EVOLVING CLUSTERING VIA THE DYNAMIC DATA ASSIGNING ASSESSMENT ALGORITHM

O. Georgieva[1], F. Klawonn[2]

[1] *Institute of Control and System Research - Bulgarian Academy of Sciences, P.O. Box 79, 1113 Sofia, Bulgaria*
phone: +359 2 979 2052, e-mail: ogeorgieva@icsr.bas.bg

[2] *Department of Computer Science, University of Applied Sciences Braunschweig/Wolfenbuettel, Germany*

Key Words: Clustering, Noise clustering, Evolving clustering, Fuzzy C-means algorithm.

1. Introduction

Although the original intention of cluster analysis is to partition a data set into "meaningful" substructures, clustering is often applied for other purposes. For instance, when fuzzy cluster analysis is applied in the context of generating fuzzy rules from data, it is very often used as a segmentation technique that simply partitions the data (in a fuzzy way), without putting a strong emphasis on well distinguished clusters. In other applications, like for example analysing gene expression data or astrophysics data, it is not necessary to partition the data into meaningful clusters, but to identify one or a few interesting clusters that might only cover a small portion of the data.

Following the idea to search for just one cluster at a time a prototype-based clustering algorithm named Dynamic Data Assigning Assessment (DDAA) was recently proposed [8,9]. It is based on the Noise clustering technique and finds single good clusters one by one and at the same time it separates the noise data. Two algorithm versions – hard and fuzzy clustering, are realizable according to the applied distance metric. The method can be used for two purposes: either in the sense of standard cluster analysis to determine the number of clusters automatically or in order to identify one or a few clusters that might cover only a portion of the data set.

The above mentioned algorithm properties were used in order to develop an extended

DDAA algorithm that is capable to separate a new data stream added to the data set. The evolving DDAA algorithm assigns every new data point to an already determined good cluster or alternatively to the noise cluster. By that it checks whether the new data collection provides a new good cluster(s) and thus changes the data structure. The assignment could be done in hard or fuzzy sense.

The paper is organised as follows. The second section briefly reviews the necessary background on objective function-based clustering, the concept of noise clustering that we exploit in our approach and the underlying idea of the DDAA algorithm itself. The evolving version of the algorithm is presented in detail in Section 4. Some conclusions are given in the fifth section.

2. Basic concepts and DDAA algorithm

Objective function-based clustering aims at minimizing an objective function J that indicates a kind of fitting error of the clusters to the data. In this function, the number of clusters has to be fixed in advance. The underlying objective function for most of the clustering algorithms is [1]:

$$J = \sum_{i=1}^{c}\sum_{k=1}^{N}(u_{ik})^m d_{ik}^2, \tag{1}$$

where N is the number of data points; c is the number of clusters; u_{ik} and d_{ik} denote correspondingly the membership degree and the distance of the k-th point $x_k = [x_{k1}, x_{k2}, ..., x_{kn}]$, k = 1, ..., N, to the i-th cluster prototype, i = 1, ..., c; $m \in [1,\infty)$ is the weighted exponent coefficient which determines how much clusters may overlap. In order to avoid the trivial solution assigning no data to any cluster, i.e. setting all u_{ik} to zero, and to avoid empty clusters, the following constraints are introduced:

$$u_{ik} \in [0,1], \qquad 1 \le i \le c, \qquad 1 \le k \le N \tag{2.a}$$

$$\sum_{i=1}^{c} u_{ik} = 1, \qquad 1 \le k \le N \tag{2.b}$$

$$0 < \sum_{i=1}^{N} u_{ik} < N, \qquad 1 \le i \le c \tag{2.c}$$

When we choose the fuzzifier m=1 we have $u_{ik} \in \{0,1\}$ at a minimum of the objective function (1) and the resulting partition will be crisp.

The parameters to be optimized are the membership degrees u_{ik} and the cluster parameters which finally determine the distance values d_{ik}. Each cluster is represented by a cluster prototype. In the simplest case, the cluster prototype is a single vector called cluster centre $v_i = [v_{i1}, v_{i2}, ..., v_{in}]$, $i = 1, ..., c$. The distance of a data point k to the i-th cluster is defined by a positive definite symmetric matrix A_i and the cluster centre as follows:

$$d_{ik}^2 = \|x_k - v_i\|_{A_i}^2 = (x_k - v_i) A_i (x_k - v_i)^T . \quad (3)$$

The matrix A_i is defined differently according to the applied objective-function based clustering algorithm [1,2,3,4,5].

