INFORMATICA, 1990, Vol.1, No.1, 020–039

SIMULATED ANNEALING FOR PARAMETER GROUPING

Gintautas DZEMYDA, Elvyra SENKIENĖ,
Jūratė VALEVIČIENĖ

Institute of Mathematics and Cybernetics,
Lithuanian Academy of Sciences,
232600 Vilnius, Akademijos St.4, Lithuania

Abstract. In this paper we deal with the problem of extremal parameter grouping.The problem formulation, the algorithms of parameter grouping and the fields of implementation are presented. The deterministic algorithms of extremal parameter grouping often find the local maximum of the functional, characterizing the quality of a partition.The problem has been formulated as a problem of combinatorial optimization and attempted to be solved using the simulated annealing strategy. The algorithms, realizing such a strategy and devoted to the solving of the problem concerned, are proposed and investigated.

Key words: simulated annealing, parameter grouping, combinatorial optimization.

1. The problem. Structural methods for empirical data processing are used widely in systems analysis. The method of extremal parameter grouping (Braverman, 1970; Braverman and Muchnik, 1983; Dzemyda, 1987a, 1987b, 1988) belongs to this class of methods. It is devoted to the partition of the parameters $x_1, \ldots, x_n$ into a fixed number p of the unintersecting and homogeneous in a sense groups $A_1, \ldots, A_p$ by the

correlation matrix $R = \{r_{x_i x_j}, i,j = \overline{1,n}\}$ characterizing the connections among the parameters ($r_{x_i x_j}$ is the correlation coefficient of parameters x_i and x_j). The covariance matrix may be used instead of the matrix R. However, the parameters with a greater dispersion will have more significance in the analysis. There is no *a priori* information regarding the number and size of groups.

There is a variety of different algorithms for parameter grouping of such a type. The modification of algorithms proposed by Harman (1976) is included in SAS (1982). The algorithms of such a type are also presented, for example, by Anderberg (1973), Hartigan (1976), Braverman and Muchnik (1983). Dzemyda (1990) has proposed a new approach to solve the problem concerned. The partitioning quality of constructed algorithms is like that of the best algorithms of extremal parameter grouping proposed by Dzemyda (1987a), but the new algorithms require significantly less computational expenditures. All these algorithms will not be considered in our paper.

The problem of extremal parameter grouping includes maximization of the functional

$$I_1 = \sum_{L=1}^{p} \sum_{x_i \in A_L} r^2_{x_i F_L} ,$$

where F_L is the factor with a unit dispersion corresponding to the group A_L; $r_{x_i F_L}$ is the correlation coefficient of the parameter x_i and the factor F_L. The factors $F_L, L = \overline{1,p}$, are selected so as to maximize the sums

$$\sum_{x_i \in A_L} r^2_{x_i F_L}, \; L = \overline{1,p}.$$

Dzemyda (1987a) proved, that

1. $F_L = \sum\limits_{x_i \in A_L} \alpha_i^L x_i / \sqrt{\lambda}_L, \; r_{x_s F_L} = \sum\limits_{x_i \in A_L} \alpha_i^L r_{x_i x_s} / \sqrt{\lambda}_L.$

where λ_L is the greatest eigenvalue of the matrix $R_L = \{r_{x_i x_j}, x_i, x_j \in A_L\}$, α_i^L are the components of the normalized eigenvector of the matrix R_L corresponding to λ_L .

2. $r_{x_s F_L} = \sqrt{\lambda_L}\alpha_s^L$ as $x_s \in A_L$.

3. $I_1 = \sum_{L=1}^{p} \lambda_L$.

4. Let some partition of the parameters into the groups $A_1, \ldots, A_p$ be given and let the factors $F_1, \ldots, F_p$ be fixed. Let us analyze some parameter $x_s \in A_k$. By using the values of $r_{x_s F_L}, L = \overline{1,p}$, it is impossible to determine the group, where the value of the functional I_1 increases at most after transferring the parameter x_s.

