The Extended Hyperbolic Smoothing Clustering Method

Adilson Elias Xavier
Vinicius Layter Xavier
Dept. of Systems Engineering and Computer Science
Graduate School of Engineering (COPPE)
Federal University of Rio de Janeiro
Rio de Janeiro,RJ 21941-972, BRAZIL
e-mail: adilson@cos.ufrj.br

Abstract

The minimum sum-of-squares clustering problem is considered. The mathematical modeling of this problem leads to a $min-sum-min$ formulation which, in addition to its intrinsic bi-level nature, has the significant characteristic of being strongly nondifferentiable. To overcome these difficulties, the resolution method proposed adopts a smoothing strategy using a special C^{∞} differentiable class function. The final solution is obtained by solving a sequence of low dimension differentiable unconstrained optimization subproblems which gradually approach the original problem. The use of this technique, called Hyperbolic Smoothing, allows the main difficulties presented by the original problem to be overcome. A simplified version of the algorithm HSC containing only the essentials of the method is presented. For the purpose of illustrating both the reliability and the efficiency of the method, a set of computational experiments was performed, making use of traditional test problems described in the literature. Moreover, a set of computational results produced by a new extended version, XHSC Algorithm, based on an experimental prunning procedure supported by a partition of the set of observations in two non overlapping parts, are also presented, making using of the larger instances of Symmetric Traveling Salesman Problem (TSP)

Keywords: Cluster Analysis, Pattern Recognition, Min-Sum-Min Problems, Nondifferentiable Programming, Smoothing

1 Introduction

Cluster analysis deals with the problems of classification of a set of patterns or observations, in general represented as points in a multidimensional space, into clusters, following two basic and simultaneous objectives: patterns in the same clusters must be similar to another (homogeneity objective) and different from patterns of other clusters (separation objective), see Hartingan (1975) and Späth (1980).

Clustering is an important problem that appears in the broadest spectrum of applications, whose intrinsic characteristics engender many approaches to this problem, see Dubes and Jain (1976), Jain and Dubes (1988) and Hansen and Jaumard (1997).

In this paper, a particular clustering problem formulation is considered. Among many criteria used in cluster analysis, the most natural, intuitive and frequently adopted criterion is the minimum sum-of-squares clustering (MSSC). This criterion corresponds to the minimization of the sum-of-squares of distances of observations to their cluster means, or equivalently, to the minimization of within-group sum-of-squares. It is a criterion for both the homogeneity and the separation objectives, as, according to the Huygens Theorem, minimizing the within-cluster inertia of a partition (homogeneity within the cluster) is equivalent to maximizing the between-cluster inertia (separation between clusters).

The minimum sum-of-squares clustering (MSSC) formulation produces a mathematical problem of global optimization. It is both a nondifferentiable and a nonconvex mathematical problem, with a large number of local minimizers. It is one of the problems in the NP-hard class (Brucker (1978)).

In the cluster analysis scope, algorithms use, traditionally, two main strategies: hierarchical clustering methods and partition clustering methods (Hansen and Jaumard (1997) and Jain et alli (1999)). Hierarchical methods, essentially heuristic procedures, produce a hierarchy of partitions of the set of observations according to an agglomerative strategy or to a divisive one. In the former case, the general algorithm starts from an initial partition, in which each cluster contains one pattern, and successively merges two clusters on the basis of a similarity measure until all patterns are in the same cluster.

In the latter case, the general algorithm starts from an initial partition with all patterns in the same cluster and, by successive bipartitions, reaches a partition in which each cluster contains one single pattern. In both strategies, the best partition is chosen, by a suitable criterion, from the hierarchy of partitions obtained.

Partition methods, in general, assume a given the number of clusters and, essentially, seek the optimization of an objective function measuring the homogeneity within the clusters and/or the separation between the clusters. Heuristic algorithms of the exchange type, as the traditional k-means algorithm (Mc Queen (1967)) and variations thereof (Anderberg (1973) and Späth (1980)) are frequently used to find a local minimum of the objective function. However, any mathematical programming technique can be applied to solve the global optimization problem: dynamic programming (Jensen (1969)), branch and bound (Koontz, Narendra and Fukuraga (1975)), interior point algorithms (du Merle et alli (1997)), bilinear programming (Mangasarian (1997)), all kinds of metaheuristics (for example: Reeves (1993) or Pacheco and Valencia (2003)) and nonsmooth optimization (Bagirov and Yearwood (2004)).

The core focus of this paper is the smoothing of the $min - sum - min$ problem engendered by the modeling of the clustering problem. In a sense, the process whereby this is achieved is an extension of a smoothing scheme, called Hyperbolic Smoothing, presented in Santos (1997) for nondifferentiable problems in general, in Chaves (1997) for the $min-max$ problem and, more recently, in Xavier and Oliveira (2004) for the covering of plane domains by circles. This technique was developed through an adaptation of the hyperbolic penalty method originally introduced by Xavier (1982).

By smoothing we fundamentally mean the substitution of an intrinsically nondifferentiable two-level problem by a C^∞ differentiable single-level alternative. This is achieved through the solution of a sequence of differentiable subproblems which gradually approaches the original problem. In the present application, each subproblem, by using the Implicit Function Theorem, can be transformed into a low dimension unconstrained one, which, owing to its being indefinitely differentiable, can be comfortably solved by using the most powerful and efficient algorithms, such as conjugate gradient, quasi-Newton or Newton methods.

Although this paper considers the particular MSSC problem, it must be emphasized that the proposed methodology, Hyperbolic Smoothing, can be used for solving other clustering problem formulations as well.

This work is organized in the following way. A step-by-step definition of the clustering problem, directly connected to the presentation of the proposed hyperbolic smoothing approach, is presented in the next section. The new methodology is described in section 3. The algorithm and the illustrative computational results are presented in sections 4 and 5. Brief conclusions are drawn in section 5.