The arbitrary noise points that do not belong to any comprehensible class have to be taken into account. The successful solution to deal with the noise in the data set is to collect the noise points in one single cluster [7]. For this purpose a virtual noise prototype with no parameters to be adjusted is introduced that has always the same (large) distance δ to all points in the data set.

Let cluster number c be the noise cluster. Then, by definition we have

$$d_{ck} = \delta, \ \forall \ k. \quad (4)$$

The remaining c-1 clusters are assumed to be the good clusters in the data set. The objective function J_{noise} that considers the noise cluster is defined in the same manner as in the general scheme for the clustering minimization functional (1) i.e. $J_{noise} \equiv J$, but with some additional specifications. The distances of every point x_k, $k = 1, ..., N$ are defined by (3) for all clusters i, $i = 1, ..., c-1$ and by

$$d_{ck}^2 = \delta^2 \text{ for } i = c . \quad (5)$$

The objective function J_{noise} has the global minimum for a fixed noise distance δ only if:

a) for hard noise clustering (i.e. $m = 1$) the membership degrees are:

$$u_{ik} = 0 \text{ for } \forall i \neq j \text{ and} \quad (6)$$

$$u_{jk} = 1 \text{ for } j \text{ such that } d_{jk} = \min(d_{ik}, \ i = 1, ..., c).$$

b) for fuzzy noise clustering ($m > 1$) the membership degrees are

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} (d_{ik} / d_{jk})^{2/(m-1)}} \quad (7)$$

and the cluster centres of the good clusters are defined by the weighted mean value:

$$v_i = \frac{\sum_{k=1}^{N} (u_{ik})^m x_k}{\sum_{k=1}^{N} (u_{ik})^m}, \quad \text{for } i = 1, ..., c-1. \tag{8}$$

The specification of the noise distance δ is a matter of consideration for the particular data set. If δ is chosen too small, then most of the data points will be classified as noise, while for a large δ value even outliers will be assigned to good clusters.

Let us assume that the data set consists of only one good cluster among a certain number of noise data considered as a noise cluster. Thus, the two clusters could be separated in minimizing the objective function J_{noise}, which is simplified for the case c=2 to the following form:

$$J_{noise} = \sum_{k=1}^{N} (u_{1k})^m d_k^2 + \delta^2 \sum_{k=1}^{N} (u_{2k})^m \ . \tag{9}$$

The distance d_k denotes the distance between every point x_k and the centre v_1 of the single good cluster. The membership degrees are calculated for a fixed δ :

a) for hard noise clustering as

$u_{1k} = 1$ and $u_{2k} = 0$ if the k-th point belongs to the good cluster, i.e. $d_k \le \delta$ and (10)

$u_{1k} = 0$ and $u_{2k} = 1$ if the k-th point belongs to the noise cluster, i.e. $d_k > \delta$. (11)

b) for fuzzy noise clustering the membership degree to the good cluster is defined as

$$u_{1k} = \frac{1}{1 + \left(\frac{d_k}{\delta} \right)^{2/(m-1)}} \tag{12}$$

and the membership degree to the noise cluster is correspondingly defined as:

$$u_{2k} = \frac{1}{1 + \left(\frac{\delta}{d_k} \right)^{2/(m-1)}} \ . \tag{13}$$

Since we are still in the framework of probabilistic clustering the following statement is valid to both clustering variants:

$$u_{1k} = 1 - u_{2k} \ , \quad k = 1,...,N \ . \tag{14}$$

The proposed method separates one cluster at a time based on the concept of noise clustering via dynamical decrease of the noise distance. Thus, by this approach, it is not necessary to seek for a proper noise distance.