2. The deterministic algorithms of I_1 maximization. The deterministic algorithms, proposed by Braverman and Muchnik (1983), Dzemyda (1987a, 1987b, 1988), are grounded on the analysis of parameters in consecutive order and on the search of a group to transfer the individual parameter with a purpose to increase I_1 value. There is a variety of strategies for such a search. The algorithms stop when the transferrings of any parameter by the chosen strategy do not increase I_1 value. The factors can be recalculated after every transferring of the parameter (Dzemyda, 1987a, 1987b, 1988) or after finding new groups for all parameters (Braverman and Muchnik, 1983).

Note 1. Let some partition of the parameters into the groups $A_1, \ldots, A_p$ be given and let the factors $F_1, \ldots, F_p$ be fixed. Let us analyze some parameter $x_s \in A_k$. The value of the functional I_1 will increase after transferring the parameter x_s from the group A_k to the group A_L, if we succeed in finding such a factor $F_L (L \neq k)$ that

$$r^2_{x_s F_k} < r^2_{x_s F_L}$$

(Braverman and Muchnik, 1983).

Definition 1. By the local maximum of the functional I_1 we shall call its value, corresponding to such a partition, where the correlation coefficient of any parameter with the factor, corresponding to the group including this parameter, is greater or equal to that of this parameter with another factors.

A number of partitioning algorithms based on Note 1 for maximization of the functional I_1 is proposed (Braverman and Muchnik, 1983; Dzemyda, 1987a, 1987b, 1988). They are grounded on calculating the correlation coefficients of parameters with the factors and transferring the parameters from one group to another one depending on the values of these coefficients. Let us denote the algorithms of such a type by A1.

Denote:

1) λ_k^{-s} is the maximal eigenvalue of the matrix $R_k^{-s} = \{r_{x_i x_j}, x_i, x_j \in A_k^{-s}\}$, where the group A_k^{-s} is obtained from the group A_k by eliminating the parameter x_s (note that $\lambda_k^{-s} = 0$ as $A_k^{-s} = \varnothing$):

2) λ_L^{+s} is the maximal eigenvalue of the matrix $R_L^{+s} = \{r_{x_i x_j}, x_i, x_j \in A_L^{+s}\}$, where the group A_L^{+s} is obtained from the group A_L by adding the parameter x_s.

Note 2. $\lambda_k - \lambda_k^{-s} \geqslant 0$, $\lambda_L^{+s} - \lambda_L \geqslant 0$, $k = \overline{1,p}$, $L = \overline{1,p}$, $s = \overline{1,n}$.

Note 3. Let some partition of the parameters into the groups $A_1, \ldots, A_p$ be given. Let us analyze some parameter $x_s \in A_k$. The value of the functional I_1 will increase after transferring the parameter x_s from the group A_k to the group A_L, if we succeed to find such a value $\lambda_L^{+s} (L \neq k)$ that

$$\lambda_k - \lambda_k^{-s} < \lambda_L^{+s} - \lambda_L.$$

Note 4. Let some partition of the parameters into groups $A_1, \ldots, A_p$ be given. The value of the functional I_1 corre-

sponding to the given partition is the local maximum in the case when for any parameter x_s (let $x_s \in A_k$)

$$\lambda_k - \lambda_k^{-s} \geqslant \lambda_L^{+s} - \lambda_L, \; L = \overline{1,p},$$

is valid.

Notes 2-4 make the basis for the other type of maximization algorithms of the functional I_1. These algorithms do not calculate the correlation coefficients of individual parameters and factors. Their operation is grounded on the calculation and comparison of the maximal eigenvalues of submatrices of R.In this way it is possible to find the group for any parameter x_s(let $x_s \in A_k$) , where the value of the functional I_1 increases at most when transferring this parameter from the group A_k to this group. Such an algorithm is proposed by Dzemyda (1987a). Let us denote it by A2.

The algorithm A2 requires considerably more expenditures of computation time in comparison with the algorithms of A1 type. It results from the fact that $A2$ calculates the maximal eigenvalues of symmetrical matrices more often. However, A2 will be more effective than A1 in the mean sense (greater values of the functional I_1 will be obtained), because in A1 the group for transferring of the parameter x_s is determined by the values of the correlation coefficients of this parameter and the factors $F_1, \ldots, F_p$. Dzemyda (1987a) proved that in this manner it is impossible to determine the group, where the value of the functional I_1 increases at most after transferring this parameter. Such a group is determined in A2.