2 The Clustering Problem as a Min-Sum-Min Problem

Let $S = \{s_1, \ldots, s_m\}$ denote a set of m patterns or observations from an Euclidean n-space to be clustered into a given number q of disjoint clusters.

To formulate the original clustering problem as a $min - sum - min$ problem, we proceed as follows. Let $x_i, i = 1, \ldots, q$ be the centroids of the clusters, where each $x_i \in \mathbb{R}^n$. The set of these centroid coordinates will be represented by $X \in \mathbb{R}^{nq}$. Given a point s_j of S, we initially calculate the distance from s_j to the center in X that is nearest. This is given by

$$z_j = \min_{x_i \in X} \|s_j - x_i\|_2. \tag{1}$$

The most frequent measurement of the quality of a clustering associated to a specific position of q centroids is provided by the sum of the squares of these distances,

$$D(X) = \sum_{j=1}^{m} z_j^2. \tag{2}$$

The optimal placing of the centroids must provide the best quality of this measurement. Therefore, if X^* denotes an optimal placement, then the problem is

$$X^* = \underset{X \in \mathbb{R}^{nq}}{\operatorname{argmin}} \; D(X), \tag{3}$$

where X is the set of all placements of the q centroids. Using (1)–(3), we finally arrive at

$$X^* = \underset{X \in \mathbb{R}^{nq}}{\operatorname{argmin}} \; \sum_{j=1}^{m} \min_{x_i \in X} \; \|s_j - x_i\|_2^2. \tag{4}$$

3 Transforming the Problem

Problem (4) above can be formulated equivalently as

$$\begin{aligned} & minimize \; \sum_{j=1}^{m} z_j^2 \\ \text{subject to} \quad & z_j = \min_{i=1,\ldots,q} \; \|s_j - x_i\|_2, \quad j = 1, \ldots, m \end{aligned} \tag{5}$$

Considering its definition, each z_j must necessarily satisfy the following set of inequalities:

$$z_j - \|s_j - x_i\|_2 \; \leq 0, \quad i = 1, \ldots, q. \tag{6}$$

Substituting these inequalities for the equality constraints of problem (5), the relaxed problem becomes

$$\begin{aligned} & minimize \; \sum_{j=1}^{m} z_j^2 \\ \text{subject to} \quad & z_j - \|s_j - x_i\|_2 \leq 0, \quad j = 1, \ldots, m, \;\; i = 1, \ldots, q. \end{aligned} \tag{7}$$

Since the variables z_j are not bounded from below, the optimum solution of the relaxed problem will be $z_j = 0$, $j = 1, \ldots, m$. In order to obtain the desired equivalence, we must, therefore, modify problem (7). We do so by first letting $\varphi(y)$ denote $max\{0, y\}$ and then observing that, from the set of inequalities in (7), it follows that

$$\sum_{i=1}^{q} \varphi(z_j - \|s_j - x_i\|_2) = 0, \quad j = 1, \ldots, m. \tag{8}$$

For fixed j and assuming $d_1 < \cdots < d_q$ with $d_i = \|s_j - x_i\|_2$, Figure (1) illustrates the first three summands of (8) as a function of z_j.

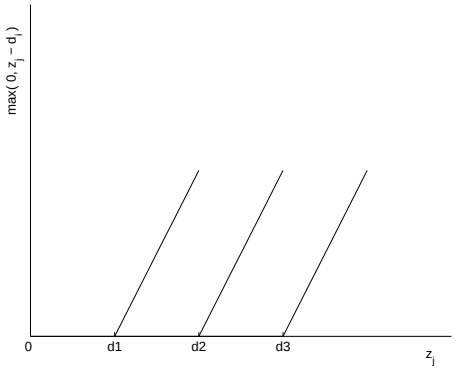

Figure 1: Summands in (8)

Using (8) in place of the set of inequality constraints in (7), we would obtain an equivalent problem maintaining the undesirable property that z_j, $j = 1, \ldots, m$ still has no lower bound. Considering, however, that the objective function of problem (7) will force each z_j, $j = 1, \ldots, m$, downward, we can think of bounding the latter variables from below by considering " $>$ " in place of " $=$ " in (8) and considering the resulting "non-canonical" problem

$$
\begin{aligned}
& minimize \ \sum_{j=1}^{m} z_j^2 \\
\text{subject to} \quad & \sum_{i=1}^{q} \varphi(z_j - \|s_j - x_i\|_2 \,) > 0, \quad j = 1, \ldots, m.
\end{aligned}
\tag{9}
$$

The canonical formulation can be recovered from (9) by perturbing (8) and considering the modified problem:

$$
\begin{aligned}
& minimize \ \sum_{j=1}^{m} z_j^2 \\
\text{subject to} \quad & \sum_{i=1}^{q} \varphi(z_j - \|s_j - x_i\|_2 \,) \ \geq \ \varepsilon \,, \quad j = 1, \ldots, m
\end{aligned}
\tag{10}
$$

for $\varepsilon > 0$. Since the feasible set of problem (9) is the limit of that of (10) when $\epsilon \rightarrow 0_+$, we can then consider solving (9) by solving a sequence of problems like (10) for a sequence of decreasing values for ε that approaches 0.

4 Smoothing the Problem

Analyzing the problem (10), the definition of function φ endows it with an extremely rigid nondifferentiable structure, which makes its computational solution very hard. In view of this, the numerical method we adopt for solving problem (1), takes a smoothing approach. From this perspective, let us define the function:

$$
\phi(y, \tau) = \left(y + \sqrt{y^2 + \tau^2} \, \right) / 2 \tag{11}
$$

for $y \in \mathbb{R}$ and $\tau > 0$.

Function ϕ has the following properties:

(a) $\phi(y, \tau) > \varphi(y), \quad \forall\, \tau > 0;$

(b) $\lim\limits_{\tau \to 0} \phi(y, \tau) = \varphi(y);$

(c) $\phi(., \tau)$ is an increasing convex C^∞ function in variable y.

