

# A Tabu Search Algorithm for Partitioning

Javier Trejos

University of Costa Rica

Eduardo Piza

University of Costa Rica

Alex Murillo

University of Costa Rica

---

This research was sponsored by the University of Costa Rica, project number 821-97-332, and the Research Program on Models and Data Analysis (PIMAD).

Author's Addresses: Javier Trejos, School of Mathematics, University of Costa Rica, 2060 San José, Costa Rica; [jtrejos@cariari.ucr.ac.cr](mailto:jtrejos@cariari.ucr.ac.cr). Eduardo Piza and Alex Murillo have the same address; electronic addresses: [epiza@cariari.ucr.ac.cr](mailto:epiza@cariari.ucr.ac.cr) and [murillof@cariari.ucr.ac.cr](mailto:murillof@cariari.ucr.ac.cr).

## Abstract

We present an original method for partitioning by automatic classification, using the optimization technique of tabu search. The method uses a classical tabu search scheme based on transfers for the minimization of the within variance; it introduces in the tabu list the indicator of the object transferred. This method is compared with two stochastic optimization-based methods proposed by the authors (one based on simulated annealing and the other on a genetic algorithm), and with the classical  $k$ -means and Ward methods. Results of the tabu search are significantly better than the classical and genetic methods, and slightly better than our simulated annealing method.

## 1 Introduction

Let  $\Omega$  be a set of  $n$  objects  $x_1, \dots, x_n$  with weights  $w_1, \dots, w_n$ , described by  $p$  numeric variables. We look for  $k$  well-separated, homogeneous classes  $C_1, C_2, \dots, C_k$  that form a partition  $P$  such that the within-classes variance

$$W(P) = \sum_{\ell=1}^k \sum_{i \in C_\ell} w_i \|x_i - g_\ell\|^2$$

is minimized, where  $g_\ell$  is the weighted centroid of  $C_\ell$  and  $\| \cdot \|$  is the euclidean norm on  $\mathbb{R}^p$ .

Many authors have proposed methods for partitioning objects ( $k$ -means by Forgy (1965) and McQueen (1967), “nuées dynamiques” by Diday (1971), transfers by Régnier (1965), Isodata by Ball and Hall (1967), etc.). However, all these methods are different variations of local search and so, the solution is only a local optimum of the criterion to be optimized. Moreover, they all depend on an initial partition (usually a random one or given by the user) and nearly all of them have the number  $k$  of classes as a parameter (or some given thresholds that control the number of classes).

The convergence to a local optimum of the classical methods has led us to consider modern global optimization methods, such as tabu search, simulated annealing and genetic algorithms. We have proposed original algorithms for the last two stochastic heuristics, described in Piza and Trejos (1996) and Trejos (1996).

## 2 Tabu Search

Tabu Search (TS) is a method of search of the optimum (here a minimum) in combinatorial optimization problems, and it was proposed by Glover (1989,1990). Let  $\mathcal{S}$  be the finite or countable set of states  $s$  of the problem, and  $f : \mathcal{S} \rightarrow \mathbb{R}$  the function to be minimized. It is assumed we can define for each  $s \in \mathcal{S}$  a neighborhood  $N(s)$  of  $s$ , that is, a subset of  $\mathcal{S}$  formed by the states that can be reached in one step (of the algorithm) from  $s$ . TS builds and handles a *tabu list*  $T$  of moves that are forbidden, that is, some states that cannot be reached, unless some special conditions—called the aspiration criterion—are satisfied. By definition, the algorithm moves from a current state  $s \in \mathcal{S}$  to another state  $s' \in N(s)$  defined by  $f(s') = \min\{f(x)/x \in N(s) - T\}$ .

The *aspiration criterion* establishes that a state  $s' \in N(s) \cap T$  can be reached from  $s$  if it has the minimum value that  $f$  has ever attained in the previous steps of the algorithm.

Usually, a TS algorithm begins with a random state  $s_0$  and it computes iteratively—by move into the neighborhood of the current state—a sequence of states  $s_1, s_2, s_3, \dots$  that may improve the criterion, but can also be worse than the previous step. It is this feature that gives TS the capacity to avoid local optima and reach the global optimum. TS ends at a maximum number of iterations fixed by the user, big enough to let the algorithm visit many “local valleys” of  $f$ .