Dzemyda (1987a) proposed some algorithms with respect to the results of Notes 1-4 taken together. They give the advantage to save the computation time in comparison with A2, but yield A2 in efficiency.

The initial partition of the parameters into p groups is necessary for the operation of maximization algorithms of the

functional I_1. Dzemyda (1987a) proposed and investigated two algorithms of the initial partition. The number p of groups can be fixed preliminary or selected automatically.

The computer programs, realizing the deterministic algorithms of extremal parameter grouping, are written in FORTRAN and can be used on any EC type computer. The properties of the algorithms were investigated by Dzemyda (1987a) on test and real data.

Dzemyda and Valevičienė (1988), Dzemyda (1990) have found that the algorithms of extremal parameter grouping may be used in the clustering of points (objects). In this case, elements of the matrix R were chosen as the values of some potential function , dependent on a distance between a pair of points (objects) to be clustered. The results of comparison of the clustering algorithms, based on extremal parameter grouping, with the well known algorithms of the cluster analysis were obtained by Dzemyda and Valevičienė (1988). Valevičienė with coworkers (1988), Dzemyda, Tiešis and Valevičienė (1989) included these clustering algorithms into the software, combining the processes of minimization and objective function calculation. This software is devoted to the solving of optimization problems, where the calculation of objective function values on some nodes of a rectangular lattice requires much less computation time than individual calculations of these values. Clustering is used in global search.

3. Simulated annealing approach. The algorithms from chapter 2 often find the local maximum of I_1 . A new way to solve the extremal parameter grouping problem is to formulate it as the combinatorial optimization one and to search for the global maximum of I_1 .

The problem can be formulated as follows:

$$\max f(X), \quad X = (X^1, \ldots, X^n) \tag{1}$$

subject to

$$X^i \in \{1, \ldots, p\}, \quad i = \overline{1,n}, \tag{2}$$

$$\exists i: \ X^i = 1, \ldots, \quad \exists i: \ X^i = p. \tag{3}$$

The function $f(\cdot)$ and the functional I_1 are related in such a manner:

$$f(X^1, \ldots, X^n) = I_1,$$

where

$$x_i \in A_L \quad \text{as} \quad X^i = L.$$

The function $f(\cdot)$ is bounded, i.e., $0 < f(X) \leqslant n \max\limits_{i=\overline{1,n}} r_{ii}$ for any X, satisfying (2) and (3) ($0 < f(X) \leqslant n$ when the correlation matrix is analyzed).

The algorithms from chapter 2 practically realize a single coordinate search: consequently, the values of individual variables from $\{X^1, \ldots, X^n\}$ run from 1 to p (the values of other variables are fixed) and there is searched for increasing of the objective function value.

The problem (1)-(3) can be solved by using a simulated annealing strategy. There is a variety of algorithms of such a type (Gidas, 1985; Mitra, Romeo and Sangiovanni-Vincentelli, 1986; Lundy and Mees, 1986; Bhanot, 1988; Chiang and Chow, 1988) which can be used for global optimization.

The performance of optimization algorithms grounded on the simulated annealing can be generalized as follows:

1. Selection of a new point Y_m from the neighbours of the current point X_{m-1}, the calculation of $f(Y_m)$.

2. Choosing of a new current point X_m such as X_{m-1} or Y_m with some probability depending on $f(X_{m-1})$, $f(Y_m)$ and T_m, where T_m is the value of some annealing parameter.

3. $m = m + 1$, transition to step 1.

The algorithm presented below has some similarities with that proposed by Mitra, Romeo and Sangiovanni-Vincentelli (1986).