Therefore, function ϕ constitutes an approximation of function φ. Adopting the same assumptions used in Figure 1, the first three summands of (8) and their corresponding smoothed approximations, given by (11), are depicted in Figure 2.

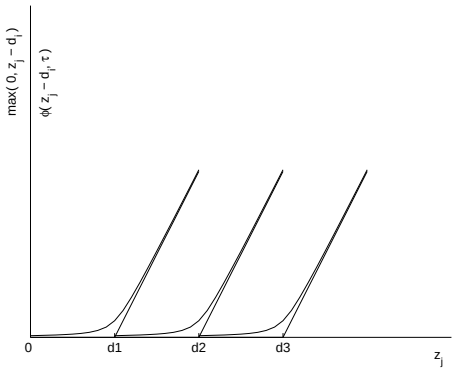

Figure 2: Original and smoothed summands in (8)

By using function ϕ in the place of function φ, the problem

$$minimize \sum_{j=1}^{m} z_j^2 \tag{12}$$

$$\text{subject to} \quad \sum_{i=1}^{q} \phi(z_j - \|s_j - x_i\|_2, \tau) \geq \varepsilon, \quad j = 1, \ldots, m.$$

is produced.

To obtain a differentiable problem, it is yet necessary to smooth the Euclidean distance $\|s_j - x_i\|_2$. For this purpose, let us define the function

$$\theta(\, s_j \,,\, x_i \,,\, \gamma \,) \;=\; \sqrt{\sum_{l=1}^{n} (s_j^l - x_i^l)^2 \;+\; \gamma^2} \tag{13}$$

for $\gamma > 0$.

Function θ has the following properties:

(a) $\lim\limits_{\gamma \to 0} \; \theta(\, s_j \,,\, x_i \,,\, \gamma \,) \;=\; \|s_j - x_i\|_2$;

(b) θ is a C^∞ function.

By using function θ in place of the distance $\|s_j - x_i\|_2$, the completely differentiable problem

$$\begin{aligned} & minimize \;\; \sum_{j=1}^{m} z_j^2 \\ \text{subject to} \;\; & \sum_{i=1}^{q} \phi(z_j - \theta(s_j, x_i, \gamma), \tau) \geq \varepsilon, \quad j = 1, \ldots, m. \end{aligned} \tag{14}$$

is now obtained.

So, the properties of functions ϕ and θ allow us to seek a solution to problem (10) by solving a sequence of subproblems like problem (14), produced by the decreasing of the parameters $\gamma \to 0$, $\tau \to 0$, and $\varepsilon \to 0$.

Since $z_j \geq 0$, $j = 1, \ldots, m$, the objective function minimization process will work for reducing these values to the utmost. On the other hand, given any set of centroids x_i, $i = 1, \ldots, q$, due to property (c) of the hyperbolic smoothing function ϕ, the constraints of problem (14) are a monotonically crescent function in z_j. So, these constraints will certainly be active and problem (14) will at last be equivalent to the problem:

$$minimize \ \sum_{j=1}^{m} z_j^2 \tag{15}$$

$$\text{subject to} \quad h_j(z_j, x) \ = \ \sum_{i=1}^{q} \phi(z_j - \theta(s_j, x_i, \gamma), \tau) \ - \ \varepsilon \ = \ 0, \quad j = 1, \ldots, m.$$

The dimension of variable domain space of problem (15) is $(nq+m)$. As, in general, the value of the parameter m, the cardinality of the set S of the observations s_j, is large, problem (15) has a large number of variables. However, it has a separable structure, because each variable z_j appears only in one equality constraint. Therefore, as the partial derivative of $h(z_j, x)$ with respect to z_j, $j = 1, \ldots, m$ is not equal to zero, it is possible to use the Implicit Function Theorem to calculate each component z_j, $j = 1, \ldots, m$ as a function of the centroid variables x_i, $i = 1, \ldots, q$. In this way, the unconstrained problem

$$minimize \ f(x) \ = \ \sum_{j=1}^{m} z_j(x)^2 \tag{16}$$

is obtained, where each $z_j(x)$ results from the calculation of a zero of each equation

$$h_j(z_j, x) \ = \ \sum_{i=1}^{q} \phi(z_j - \theta(s_j, x_i, \gamma), \tau) \ - \ \varepsilon \ = \ 0, \quad j = 1, \ldots, m. \tag{17}$$

Due to property (c) of the hyperbolic smoothing function, each term ϕ above is strictly increasing with variable z_j and therefore the equation has a single zero.

Again, due to the Implicit Function Theorem, the functions $z_j(x)$ have all derivatives with respect to the variables x_i, $i = 1, \ldots, q$, and therefore it is possible to calculate the gradient of the objective function of problem (16),

$$\nabla f(x) = \sum_{j=1}^{m} 2\, z_j(x)\, \nabla z_j(x) \tag{18}$$

where

$$\nabla z_j(x) = -\nabla h_j(z_j, x) \;/\; \frac{\partial h_j(z_j, x)}{\partial z_j}, \tag{19}$$

while $\nabla h_j(z_j, x)$ and $\partial h_j(z_j, x)/\partial z_j$ are obtained from equations (11), (13) and (17).

The above approach employs the same basic idea as Abadie and Carpentier (1969) for the development of the general reduced gradient algorithm, intended for the solution of the general nonlinear programming problem subject to equality constraints.

In this way, it is easy to solve problem (16) by making use of any method based on first order derivative information. At last, it must be emphasized that problem (16) is defined on a $(nq)-$dimensional space, so it is a small problem, since the number of clusters, q, is, in general, very small for real applications.

The solution of the original clustering problem can be obtained by using the Hyperbolic Smoothing Clustering Algorithm, described below in a simplified form.

Simplified HSC Algorithm

Initialization Step: Choose initial values: $x^0,\ \gamma^1\ ,\ \tau^1\ ,\ \varepsilon^1$.