The procedure starts by choosing a large noise distance, for instance the diameter of the

data set, so that all data points are assigned to the good cluster and no data are considered as noise. Then, decreasing the noise distance stepwise by a prescribed decrement $\Delta\delta$, for each $\delta = \delta_j$ value we can determine the number $N_{in}(j)$ of data belonging to the good cluster according to the membership degrees (crisp or fuzzy) to the good cluster. The index *j* denotes the current step of the noise distance reduction. At every noise distance δ_j the distance d_k, $k = 1,...,N$ is calculated by (3). If the distance is less or equal to δ_j the current point x_k is assigned to the good cluster, if not – the point is separated to the noise cluster. It is obvious that by decreasing the distance δ a process of 'loosing' data, i.e. assigning them to the noise cluster will begin. Continuing to decrease the noise distance, we will start to separate points from the good cluster and add them to the noise cluster until the good cluster will be entirely empty as all data will be assigned to the noise cluster.

The described dynamics of moving data from the good cluster to the noise cluster can be characterized by a curve showing the number of data points assigned to the good cluster over the varying noise distance. The velocity $\Delta N_{in}(j)$ via the noise distance alteration is also evaluated:

$$\Delta N_{in}(j) = N_{in}(j-1) - N_{in}(j). \tag{15}$$

Note that not the time is used for the δ-axis so these curves must be viewed from right to left, when we want to observe the behaviour over time.

It is clear that if we loose actual noise points (i.e. from a region of low data density) the curve will almost remain in a plateau, whereas a strong slope should be observed when data from an actual cluster (with higher density than the noise data) are moved to the noise cluster. In the general case of a complex data set, we will have a number of plateaus and a number of strong slopes in the curve.

The peaks obtained from the curve ΔN_{in} correspond to the slopes in the N_{in} curve. The area of every peak is proportional to the number of points that are separated to the noise cluster within the current slope. Only significant peaks, whose area is larger than a predefined threshold *tol* are of a real concern:

$$S(s) = \sum_{j:\delta_{\min}(s) \le \delta_j \le \delta_{\max}(s)} \Delta N_{\text{in}}(j) > tol\,. \tag{16}$$

The last significant peak (the left-most one in $\Delta N_{\text{in}}(j)$ curve) occurs when the data points of the last data group are moved to the noise cluster. These data define a cluster that we remove from the whole data set. The other significant peaks also correspond to phases where at least one

comprehensible cluster is shifted to the noise cluster. The data of the none significant peaks i.e. of the (non-)plateau phases should be considered as noise data. The whole procedure is applied again to the reduced data set and repeated until no more significant peaks are identified.

The algorithm can also be applied in the context of fuzzy noise clustering if the membership degree is calculated by (12). A data point is assumed to belong to the good cluster if its membership degree is bigger than a predefined value μ_{tol} . However, being stricter in the identification of proper clusters the prescribed membership value should be increased and for a more tolerant identification, it should be decreased.

The Dynamic Data Assigning Assessment cluster identification algorithm is summarazed as follows:

Step 1: Specify the decrement step $\Delta\delta$ and threshold *tol*. If fuzzy DDAA clustering is applied specify μ_{tol} .

Step 2. Compute the curves N_{in} and ΔN_{in} decreasing δ by the prescribed decrement $\Delta\delta$.

Step 3: Find all peaks of ΔN_{in} curve and select the significant peaks.

Step 4: Separate one good cluster determined by the last significant peak.

Step 5: Subtract the separated points from the data set and repeat the procedure from ***Step 2*** for the remaining data points until no significant peaks could be found.

As in most clustering algorithms we normalize the data set in advance in order to let each feature have approximately the same influence on the distance used for clustering. Note that the proposed algorithm automatically determines the number of clusters, whereas in standard objective function-based clustering additional strategies have to be applied in order to define the number of clusters. The next section deals with the evolving evaluation of the new data collected to the data set.

The cluster selection abilities of DDAA are illustrated in Fig. 1 to artificial data set (Fig.1.a). Three significant peaks (Fig. 1.b) determined on the data assignment curves correspond to three good clusters that are selected in three algorithm passes (Fig. 1c).

3. Evolving DDAA

Let us suppose that the DDAA algorithm was applied to the data set $X=\{x_k\}$, $k=1, \ldots, N$ as it was described in the previous section. Thus, c-1 good clusters and their cluster centres have

been found. The noise data are collected in one noise cluster. Let us also suppose that a new input data stream currently is added to the data set. The task is to find the proper assignment of the new coming points, entering one by one, to the already determined cluster structure or alternatively to account for a new cluster structure as a result of the input stream information.