The search for the global maximum of $f(X)$ ($X = (X^1,\dots,X^n) \in D = [A,B]^n \subset R^n$, $A = (A^1,\dots,A^n)$, $B = (B^1,\dots,B^n)$, $A^i \leqslant X^i \leqslant B^i$, $i = \overline{1,n}$; X^i, A^i and B^i are integer numbers) can be performed in such a manner: the m-th step of the algorithm is as follows:

$$X_m^i = X_{m-1}^i + \xi^i, \quad m = 1,2,\dots, \; i = \overline{1,n}, \tag{4}$$

where ξ^i, $i = \overline{1,n}$, are integer numbers taking the values with some probabilities:

a) $P\{\xi^k = -1\} = P\{\xi^k = 1\} = \frac{1}{2}$,

$$\xi^i = 0, \; i = 1,2,\dots,k-1,k+1,\dots,n, \tag{5a}$$

b) ξ^ktakes the values from

$$D^k = \{A^k - X_{m-1}^k, \; A^k + 1 - X_{m-1}^k,\dots,B^k - X_{m-1}^k\}\backslash\{0\}$$

with the same probabilities:

$$p_k = 1/(B^k - A^k), \tag{5b}$$

$$\xi^i = 0, \quad i = 1,2,\dots,k-1,k+1,\dots,n.$$

The point X_{m-1} denotes the initial point for the m-th step and the current point of the algorithm after m-1 steps. The relation of m and k must be determined specially (the example of such a relation see in chapter 3.1.) $A^k = 1$, $B^k = p$, $k = \overline{1,n}$, in the case of a problem (1)-(3).

The probability of transition to the point X_m is determined by the formula:

$$P\{X_m\} = \begin{cases} 1, & \text{as } f(X_m) > f(X_{m-1}) \\ \exp\{[f(X_m) - f(X_{m-1})]/T_m\}, & \\ & \text{as } f(X_m) \leqslant f(X_{m-1}) \end{cases} \qquad (6)$$

i.e., $P\{X_m\} = 1$ as $f(X_m) > f(X_{m-1})$ and in the other case a random number $\eta \in [0,1]$ is generated: the point X_m will be the initial for a new step ($(m+1)$-st) of the algorithm and in the formula (4) it will take the place of X_{m-1} as $\eta < \exp\{[f(X_m) - f(X_{m-1})]/T_m\}$, and the point X_{m-1} remains as the initial one for a new step, otherwise

$$T_m = c/\lg(1 + m_0 + m), \qquad (7)$$

m is the number of the step, c is some positive constant, m_0 is some constant from $[1, \infty)$.

We propose such a way to solve the problem of c selection for the algorithm. The formula (6) for the case $f(X_m) \leqslant f(X_{m-1})$ can be transformed taking into account (7):

$$P\{X_m\} = \exp\left\{\frac{0.43\ln(1 + m_0 + m)}{c}[f(X_m) - f(X_{m-1})]\right\} =$$

$$= (1 + m_0 + m)^{\frac{0.43}{c}[f(X_m) - f(X_{m-1})]}.$$

When we use some initial probability $P\{X_1\} = \delta$ as $m = 1$, the constant c can be expressed :

$$c = 0.43[f(X_1) - f(X_0)]\ln(2 + m_0)/\ln\delta \ ,$$

where X_0 and X_1 are such, that $f(X_0) > f(X_1)$. Then (6) will have such a form (for $m = 2, 3, \ldots$) :

$$P\{X_m\} = \begin{cases} 1, & \text{as } f(X_m) > f(X_{m-1}) \\ (1+m_0+m)^{\frac{f(X_m)-f(X_{m-1})}{f(X_1)-f(X_0)} \frac{\ln \delta}{\ln(2+m_0)}}, & \\ & \text{as } f(X_m) \leqslant f(X_{m-1}) \end{cases} \quad (8)$$

The problem of extremal parameter grouping belongs to the class of partitioning problems. The simulated annealing was used to solve the partitioning problems by Trzebiatowski (1985), namely, for the partitioning of networks. The imperfection of the algorithms proposed by Trzebiatowski, preventing to use them in a general case, is such: in each iteration a new partition is constructed by interchanging two (or more) elements from different groups (so the number of elements in any group remains the same doing the optimization).