It should be noted that TS makes a systematic use of the memory for handling the tabu list, and for this reason the size of  $T$  must be limited. Hence, a “forget strategy” is needed: it is necessary to define which state must be deleted from  $T$  when it is full and a new state should be put into  $T$ . Many forget strategies can be proposed: delete the worst state in  $T$ , delete the first state in  $T$ , etc.

TS has been used in many combinatorial optimization problems. See for example Amorim, Barthélemy and Ribeiro (1992), Cvijović and Klinowski(1995), Jung and Yum (1996) and Osman (1993).

### 2.1 Partitioning based on TS

We have applied the TS for the partitioning problem stated in the introduction. A state  $s$  is here a partition  $P$  into  $k$  classes of  $\Omega$ , and the cost function

to be minimized (the criterion) is the within-classes variance  $W$ .

A neighborhood  $N(P)$  of  $P$  will be the set of partitions  $\tilde{P}$  of  $\Omega$  defined by the transfer of one object to a new class in  $P$ . That is, a new partition  $P'$  is defined by the choice of one object  $x \in \Omega$ , which necessarily belongs to a class  $C_l$  of  $P$ , and the choice of an index  $l' \in \{1, \dots, k\}, l' \neq l$ , such that  $x$  will belong to  $C_{l'}$  in  $P'$ . It should be noted that the above definition of new partitions can yield empty classes; this is not disturbing since fewer number of classes does not decrease  $W$ . It is expected that partitions with fewer than  $k$  classes will be automatically discarded by the algorithm on the long run; in fact, the theoretical best partition of the problem has exactly  $k$  classes and no less (see Piza et al. (1994)).

When the partition  $P'$  is defined from  $P$ , then it is not necessary to compute the whole criterion  $W(P')$  to establish whether it is the best partition in the neighborhood of  $P$ . Indeed, when an object  $x$  moves from  $C_j$  in  $P$  to  $C_\ell$  in  $P'$ , it is only necessary to compute the difference

$$\Delta W = W(P') - W(P) = \frac{|C_j|}{n(|C_j| - 1)} \|g(C_j) - x\|^2 - \frac{|C_\ell|}{n(|C_\ell| + 1)} \|g(C_\ell) - x\|^2,$$

where  $|C_j|$  is the cardinality of class  $C_j$  and we suppose that all the objects have the same weight  $w_i = 1/n$  (a similar formula can be found in the case of general weights). The centroids of the classes change in the following way:

$$g(C_j - \{x\}) = \frac{1}{|C_j| - 1} (|C_j|g(C_j) - x), g(C_\ell \cup \{x\}) = \frac{1}{|C_\ell| + 1} (|C_\ell|g(C_\ell) + x).$$

In our algorithm, the tabu list  $T$  contains all partitions  $P'$  that contain, as a subset of one of their classes, any class  $C_j \subset \Omega$  from which an element  $x \in C_j$  has been moved in a former step (and included into another class). For this, the class indicator of  $C_j$  is enough for encoding these partitions in the tabu list. For example, if  $n = 7$  and  $k = 3$ ,  $P = (C_1, C_2, C_3)$  with  $C_1 = \{x_2, x_3, x_7\}, C_2 = \{x_1, x_4, x_6\}, C_3 = \{x_5\}$  and object  $x_2$  moves from  $C_1$  to  $C_2$ , then the class indicator of  $C_1$ , that is 0110001, enters in  $T$ . This indicator forbids that the objects  $x_2, x_3$  and  $x_7$  will be together again, unless the aspiration criterion is reached or this class indicator leaves  $T$ . Thus, the class indicators describe the elements of  $T$ . That is to say, each element of  $T$  is not only one partition, but the set of partitions such that one class is specified (by the class indicator). We use the “first in–first out” forget

strategy for  $T$  (that is,  $T$  is a queue). This encoding of the tabu list yielded to better results than some others we also tried, such as  $W$  values, object index and class index. We define the length of the tabu list as the number of class indicators that describe the partitions in the tabu list, and denote it  $|T|$ .

As in the TS scheme, a new partition  $P'$  is accepted if it has the minimum value of  $W$  in  $N(P) - T$ . In what follows, our algorithm will be called TabuClus.

## 2.2 Example: French scholar notes

Let us consider a simple data set, known as the French scholar notes used by Schektman (1978) (see Table 1).