Choose values $0 < \rho_1 < 1,\ 0 < \rho_2 < 1,\ 0 < \rho_3 < 1;$ let $k = 1$.

Main Step: Repeat until a stopping rule is attained

Solve problem (16) with $\gamma = \gamma^k,\ \tau = \tau^k$ and $\varepsilon = \varepsilon^k,$ starting at the initial point x^{k-1} and let x^k be the solution obtained.

Let $\gamma^{k+1} = \rho_1\, \gamma^k\ ,\ \tau^{k+1} = \rho_2\, \tau^k\ ,\ \varepsilon^{k+1} = \rho_3\, \varepsilon^k\ ,\ k := k + 1.$ ∎

Just as in other smoothing methods, the solution to the clustering problem is obtained, in theory, by solving an infinite sequence of optimization

problems. In the HSC algorithm, each problem that is minimized is unconstrained and of low dimension.

Notice that the algorithm causes τ and γ to approach 0, so the constraints of the subproblems it solves, given as in (14), tend to those of (10). In addition, the algorithm causes ε to approach 0, so, in a simultaneous movement, the solved problem (10) gradually approaches problem (9).

5 Computational Results

The computational results presented below were obtained from a preliminary implementation developed by Sousa (2005) for his M.Sc. thesis. The numerical experiments have been carried out on a PC Intel Celeron with 2.7GHz CPU and 512MB RAM. The programs are coded with Compac Visual FORTRAN, Version 6.1. The unconstrained minimization tasks were carried out by means of a Quasi-Newton algorithm employing the BFGS updating formula from the Harwell Library (http://www.cse.scitech.ac.uk/nag/hsl/). In the initialization step of the algorithm, the following choices were made: $\rho_1 = 1/2$, $\rho_2 = 1/2$, $\rho_3 = 1/2$, $\gamma^1 = 1/10$, $\tau^1 = 1/10$ and $\varepsilon^1 = 1/10$.

First, to illustrate the method, we present some computational results on one small test instance $(n = 2\,,\, q = 3\,,\, m = 41)$, originally presented in Demyanov (2004). The initial point, $x^0 = (2.1909362, 4.6229634, 5.5944853, 4.6315088, 3.8841220, 8.0480442)$, was taken over the inertia ellipse constructed by using the center of gravity of the observation points and the first and second eigen values and eigen vectors of the matrix $S^T S$, where S is the (m, n) matrix whose row j is formed with the components of the observation s_j.

Table 1 shows the sequence of points generated by the method in solving the first instance. The exact solution is presented in the last row. Column k represents the iteration numbers, while the pairs (a_i^k, b_i^k) represent the two coordinates of the centroids x_i^k, $i = 1, 2, 3$, and column $f(x^k)$ shows the objective function values. It is possible to observe a linear convergence of the sequence of the points generated by the algorithm to the optimum solution, with the controlled linear decreasing of the parameters γ^k , ε^k and τ^k. Figure 3 shows the original observation points of the Demyanov instance and the optimum clustering produced by the algorithm.

k	a_1^k	b_1^k	a_2^k	b_2^k	a_3^k	b_3^k	$f(x^k)$
1	2.1247416	6.0000027	4.8000021	1.9000180	5.1665170	7.6664565	99.411220
2	2.1249361	6.0000008	4.8000006	1.9000041	5.1666298	7.6666150	99.415309
3	2.1249842	6.0000002	4.8000002	1.9000010	5.1666644	7.6666538	99.416328
4	2.1249961	6.0000001	4.8000000	1.9000003	5.1666661	7.6666635	99.416582
5	2.1249990	6.0000000	4.8000000	1.9000001	5.1666665	7.6666659	99.416646
6	2.1249998	6.0000000	4.8000000	1.9000001	5.1666666	7.6666665	99.416661
7	2.1250000	6.0000000	4.8000000	1.9000000	5.1666667	7.6666666	99.416666
8	2.1250000	6.0000000	4.8000000	1.9000000	5.1666667	7.6666667	99.416667
-	2.1250000	6.0000000	4.8000000	1.9000000	5.1666667	7.6666667	99.416667

Table 1: Sequence of points generated in solving the Demyanov instance

Let $S_i, i = 1, \ldots, q$ be the partition of S generated by the centroids $x_i, i = 1, \ldots, q$

$$S = \bigcup_{i=1}^{q} S_i \tag{20}$$

$$S_{i_1} \cap S_{i_2} = \emptyset, \quad \forall\, i_1, i_2 = 1, \ldots, q, \quad i_1 \neq i_2, \tag{21}$$

$$S_i \neq \emptyset, \quad \forall\, i = 1, \ldots, q. \tag{22}$$

The center of gravity of the clusters is given by

$$v_i = \frac{1}{|S_i|} \sum_{s_j \in S_i} s_j, \quad \forall\, i = 1, \ldots, q. \tag{23}$$

For the Demyanov instance, the coordinates of v_i^k, $i = 1, 2, 3$, the centers of gravity of the clusters associated to the centroids x_i^k, $i = 1, 2, 3$, calculated by the HSC Algorithm at all iterations, assume a constant value: $v = (2.125, 6., 4.8, 1.9, 5.1666667, 7.6666667)$. So, already at the first iteration, the centers of gravity of the formed clusters are equal to the optimum solution presented at the last row of Table 1. This is an usual behavior noticed in the computacional results produced by HSC algorithm, that provides an excellent stopping rule.