In general there are three possibilities for every new coming data point x_s: first, the new data could belong to a good cluster or secondly, it can be a noise data point (in case of fuzzy clustering, to belong to all clusters – good and fuzzy, but with different membership degrees). Finally, the new stream can change the data structure by forming additional clusters.

The solution of the above considerations will be discussed in detail. Let us first suppose that we have a fixed threshold δ_h that is a distance limit value under which a given point is specified as a part of the good cluster. It could be defined equal to the minimal cluster radius:

$$\delta_h = \min_{i=1,\ldots,c-1}(r_i), \tag{17}$$

where r_i is the cluster radius of the i-th cluster defined as the maximal distance between the cluster centre v_i and the points belonging to the cluster:

$$r_i = \max_{x_j \in i-\text{th cluster}} \left\| v_i - x_j \right\| \tag{18}$$

In order to evaluate the belonging of the current point to a cluster we need to assess the distance $d_s(i)$ between the new data point x_s and each cluster centre v_i, $i = 1,\ldots, c\text{-}1$ calculated in the sense of the distance measure applied in the DDAA algorithm:

$$d_s(i) = \sqrt{(v_i - x_s)^2_{A_i}} \quad i = 1,\ldots,c-1.. \tag{19}$$

The data point is assigned to this good cluster that is closest to the new data point i.e. has minimal distance $d_s(i)$ and the minimal distance is less then the threshold δ_h. If the data of the input stream can not be assigned to any good cluster than it is added to the noise cluster. However, in this case we should presume that the new coming data can change the cluster structure. In order to check this the DDAA algorithm is implemented over the new-formed noise cluster.

The assignment of x_s is provided easily according to the following procedure:

Step 1: Calculate $d_s(i)$, $i = 1, \ldots, c\text{-}1$.

Step 2: Find $d_n = \min_{i=1,\ldots,c-1}(d_s(i))$ that defines the closest good cluster p.

Step 3: If $d_n \le \delta_h$

assign x_s to the cluster p and update cluster p thorough calculating the new cluster centre value by (8).

If $d_n > \delta_h$

assign x_s to the noise cluster and apply DDAA algorithm to the data of the newly updated noise cluster. If new good cluster(s) is separated update the data structure.

end

If fuzzy clustering is applied then the procedure is carried out according to the membership degree $u_s(i)$ of x_s to every cluster presented in the data set. Let us suppose that a threshold membership degree u_h is given in advance. It could be equal to the predefined value μ_{tol} used in the fuzzy DDAA algorithm. The assignment procedure is rewritten as follows:

Step 1: Calculate $d_s(i)$, $i = 1, \ldots, c\text{-}1$.

Step 2: Find $d_n = \min_{i=1,\ldots c\text{-}1}(d_S(i))$ that defines the closest good cluster p.

Step 3: Determine $u_s(p)$ according to the eq. (7).

Step 4: If $u_s(p) \geq u_h$

assign x_s to the cluster p and update cluster p thorough calculating the new cluster centre value by (8).

If $u_s(p) < u_h$

assign x_s to the noise cluster and apply DDAA algorithm to the data of the newly updated noise cluster. If new good cluster(s) is separated update the data structure.

end

If the data are grouped in complex clusters that have different shape, orientation and density then the assignment level will vary for the distinct clusters. It will be difficult to set a proper assignment threshold neither for hard, nor for the fuzzy clustering variant. In such case the evolving algorithm is organized so that the DDAA algorithm is spread and applied over the currently formed data set consisting of the data of the good cluster p closest to the point x_s, all noise data and the current input data point x_s itself. The separated good clusters update the cluster structure. In the fuzzy alternative variant the new data set includes all good clusters to which the

current point x_s belongs with membership degree larger than the given threshold $u_h=\mu_{\text{tol}}$ used in the DDAA algorithm.

The clustering capabilities of the evolving extension of the DDAA algorithm are shown in Fig.2 where the new data stream marked by 'x' is added point by point to the already known data set given in Fig.1. All new points are added to the noise cluster according to the set threshold δ_h =1.5. After the eight data point a new cluster (fourth for the considered data set) is separated and by that updated the cluster structure is updated (Fig.2.b).