$\Omega$	Math	Scien	French	Latin	Sport
Jean	6	6	5	5.5	8
Alain	8	8	8	8	9
Anne	6	7	11	9.5	11
Monique	14.5	14.5	15.5	15	8
Didier	14	14	12	12.5	10
André	11	10	5.5	7	13
Pierre	5.5	7	14	11.5	10
Brigitte	13	12.5	8.5	9.5	12
Evelyne	9	9.5	12.5	12	18

Table 1: The data set of the French scholar notes.

TabuClus found always the optimal partitions in  $k = 2, 3$  and 4 classes, shown in Table 2, using  $|T| = 5$ . Since the data table is quiet little, these optimal partitions can be also found by exhaustive enumeration of all partitions, whose quantity are given by Stirling numbers  $S(9, 2) = 255$ ,  $S(9, 3) = 3025$  and  $S(9, 4) = 7770$ . Table 2 contains the centroids of the classes, given by the variable means.

$W$	Classes	Mat	Sci	Fre	Lat	Gym
271.8	$C_1$ : J,Al,Ad,An,P	7.3	7.6	8.7	8.3	10.2
	$C_2$ : E,B,D,M	12.6	12.6	12.1	12.2	12.0
235.0	$C_1$ : J,Al,An	8.3	8.0	6.2	6.8	10.0
	$C_2$ : An,P,E	6.8	7.8	12.5	11.0	13.0
	$C_3$ : B,D,M	13.8	13.7	12.0	12.3	10.0
202.6	$C_1$ : J,Al	7.0	7.0	6.5	6.7	8.5
	$C_2$ : An,P,E	6.8	7.8	12.5	11.0	13.0
	$C_3$ : An,B	12.0	11.2	7.0	8.2	12.5
	$C_4$ : D,M	14.2	14.2	13.7	13.7	9.0

Table 2: Partitions and class centroids found with TabuClus for the French scholar notes

### 3 Two other partitioning methods based on stochastic optimization

We have also applied two other global optimization techniques for the partitioning problem: simulated annealing and a genetic algorithm.

#### 3.1 Simulated Annealing

Simulated annealing (SA) is perhaps the best known of the three techniques of combinatorial optimization we have used. It was proposed by Kirkpatrick et al. (1983) and Černý (1985), and it is based on an analogy with the annealing method of Statistical Physics (see Aarts and Korst (1990) or Laarhoven and Aarts (1988)), which deals with a control parameter  $c$  that plays the role of the temperature and the Metropolis rule that simulates the Boltzmann distribution. We will not describe this general method, the interested reader may consult one of the above references. However, we mention an important feature: the SA algorithm converges, under some conditions, asymptotically to a global optimum. These conditions of convergence deal (1) with the size and the access probability of the neighborhoods of a state, (2) a condition of reversibility and (3) a condition of connectivity. In any SA implementation there must be defined a cooling schedule: initial and final values of the control parameter  $c$ , the decreasing law of  $c$  and the length of Markov chains

associated with each value of  $c$ .

The partitioning algorithm based on SA which we have studied (see Piza and Trejos (1996) for the details of the method), defines a new partition  $P'$  from a partition  $P = (C_1, \dots, C_k)$  by:

1. choosing at random an object  $x \in \Omega$ ,
2. choosing at random a class index  $\ell \in \{1, \dots, k\}$ ,
3. putting  $x$  into the class  $C_\ell$ .

In case that the object  $x$  was not in class  $C_\ell$  in the previous step, this is equivalent to make a random *transfer*, as used by Régnier (1965). We apply then the usual SA algorithm; this method is similar to the clustering approach proposed by Klein and Dubes (1989).

It should be noted that, with our method, the asymptotic convergence conditions are satisfied: for the reversibility, it is possible to go back to a previous configuration by the choice of the same object  $x$  and its previous class index; for the connectivity, by a finite number of transfers it is possible to go from any partition  $P_i$  to any partition  $P_{i'}$ ; the size of a neighborhood of a partition  $P$  is  $n(k-1)$ ; and the probability of generating a partition  $P'$  from a partition  $P$  is  $\frac{1}{n(k-1)}$ . Simplification of  $\Delta W$  is again important for improving the time in the application of the method.