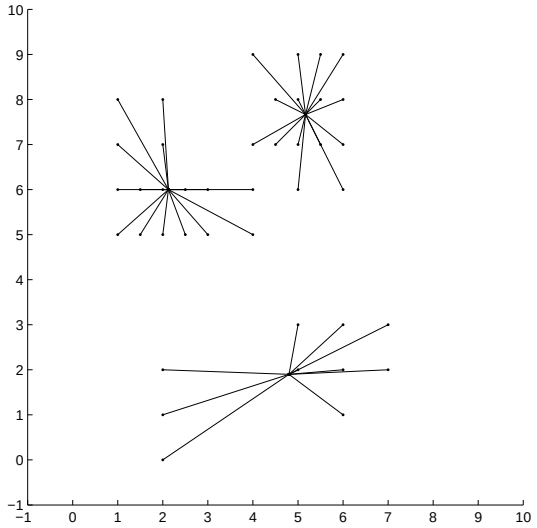

Figure 3: Solution for Demyanov Instance

In order to show some further performance of the proposed algorithm, results obtained by solving four standard test problems from the cluster analysis literature are shown below. The problems are:

1 - German Towns, which uses the two Cartesian coordinates of 59 towns, originally presented by Späth (1980);

2 - Ruspini example, which uses 75 artificial points in the Euclidean plane (Ruspini (1970));

3 - Iris Fisher example, which uses 4-dimensional data on 150 iris from the Gaspé peninsula, published by Anderson(1935) and used by Fisher(1936);

4 - TSPLIB-3038, which uses 3038 points in the plane from a traveling salesman problem of Reinelt (1991);

5 - Pla85900, the largest instance in the TSPLIB collection of challenge problems, which uses 85900 points.

The last two data sets are available in the site: http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/

Tables 2–3 contain the number of clusters (q), the best known value for the global optimum (f_{opt}) taken from du Merle et alli (2000) and Bagirov and Yearwood (2004), the value produced by the HSC algorithm (f_{HSC}) by using only one starting point, the error (E) for this solution and the CPU time given in seconds, where the error is calculated as

$$E = \frac{100\,(f_{HSC} - f_{opt})}{f_{opt}}. \tag{24}$$

q	f_{opt}	f_{HSC}	E	$Time$
2	0.121426 E6	0.121426 E6	0.00	0.05
3	0.77009 E5	0.77009 E5	0.00	0.08
4	0.49601 E5	0.49601 E5	0.00	0.09
5	0.38716 E5	0.38716 E5	0.00	0.25

Table 2: Results for the German Towns Instance

The results presented in Table 3 show that for German towns data set, Algorithm HSC reaches the best known results for $q = 2, 3, 4$ and 5 . The CPU time is small for all q.

q	f_{opt}	f_{HSC}	E	$Time$
2	89337.83	89337.83	0.00	0.07
3	51063.47	51063.48	0.00	0.10
4	12881.05	12881.05	0.00	0.10
5	10126.72	10126.72	0.00	0.20
6	8575.41	8575.41	0.00	0.21
7	7126.20	7126.20	0.00	0.22
8	6149.64	6149.64	0.00	0.28
9	5181.65	5181.65	0.00	0.38
10	4446.28	4446.28	0.00	0.39

Table 3: Results for the Ruspini Instance

From Table 3 it is possible to see that HSC Algorithm again, in small CPU times, gives the best known results for Ruspini data set for all values of q, except $q = 3$ where the error, $E = 0.00002\,\%$, is negligible.

In Table 4, the results for the Fisher Iris data set are presented. Ten different randomly chosen starting points were used. The third column gives

the best objective function value produced by HSC Algorithm. The next four columns give the number of occurrences of the best solution, the error of the best solution, the average error and the CPU mean time.

q	f_{opt}	$f_{HSC_{Best}}$	Occur.	E_{Best}	E_{Mean}	$Time_{Mean}$
2	15235.0	15234.8	10	0.00	0.00	0.10
3	7885.1	7885.1	10	0.00	0.00	0.16
4	5722.8	5722.8	10	0.00	0.00	0.29
5	4644.6	4644.6	7	0.00	2.21	0.44
6	3904.0	3904.0	10	0.00	0.00	0.52
7	3429.8	3430.0	7	0.01	2.39	0.97
8	2998.9	2998.9	5	0.00	3.52	1.04
9	2778.6	2778.6	9	0.00	1.20	1.34
10	2583.4	2583.4	9	0.00	0.38	1.70

Table 4: Results for the Fisher Iris Instance

From Table 4, it is possible to see that the HSC Algorithm with 10 restarts solves optimally, in small computational times, the Fisher Iris problems except for $q = 7$, where the relative error to the best known solution, $E = 0.006\,\%$, is nevertheless small.

Table 5 presents the results for the TSPLIB-3038 data set. The sixth column corresponds to the number of random starting points.

It is possible to observe in each row of Table 5 that the best solution produced by the HSC Algorithm is very close to the putative global minimum, the best known solution of the TSPLIB-3038 instance taken from [12, 16, 28]. Moreover, in this preliminary experiment, by using a relatively small number of initial starting points, seven new putative global minimum results ($q = 9$, $q = 20$, $q = 30$, $q = 40$, $q = 60$, $q = 80$ and $q = 100$) have been established, as recorded in the literature. In addition, it must be mentioned that for the biggest cases ($q = 60$, $q = 80$ and $q = 100$), the HSC Algorithm, by using 10 initial starting points, obtained respectively 4, 3 and 5 different solutions better than the old putative global minima. On the other hand, the low values shown in column E_{Mean} indicates that the HSC Algorithm computes deep local minima, meaning a good average performance.