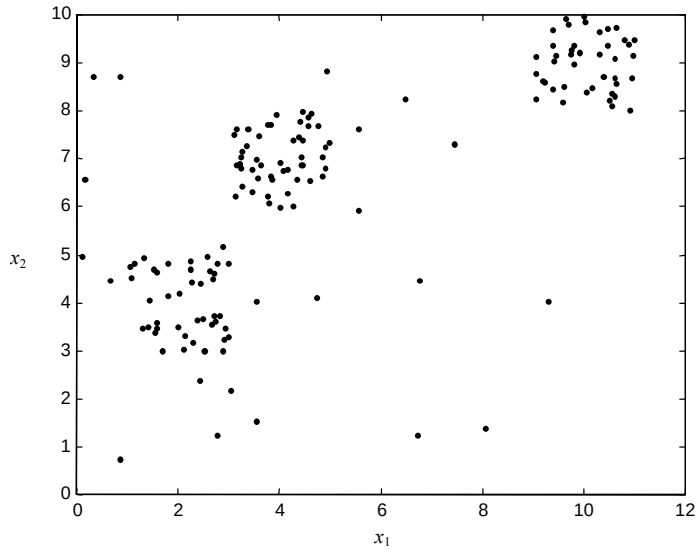
4. Conclusions

The cluster identification method presented here is based on the assessment of the dynamics of the number of points that are assigned to only one cluster through noise clustering of the data set by slowly changing the noise distance from a reasonable large to a sufficiently small value. It successfully assigns the new input data stream to the already known data structure or discovers new interesting groups of the data set that currently appeared. Two algorithm variants – hard and fuzzy, are presented in parallel. Additionally two alternatives – when there is or there is no preliminary information about the assignment threshold are considered.

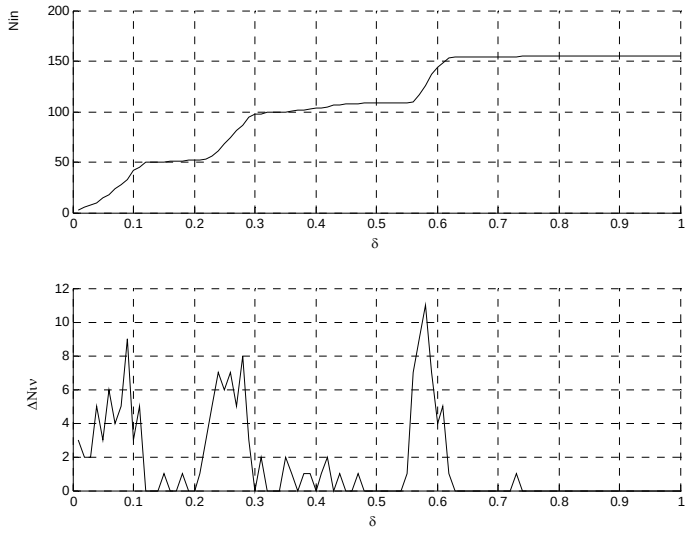
References

1. Bezdek, J.C., Pattern Recognition with Fuzzy Objective Function Algorithms Plenum Press, New York, 1981.
2. Gath, I., A.B. Geva, "Unsupervised optimal fuzzy clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v.7, pp. 773-781, 1989.
3. Gustafson, D., W. Kessel, Fuzzy clustering with a fuzzy covariance matrix, *Advances in fuzzy set theory and applications*, North-Holland, 1979, 605-620.
4. Hoeppner, F., F. Klawonn, R. Kruse, T. Runkler, Fuzzy Cluster Analysis, John Wiley & Sons, Chichester, 1999.
5. Keller A., F. Klawonn, Adaptation of cluster sizes in objective function based fuzzy clustering, in: C.T. Leondes (ed.): *Intelligent Systems: Technology and Applications* vol. IV: Database and Learning Systems. CRC Press, Boca Raton, 2003, 181-199.
6. Dave, R. H., "Characterization and detection of noise in clustering", *Pattern Recognition Letters* 12, 657-664, 1991.