## 3.2 Genetic Algorithm

Genetic algorithms (GA) were proposed by Holland (1976) (see also Goldberg (1989)). They are based on an analogy between the evolution of genes of species and the adaptation of the solution of a combinatorial optimization problem to its “environment” measured by a good value of the cost function, also called fitness function. The reader interested in this topic may consult the above references, here we will only present an outline of the GA we have proposed for partitioning; some further details can be found in Piza et al. (1997) or Piza and Trejos (1996).

The genetic partitioning algorithm we proposed is based on a “chromosomal” representation of a partition  $P = (C_1, \dots, C_k)$ , with  $n$  alleles  $1, \dots, n$  and an alphabet of  $k$  letters; for example the “chromosome” (22311132) indicates that  $C_1 = \{x_4, x_5, x_6\}$ ,  $C_2 = \{x_1, x_2, x_8\}$  and  $C_3 = \{x_3, x_7\}$ . It is

well known that the total variance is the sum of  $W$  and the between-classes variance  $B$ , so minimizing  $W$  is equivalent to maximizing  $B$ . Since GA are stated for maximizing functions, the cost function to be maximized in our GA is  $B$ , which plays the role of the fitness function. The genetic operators are (1) the classical selection proportional to  $B$  (roulette wheel), (2) crossover (in this case, equivalent to the transfer of several objects at the same time) and (3) mutation (equivalent to the transfer of a single object). However, since this raw application of the classical GA did not yield good results, we studied some new genetic operators:

- A *forced crossover*: we choose at random two partitions (parents)  $P_1, P_2$ , and a class index  $\ell$  from  $P_1$ , then we copy this class of  $P_1$  into  $P_2$ , constructing a new partition  $P'$  (son). That is, using the “chromosomic” representation of partitions, if  $P_1 = (\alpha_1 \dots \alpha_n), P_2 = (\beta_1 \dots \beta_n)$  and  $\alpha_i = \ell$ , then  $P' = (\gamma_1 \dots \gamma_n)$  is defined by  $\gamma_i := \beta_i$  if  $\alpha_i \neq \ell$  and  $\gamma_i := \alpha_i$  if  $\alpha_i = \ell$ . This operator has the advantage over classical crossover that some useful knowledge about partitioning (the fact that some elements are together in  $P_1$ ) is used.
- A *weak mutation*, equivalent to an exchange between two objects chosen at random.

This GA method has been improved with the application of the Forgy’s  $k$ -means method in each partition of the genetic population after some iterations of the GA.

## 4 Comparative results

In Table 3 we show some comparative results of our partitioning methods using TabuClus, SA and GA, as well as the classical ones of Forgy’s  $k$ -means and Ward’s hierarchical method (the hierarchical tree is cut at a level such that a partition is created with the indicated number of classes). The underlying data tables were the French scholar notes (9 individuals described by 5 variables), the Amiard’s fishes (23 fishes described by 16 variables) and the Thomas’ sociomatrix (a square matrix  $24 \times 24$ ) used in Cailliez and Pagès (1976), and the Fisher’s well-known Iris data (150 species of Iris described by 4 variables). In the table we present the values of the criterion  $W$  and the



quality of the solutions obtained with each method, that is, the percentage of the best solution shown, obtained in repetitions of the method.

The parameters of TabuClus, for each data set, were: for the French scholar notes, the length of the tabu list  $|T| = 5$  and the number of iterations  $Iter = 10$ , for the Amiard's fishes  $|T| = 10$  and  $Iter = 20$ , for the Thomas' sociomatrix  $|T| = 10$  and  $Iter = 20$  and for Fisher's Iris data  $|T|$  is 10, 15, 15 and 20, and  $Iter$  is 75, 100, 100 and 120 respectively for 2, 3, 4 and 5 classes.

It can be seen in Table 3 that the better results are obtained with the TS and the SA, and these are significantly better than those obtained with the classical methods. The chance for finding a better solution with our methods is much greater than that with classical methods. Although not considered here, search time with TS is usually less than that with SA.