q	f_{opt}	$f_{HSC_{Best}}$	Occur.	E_{Best}	$Start$	E_{Mean}	$Time_{Mean}$
2	0.31688E10	0.31705E10	10	0.05	10	0.05	0.60
3	0.21763E10	0.21776E10	7	0.06	10	1.08	1.08
4	0.14790E10	0.14793E10	10	0.02	10	0.02	1.81
5	0.11982E10	0.11986E10	9	0.03	10	0.06	3.09
6	0.96918E09	0.96936E09	10	0.02	10	0.02	9.56
7	0.83966E09	0.83967E09	7	0.001	10	4.66	10.65
8	0.73475E09	0.73491E09	1	0.02	10	1.35	16.24
9	0.64477E09	0.64471E09	2	**-0.01**	10	0.38	10.05
10	0.56025E09	0.56030E09	10	+0.01	10	0.01	16.16
20	0.26681E09	0.26675E09	1	**-0.02**	10	2.51	62.90
30	0.17557E09	0.17543E09	3	**-0.08**	10	0.36	169.17
40	0.12548E09	0.12541E09	1	**-0.06**	20	1.34	333.92
50	0.98400E08	0.98893E08	1	+0.50	40	1.46	662.33
60	0.82006E08	0.81115E08	1	**-1.09**	10	0.19	1049.97
80	0.61217E08	0.61025E08	1	**-0.31**	10	0.63	2038.31
100	0.48912E08	0.48470E08	1	**-0.90**	10	0.19	3499.76

Table 5: Results for the TSPLIB-3038 Instance

q	$f_{HSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.374908D+16	4	0.86	23.07
3	0.228057D+16	10	0.00	47.41
4	0.159308D+16	10	0.00	76.34
5	0.133969D+16	1	0.80	124.32
6	0.113663D+16	8	0.12	173.44
7	0.971101D+15	4	0.42	254.37
8	0.837741D+15	8	0.55	353.61
9	0.746602D+15	3	0.68	438.71
10	0.682942D+15	4	0.29	551.98

Table 6: Results for the Pla85900 Instance

Table 6 presents the results for the Pla85900 data set. Ten different randomly chosen starting points were used. The second and third columns give the best objective function value produced by the HSC Algorithm and the number of occurrences of this best solution. The last two columns correspond to the average error of the 10 solutions in relation to the best solution obtained and CPU mean time given in seconds.

The results presented in Table 6 shows an efficient performance of the HSC Algorithm, since the mean CPU time was consistently small despite the big size of the Pla85900. On the other hand, the high number of occurrences of the best solution shows a consistent performance of the HSC Algorithm. It was impossible to find any record of solutions of this instance. Indeed, the clustering literature seldom considers instances with such number of observations.

From the results presented in Tables 2–6 one sees that the first implementation of the HSC Algorithm reaches the best known solution in most of the cases, otherwise it calculates a solution which is close to the best. Moreover, in this first experiment, seven new best results in the cluster literature have been established for the TSPLIB-3038 instance. At last, a big size clustering problem, Pla85900, was suitably solved in small CPU times, given both the low dimension of nonlinear problem (16), defined on a $(nq)-$dimensional space, and the use of a minimization algorithm that takes advantage of its C^{∞} differentiable property.

Tables 7-17 present the computational results produced by a new XHSC Algorithm, including an experimental prunning procedure based on a partition of the set of observations in two non overlapping parts, for larger instances of Symmetric Traveling Salesman Problem (TSP) of the Reinelt (1991) collection: FL3795, FNL4461, RL5915, RL5934, Pla7397, RL11849, USA13509, BRD14051, D15112, BRD18512 and Pla33810. Ten different randomly chosen starting points were used. The second and third columns give the best objective function value produced by the HSC Algorithm and the number of occurrences of this best solution. The last two columns correspond to the average error of the 10 solutions in relation to the best solution obtained and CPU mean time given in seconds. It was impossible to perform a comparasion given the lack of records of solutions of these instances . Indeed, the clustering literature seldom considers instances with such number of observations.

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.105918E+10	4	0.33	0.24
3	0.721176E+09	2	5.12	0.29
4	0.542131E+09	2	2.23	0.33
5	0.368291E+09	1	6.12	0.40
6	0.265694E+09	2	4.07	0.45
7	0.197312E+09	2	3.29	0.51
8	0.157997E+09	1	2.97	0.62
9	0.121239E+09	1	8.04	0.71
10	0.106409E+09	1	4.08	0.91

Table 7: Results for the FL3795 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.467708E+10	5	0.18	0.26
3	0.294958E+10	3	0.26	0.34
4	0.231441E+10	1	0.32	0.40
5	0.181675E+10	1	0.67	0.47
6	0.150569E+10	1	0.52	0.61
7	0.122668E+10	1	1.47	0.67
8	0.108361E+10	1	1.91	0.68
9	0.952660E+09	1	3.02	0.92
10	0.853319E+09	1	1.09	1.04

Table 8: Results for the FNL4461 Instance

6 Conclusions

In this paper, a new method for the solution of the minimum sum-of-squares clustering problem has been proposed. By using the Hyperbolic Smoothing technique, the problem has been reformulated, in an approximation approach, as a completely differentiable constrained optimization problem. By using the Implicit Function Theorem, the problem was further reformulated as a low dimension unconstrained optimization problem, in the

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.100036E+12	6	0.19	0.17
3	0.642032E+11	4	0.40	0.24
4	0.485154E+11	2	0.69	0.47
5	0.379585E+11	3	0.59	0.51
6	0.318596E+11	1	0.54	0.69
7	0.267754E+11	2	0.70	0.67
8	0.234647E+11	1	0.51	0.84
9	0.208871E+11	1	2.55	1.17
10	0.187811E+11	1	0.87	1.27

Table 9: Results for the RL5915 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.920861E+11	1	0.66	0.29
3	0.681943E+11	1	1.55	0.37
4	0.488114E+11	2	0.81	0.49
5	0.393672E+11	1	0.97	0.71
6	0.317269E+11	2	0.65	0.98
7	0.279411E+11	1	1.12	0.96
8	0.244099E+11	1	1.44	1.12
9	0.215587E+11	1	1.34	1.37
10	0.191812E+11	1	1.89	2.07

Table 10: Results for the RL5934 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.178155E+15	4	0.21	0.30
3	0.111206E+15	3	0.97	0.45
4	0.629983E+14	2	0.31	0.61
5	0.506261E+14	1	1.75	0.69
6	0.396793E+14	1	2.14	0.75
7	0.352162E+14	1	1.32	0.81
8	0.308137E+14	1	2.31	1.32
9	0.272712E+14	1	4.07	0.93
10	0.243525E+14	1	3.09	1.21