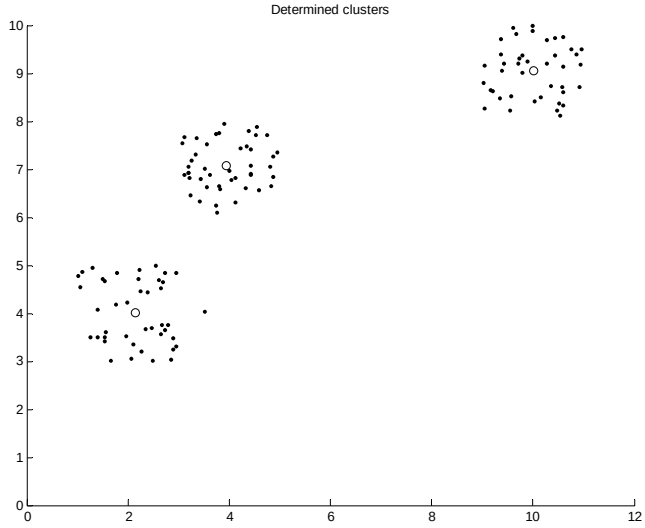
7. Dave, R. H., R. Krishnapuram, “Robust clustering methods: A unified view”, *IEEE Trans. Fuzzy Systems*, vol. 5, 1997, 270-293.
8. Georgieva O., F. Klawonn, A clustering algorithm for identification of single clusters in large data sets, in Proc. *East West Fuzzy Colloquium*, Sept. 8-10, Zittau, Germany, Heft 81, 2004, 118-12.
9. F. Klawonn, O.Georgieva, 2006, Identifying single clusters in large data Sets. In: J. Wang: Encyclopedia of Data Warehousing and Mining. Idea Group, Hershey, 582-585 (ISBN: 1591405572).



a) Data set

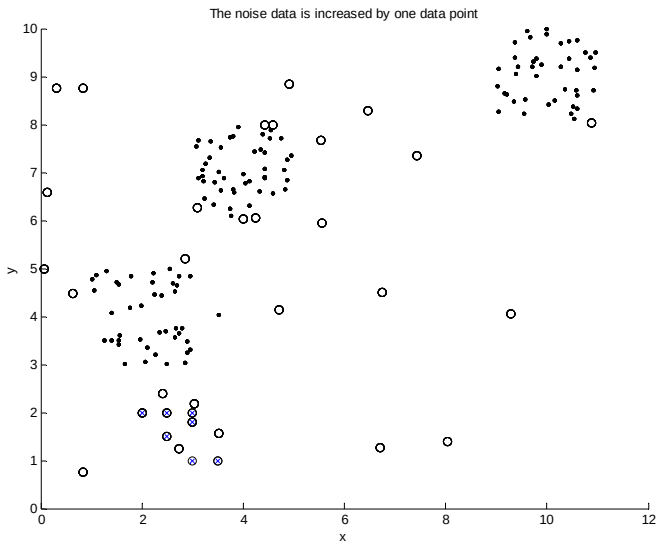


b) Data assignment dynamics in the first algorithm pass via normalized data set

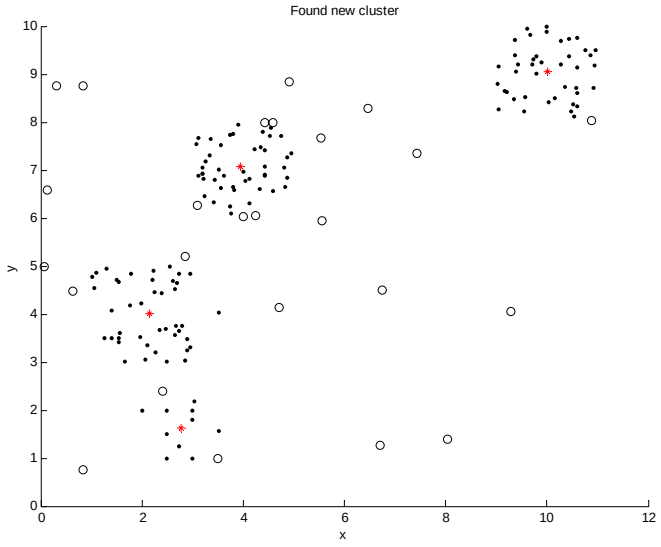


c) Determined three clusteres corresponding to three significant peaks

Figure 1. DDAA clustering algorithm applied to the artificial data set



a) The already clustered data are given by dots, the existed noise data are presented by circles, whereas new data stream is marked by circled ‘x’.



b) Selected additional (forth) cluster of the input data stream surrounded by the left noise data. The stars are cluster centeres.

Figure 2. Evolving DDAA algorithm