French Scholar Notes ( $9 \times 5$ )									
	TS		SA		GA		kM		Ward
	# = 1000		# = 150		# = 100		# = 10000		W
	W	%	W	%	W	%	W	%	W
2 classes	28.2	100%	28.2	100%	28.2	100%	28.2	12%	28.8
3 classes	16.8	100%	16.8	100%	16.8	95%	16.8	12%	17.3
4 classes	10.5	100%	10.5	100%	10.5	97%	10.5	5%	10.5
5 classes	4.9	100%	4.9	100%	4.9	100%	4.9	8%	4.9
Amiard's Fishes ( $23 \times 16$ )									
	TS		SA		GA		kM		Ward
	# = 200		# = 150		# = 100		# = 10000		W
	W	%	W	%	W	%	W	%	W
2 classes	69849	96%	69368	100%	69368	52%	69849	49%	–
3 classes	32213	100%	32213	100%	32213	87%	32213	8%	33149
4 classes	18281	100%	18281	100%	22456	90%	18281	9%	19589
5 classes	14497	97%	14497	100%	20474	38%	14497	1%	14497
Thomas' Sociomatrix ( $24 \times 24$ )									
	TS		SA		GA		kM		Ward
	# = 200		# = 150		# = 100		# = 10000		W
	W	%	W	%	W	%	W	%	W
2 classes	333.7	99%	333.7	100%	33.7	69%	333.7	5%	–
3 classes	271.8	100%	271.8	100%	272.9	85%	271.8	2%	279.3
4 classes	235.0	100%	235.0	100%	250.8	24%	235.0	0.15%	239.4
5 classes	202.6	98%	202.4	100%	223.8	4%	202.6	0.02%	204.7
Fisher's Iris ( $150 \times 4$ )									
	TS		SA		GA		kM		Ward
	# = 25		# = 150		# = 50		# = 10000		W
	W	%	W	%	W	%	W	%	W
2 classes	0.999	100%	0.999	100%	0.999	100%	0.999	100%	–
3 classes	0.521	76%	0.521	100%	0.521	100%	0.521	4%	–
4 classes	0.378	60%	0.378	55%	0.378	82%	0.378	1%	–
5 classes	0.312	32%	0.329	100%	0.312	6%	0.312	0.24%	–

Table 3: Results (within-variance  $W$ ) of partitioning using tabu search (TS), simulated annealing (SA), genetic algorithm (GA), Forgy  $k$ -means (kM) and Ward's hierarchical method; # indicates the number of times that the method was applied and % is the percentage of repetitions of the method that found the best solution reported.

## 5 Concluding remarks

The tabu search method proposed for partitioning is an easy implementation of the tabu search method. Its performance is better than that of the classical methods ( $k$ -means and Ward) and its results are in nearly all cases comparable with the simulated annealing-based method, even though in some cases the SA method found the optimum in all cases and the tabu search did not. Even though we do not report the time of execution, we have observed that TabuClus is faster than the SA and GA methods.

However, the choice of the parameters is a delicate operation and should be studied further. Indeed, the best solution may not be found if the tabu list is too short or the number of iterations is not large enough. We recommend to use short tabu lists, with  $|T|$  between 5 and 20, and a maximum number of iterations between 200 and 1000, depending on the number of objects to be clustered. The simulated annealing and the genetic algorithm methods also share this problem of handling the parameters: the decreasing law of the control parameter and the length of the Markov chains, in the SA case, and the probabilities for applying the genetic operators, the size of the population and the stop criterion, in the GA case. If the data set has too many objects, the tabu search method is slow, but its quality remains good as reported here.

In the near future we will report on our present investigation on some hybrid methods. Indeed, we are testing the use of SA in this TS method, which we expect will help as a stopping criterion when the control parameter tends to zero. Also, a genetic operator based on the Metropolis rule of SA, may define a new kind of mutation and could help as a stopping criterion for the GA. Genetic operators that handle a tabu memory can also be proposed.

Additionally, we are implementing the use of non-euclidean distances over numerical data, as well as the adaptation of the methods to the case of categorical data.

## References

AARTS, E. and KORST, J. (1990) *Simulated Annealing and Boltzmann Machines. A Stochastic Approach to Combinatorial Optimization and Neural Computing*. John Wiley & Sons, Chichester.