Table 11: Results for the Pla7397 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.210287E+12	2	1.23	0.61
3	0.152061E+12	1	1.09	3.42
4	0.104991E+12	1	2.11	1.27
5	0.809571E+11	1	1.18	1.38
6	0.637439E+11	2	0.75	1.78
7	0.552341E+11	1	3.01	2.45
8	0.472998E+11	1	2.27	2.33
9	0.416351E+11	1	1.35	2.61
10	0.369192E+11	2	4.27	3.19

Table 12: Results for the RL11849 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.109770E+15	1	1.41	1.35
3	0.573853E+14	3	0.71	1.47
4	0.434554E+14	2	0.61	2.27
5	0.329531E+14	1	0.72	7.43
6	0.265986E+14	2	0.75	2.51
7	0.222732E+14	1	2.43	2.17
8	0.194874E+14	1	1.28	8.07
9	0.167993E+14	1	2.31	2.75
10	0.149840E+14	1	1.75	3.14

Table 13: Results for the USA13509 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.371152E+11	4	1.71	1.27
3	0.197682E+11	3	1.23	2.17
4	0.152334E+11	1	0.48	3.12
5	0.122288E+11	2	1.01	4.17
6	0.101929E+11	1	2.17	5.17
7	0.832857E+10	1	0.91	6.48
8	0.736814E+10	1	4.78	6.25
9	0.656451E+10	1	7.27	7.17
10	0.593939E+10	1	3.12	9.17

Table 14: Results for the BRD14051 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.368403E+12	4	0.35	0.79
3	0.253240E+12	2	0.72	1.35
4	0.173603E+12	2	0.81	1.47
5	0.132707E+12	1	1.23	1.61
6	0.111553E+12	1	1.11	2.04
7	0.994057E+11	1	2.11	2.27
8	0.816971E+11	1	2.17	2.74
9	0.713094E+11	1	3.21	3.12
10	0.644923E+11	1	2.42	3.49

Table 15: Results for the D15112 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.597589E+11	3	0.27	1.27
3	0.399281E+11	2	0.32	2.08
4	0.280691E+11	1	1.27	2.05
5	0.233421E+11	1	0.98	7.34
6	0.190493E+11	1	3.27	2.47
7	0.162834E+11	2	1.25	2.52
8	0.137893E+11	1	1.87	6.30
9	0.117489E+11	1	0.99	4.61
10	0.105912E+11	2	1.35	5.04

Table 16: Results for the BRD18512 Instance

q	$f_{XHSC_{Best}}$	Occur.	E_{Mean}	$Time_{Mean}$
2	0.946269E+15	2	1.27	2.23
3	0.605695E+15	3	1.72	4.08
4	0.404399E+15	4	1.20	5.32
5	0.335715E+15	2	4.27	6.27
6	0.280991E+15	1	1.73	7.28
7	0.238088E+15	1	1.68	8.75
8	0.204086E+15	2	2.13	8.70
9	0.179983E+15	1	1.11	12.12
10	0.164841E+15	1	1.38	11.27

Table 17: Results for the Pla33810 Instance

Euclidean space $\mathbb{R}^{nq}$. Then, the basic steps of an algorithm for solving the original clustering problem were presented.

Although only the particular MSSC formulation has been considered, it must be emphasized that this approach can be used for solving other clustering problems. For example, for the norm 1 formulation, only a trivial adaptation is necessary.

Moreover, it must be observed that the methodology can be applied to any $min - sum - min$ problem. Among them, we consider it to be particularly interesting to apply this approach to the general problem considered by Demyanov (2004).

The performance of the HSC Algorithm can be attributed to the complete differentiability of the approach. Within this context, each centroid gets to permanently see every observation point. Conversely, each observation point can permanently see every centroid and attract it. Figure 4 tries to depict this idea.

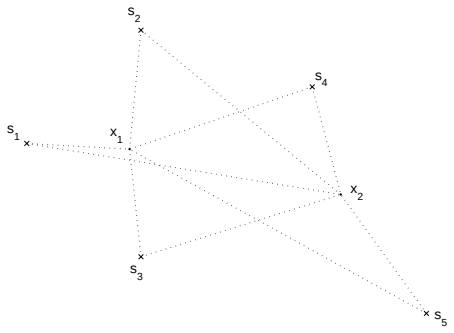

Figure 4: The C^{∞} differentiability property effect

There are several possibilities for the continuation of this work. One obvious alternative to be explored is to study simple algorithmic modifications, such as:

- Like other clustering algorithms, such as Eikelder and Erk (2004), Hansen and Mladenovic (2001) and du Merle et alli (2000), heuristics can be used to produce a good initial point through an auxiliary fast algorithm such as k-means or any of its numerous variations, Hansen el alli (2002).

- In a complementary way, it is possible to connect it with a heuristic or metaheuristic algorithm to do a local search, so as to pick up a better point around the initially found local minimum.

- It is also possible to calculate clusters step-by-step, gradually increasing the number of data clusters until reaching the specified q parameter value. This approach has been successfully adopted, for example, by Hansen et alli (2002), Bagirov and Yearwood (2004) and Bagirov(2008).

Finally, it must be remembered that the MSSC problem is a global optimization problem with a lot of local minima. Although the HSC Algorithm does not offer the guarantee of obtaining a global optimum point, in view of the results obtained, where a preliminary implementation of the proposed algorithm produced efficiently and reliably very deep local minima, perfectly adequate to the necessities and demands of real applications, we believe that the methodology introduced in this article can be used for solving large, practical optimal clustering problems.

Acknowledgments

The author would like to thank Prof. Fábio Dias Fagundez and Prof. Valmir Carneiro Barbosa of the Federal University of Rio de Janeiro for the helpful review of the work and constructive comments. The author also wishes to thank Mauricio G.C. Resende of ATT Labs - Research for supplying him with a set of suggestions which improved the paper.

