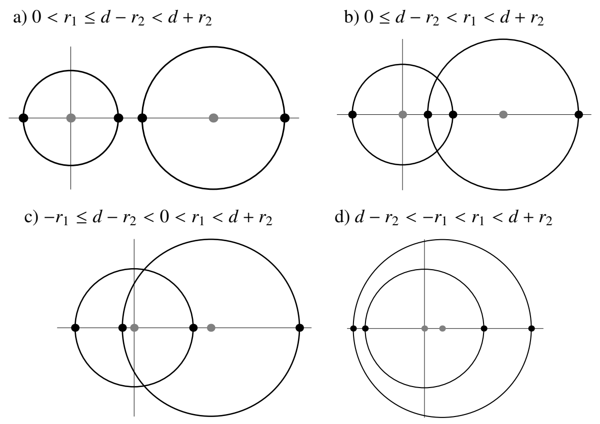

Figure 6: Possible positions of circles C_1, C_2 with conditions in (23)

assertion is true, because then $d_H = r_2 - r_1$.

Therefore, suppose that $d > 0$ and consider all possible positions of circles C_1, C_2, by which conditions (23) remain fulfilled (see Fig. 6):

a) $0 < r_1 \leq d - r_2 < d + r_2$
b) $0 \leq d - r_2 < r_1 < d + r_2$
c) $-r_1 \leq d - r_2 < 0 < r_1 < d + r_2$
d) $d - r_2 < -r_1 < r_1 < d + r_2$

It should be proved that equality (24) holds in each of the mentioned cases. Only the proof of assertion in case *a)* will be considered. In other cases the proof is analogue.

Suppose that $0 < r_1 \leq d - r_2 < d + r_2$ (see Fig. 6a). Then there holds

$$\min\{\|X_1 - Y_1\|, \|X_2 - Y_1\|\} = \|X_2 - Y_1\| = d - r_2 - r_1,$$
$$\min\{\|X_1 - Y_2\|, \|X_2 - Y_2\|\} = \|X_2 - Y_2\| = d + r_2 - r_1,$$

from which it follows that

$$\max\{\min\{\|X_1 - Y_1\|, \|X_2 - Y_1\|\}, \min\{\|X_1 - Y_2\|, \|X_2 - Y_2\|\}\} = $$
$$= d + r_2 - r_1. \quad (26)$$

Similarly, from

$$\min\{\|X_1 - Y_1\|, \|X_1 - Y_2\|\} = \|X_1 - Y_1\| = d - r_2 + r_1,$$
$$\min\{\|X_2 - Y_1\|, \|(X_2 - Y_2\|\} = \|X_2 - Y_1\| = d - r_2 - r_1,$$

follows

$$\max\{\min\{\|X_1 - Y_1\|, \|X_1 - Y_2\|\}, \min\{\|X_2 - Y_1\|, \|(X_2 - Y_2\|\}\} = $$
$$= d - r_2 + r_1. \quad (27)$$

By using (26) and (27) in (25) the required equality (24) for case *a)* is obtained.