- AMORIM, S. G. de; BARTHÉLEMY, J.-P. and RIBEIRO, C. C. (1992) "Clustering and clique partitioning: simulated annealing and tabu search approaches", *Journal of Classification* 9: 17–41.
- BALL, G.H. and HALL, D.J. (1965) "ISODATA, a novel method of data analysis and data classification", Stanford Res. Inst. Menlo Park, California.
- CAILLIEZ, F. and PAGÈS, J.-P. (1976) *Introduction à l'Analyse des Données*. SMASH, Paris.
- ČERNÝ (1985) *Journal of Optimization, Theory and Applications*, 45: 41.
- CVIJOVIĆ, D. and KLINOWSKI, J. (1995) "Taboo search: an approach to the multiple minima problem", *Science* 267: 664–666.
- DIDAY, E. (1971) "Une nouvelle méthode en classification automatique et reconnaissance des formes", *Revue de Statistique Appliquée*, XIX (2).
- EVERITT, B.S. (1993) *Cluster Analysis*. 3rd. edition. Edward Arnold, London.
- FISHER, R. A. (1936) "The use of multiple measurements in taxonomic problems", *Ann. Eug.* 7: 179–188.
- FORGY (1965) "Cluster analysis of multivariate data: efficiency versus interpretability of classification", *Biometrics* 21: 768–769.
- FOX, B.L. (1993) "Integrating and accelerating tabu search, simulated annealing, and genetic algorithms", *Annals of Operations Research* 41: 47–67.
- GLOVER, F. (1989) "Tabu search - Part I", *ORSA J. Comput.* 1: 190–206.
- GLOVER, F. (1990) "Tabu search - Part II", *ORSA J. Comput.* 2: 4–32.
- GLOVER, F.; TAILLARD, E. and WERRA, D. de (1993) "A user's guide to tabu search", *Annals of Operations Research* 41: 1–28.
- GOLDBERG, D.E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading-Mass.
- HOLLAND, J.H. (1975) *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor.
- JUNG, J. S. and YUM, B. J. (1996) "Construction of exact D-optimal designs by tabu search", *Computational Statistics & Data Analysis* 21: 181–191.
- KIRKPATRICK, S.; GELATT, D. and VECCHI, M.P. (1983) "Optimization by simulated annealing", *Science* 220: 671–680.
- KLEIN, R. W. and DUBES R. C. (1989) "Experiments in projection and

- clustering by simulated annealing”, *Pattern Recognition 22*: 213–220.
- LAARHOVEN, P.J.M. van and AARTS, E.M. (1988) *Simulated Annealing: Theory and Application*. Kluwer Academic Publ., Dordrecht.
- MacQUEEN, J. (1967) “Some methods for classification and analysis of multivariate observations”, *Proc. 5th Berkeley Symp.*, 1, 281–297.
- MURILLO, A. and TREJOS, J. (1996) “Classification tabou basée en transferts”, *IV Journées de la Société Francophone de Classification*, S. Joly & G. Le Calvé (Eds.), Vannes: 26.1–26.4
- OSMAN, I. H. (1993) “Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem” *Annals of Operations Research 41*: 421–451.
- PIZA, E.; TREJOS, J. and MURILLO, A. (1994) “Clasificación automática: particiones usando algoritmos genéticos y de sobrecalentamiento simulado”. Research Report #114-94-228, Univ. Costa Rica.
- PIZA, E. and TREJOS, J. (1996) “Partitionnement par recuit simulé”, *IV Journées de la Société Francophone de Classification*, S. Joly & G. Le Calvé (Eds.), Vannes: 27.1–27.4
- RÉGNIER, S. (1965) “Sur quelques aspects mathématiques des problèmes de classification automatique”, *I.C.C. Bulletin*, 4: 175–191.
- SCHEKTMAN, Y. (1978) “Estadística descriptiva (análisis lineal de datos multidimensionales)”, I part. *Proceedings of the 1st. Symposium on Mathematical Methods Applied to the Sciences*, J. Badia, Y. Schektman & J. Poltronieri (Eds.), Universidad de Costa Rica, San Pedro, pp. 9–67.
- TREJOS, J. (1996) “Un algorithme génétique de partitionnement”, *IV Journées de la Société Francophone de Classification*, S. Joly & G. Le Calvé (Eds.), Vannes: 31.1–31.4
- TREJOS, J.; MURILLO, A. and PIZA, E. (1998) “Global stochastic optimization for partitioning”, in: *Advances in Data Science and Classification*, A. Rizzi, M. Vichi & H.-H. Bock (Eds.), Springer, Heidelberg: 185–190.