References

ABADIE, J. and CARPENTIER, J. (1969) " Generalization of the Wolfe Reduced Gradient Method to the Case of Nonlinear Constraints", in "Optimization", R. Fletcher (editor), Academic Press, London, pages 37-47

ANDERBERG, M. R. (1973) "Cluster Analysis for Applications", Academic Press Inc., New York.

ANDERSON, E. (1935) "The Irises of Gaspe Peninsula", Bull. American Iris Society, 55 pp. 2-5

BAGIROV, A. M. and YEARWOOD, J. (2006) "A New Nonsmooth Optimization Algorithm for Minimum Sum-of-Squares Clustering Problems",

European Journal of Operational Research, 170 pp. 578-596

BAGIROV, A. M. (2008) "Modified Global k-means Algorithm for Minimum Sum-of-Squares Clustering Problems", Pattern Recognition, Vol 41 Issue 10 pp 3192-3199.

BRUCKER, P. (1978) "On the complexity of Clustering Problems", Lecture Notes in Economics and Mathematical Systems, 157, pp 45-54.

CHAVES, A.M.V. (1997) " Resolução do Problema Minimax Via Suavização", M.Sc. Thesis - COPPE - UFRJ, Rio de Janeiro.

DEMYANOV, A. (2005) "On the Solution of Min-Sum-Min Problems", Journal of Global Optimization, Kluwer Acad. Publ., 31, 3 pp. 437-453.

DUBES, R. C. and JAIN, A. K. (1976) "Cluster Techniques: The User's Dilemma", Pattern Recognition No. 8 pp. 247-260.

EIKELDER, H. M. M. and ERK, A. A. (2004) "Unification of Some Least Squares Clustering Methods", Journal of Mathematical Modeling and Algorithms No. 3 pp. 105-122.

FISHER, R.A. (1936) "The Use of Multiple Measurements in Taxonomic Problems", Ann. Eugenics, VII part II pp. 179-188. Reprinted in R. A. Fisher (1950) "Contributions to Mathematical Statistics", Wiley

HANSEN, P. and JAUMARD, B. (1997) "Cluster Analysis and Mathematical Programming", Mathematical Programming No. 79 pp. 191-215.

HANSEN, P. and MLADENOVIC, N. (2001) "J-Means: A New Heuristic for Minimum Sum-of-Squares Clustering", Pattern Recognition, Vol. 34 pp 405-413.

HANSEN, P., NGAI, E., CHEUNG, B. K., MLADENOVIC, N. (2005) "Analysis of Global k-Means, an Incrimental Heuristic for Minimum Sum-of-Squares Clustering", Journal of Classification 22, 287-310

HARTINGAN, J. A. (1975) "Clustering Algorithms", John Wiley and Sons, Inc., New York, NY.

JAIN, A. K. and DUBES, R. C. (1988) "Algorithms for Clustering Data", Prentice-Hall Inc., Upper Saddle River, NJ.

JAIN, A. K.; MURTY, M. N. and FLYNN, P. J. (1999) "Data Clustering: A Review", ACM Computing Surveys, Vol. 31, No. 3, Sept 1999.

JENSEN, R. E. (1969) "A Dynamic Programming Algorithm for Clustering Analysis", Operations Research No. 17 pp. 1043-1057.

KOONTZ, W. L. G.; NARENDRA, P. M. and FUKUNAGA (1975) "A Branch and Bound Clustering Algorithm", IEEE Transactions on Computers No. 24 pp. 908-915.

MANGASARIAN, O. L. (1997) "Mathematical Programming in Data Mining", Data Mining and Knowledge Discovery, Vol. 1, No. 2, pp 183-201.

MC QUEEN, J. (1967) "Some Methods for Classification and Analysis of Multivariate Observations", in Proceedings of the Fifth Berkeley Symposiums on Mathematical Statistics and Probability, pp. 281-297.

du MERLE, O. ; HANSEN, P.; JAUMARD, B.; MLADENOVIC, V. (2000) "An Interior Point Algorithm for Minimum Sum-of-Squares Clustering", SIAM Journal on Scientific Computing No. 21 pp. 1485-1505.

PACHECO, J. and VALENCIA, O. (2003) "Design of Hybrids for Minimum Sum-of-Squares Clustering Problem", Computational Statistics and Data Analysis No. 43 pp. 235-248.

REEVES, C. R. (Ed.) (1993) "Modern Heuristics Techniques for Combinatorial Problems", Blackwell, London.

REINELT, G. (1991) "TSP-LIB A Traveling Salesman Library", ORSA J. Comput. pp. 376-384

RUSPINI, E. (1970) "Numerical Methods for Fuzzy Clustering", Information Sci. pp. 319-350

SANTOS, A.B.A. (1997) "Problemas de Programação Não- Diferenciável: Uma Metodologia de Suavização", M.Sc. thesis - COPPE - UFRJ, Rio de Janeiro.

SOUSA, L.C.F. (2005) "Desempenho Computacional do Método de Agrupamento Via Suavização Hiperbólica", M.Sc. thesis - COPPE - UFRJ, Rio de Janeiro.

SPÄTH, H. (1980) "Cluster Analysis Algorithms for Data Reduction and Classification", Ellis Horwood, Upper Saddle River, NJ.

XAVIER, A.E. (1982) "Penalização Hiperbólica: Um Novo Método para Resolução de Problemas de Otimização", M.Sc. Thesis - COPPE - UFRJ, Rio de Janeiro.

XAVIER, A. E. and OLIVEIRA, A. A. F. (2004) "Optimal Covering of Plane Domains by Circles Via Hyperbolic Smoothing", to appear in Journal of Global Optimization, Kluwer.

XAVIER, A.E. (2008) "The Hyperbolic Smothing Clustering Method", submited to Pattern Recognition.