3.1. The algorithms SA1, SA2 and SA3. The algorithms SA1 and SA2 are the concrete realizations of the algorithm proposed above. Their peculiarities are the following:

1. The algorithm starts from the point $\overline{X}_0$; the m-th step of algorithm is as follows (initially $X_0 = \overline{X}_0$, $m = 1$, $P\{X_m\} = 1$):

$$X_m^i = X_{m-1}^i + \xi^i, \; i = \overline{1,n}, \quad (9)$$

where $\xi^i, i = \overline{1,n}$, are integer numbers taking the values by (5) (i.e. by 5(a) or (5b)). $X_0 = X_m$ as $f(X_m) \geqslant f(X_{m-1})$. $X_1 = X_m$ and further calculations are being performed by the formulae (5),(8) and (9) starting from the $(m+1)$-st step (various strategies for a further m_0 and m selection are presented in the sixth peculiarity) as $f(X_m) < f(X_0)$.

2. The relation of m and k (the strategy of k changing) is the following: $p-1$ steps of the algorithm are being performed for every fixed value of k. Thus, k corresponds to the number

of the variable, the value of which is changed, when values of other variables are fixed. The totality of the calculations above,when the value of k runs from 1 to n, is called the iteration of the algorithm. One iteration requires no more $n(p-1)$ calculations of the function $f(\cdot)$ values.

3. The restriction (3) is taken into account as follows. Let the new value of k be fixed and the parameter x_k be the only one in its group. Then the algorithm passes to the next k value.

4. Let the value of k be fixed. There may be some coincidental argument points among $p-1$ points in which it is necessary to calculate the $f(\cdot)$ value. Calls to the program realizing $f(\cdot)$ are not reiterated in such a situation, but the number of calculated $f(\cdot)$ values is increased.

5. Only the necessary part of the function $f(\cdot)$ is recalculating when it is necessary to compute the unknown $f(\cdot)$ value.

6. The referred below strategies for m_0 and m initial selection were investigated:

a) $m_0 = 1$, m is equal to the number of function $f(\cdot)$ values calculated, the calculation of $f(\overline{X}_0)$ value is also taken into account;

b) $m_0 = 1$, $m = 1$;

c) m_0 is the number of $f(\cdot)$ calculations used to obtain the best value of $f(X_0), m = 1$.

7. SA1 uses (5a), SA2-(5b).

8.$X^i_m = A^i$ as $X^i_{m-1} = B^i$ and $\xi^i = 1$, and $X^i_m = B^i$ as $X^i_{m-1} = A^i$ and $\xi^i = -1$ are used when (5a) is used.

The algorithms above use some probabilistic choosing (described by (5a) or (5b)) of a point for the next calculation of the objective function value. The algorithm below uses the deterministic choosing. Let us denote it by SA3. Its peculiarities are the following:

1. This peculiarity differs from SA1 and SA2 first one as

follows: in (9) ξ^k takes the values from D^k in a deterministic way and $\xi^i = 0$ $(i = \overline{1,n},\ i \neq k)$ during calculations when k value is fixed.

2. The strategy of k changing and ξ^k selection is the following: for every fixed value of k,the value of ξ^k runs from $A^k - X^k_{m-1}$ to $B^k - X^k_{m-1}$ with the exception of $\xi^k = 0$. The totality of the calculations above, when the value of k runs from 1 to n, is called the iteration of SA3.

3. The third, fifth and sixth peculiarities of SA1 and SA2 remain the same for SA3.

Some properties of the algorithms are presented below.

Note 5. SA3 coincides with A2 (Dzemyda,1987a) as $\delta = 0$.

Let us denote:

1) $X_M = (X^1_M, \ldots, X^n_M)$ is the current point of the algorithm after M iterations (X_M uncertainly coincides with the point, where the $f(\cdot)$ maximum is achieved after M iterations);

$$2)\ S_1 = \bigcup_{i=1}^{n} [\{(X^1_M, \ldots, X^{i-1}_M, 1, X^{i+1}_M, \ldots, X^n_M), \ldots, (X^1_M, \ldots, X^{i-1}_M, p, X^{i+1}_M, \ldots, X^n_M)\} \backslash \{X_M\}];$$

3) S is the subset of S_1 consisting of the points, which satisfy the restriction (3).

Note 6. Let X_M be the current point of SA3 after M iterations. If the current point remains the same after the $(M+1)$-st iteration (i.e.$X_{M+1} = X_M$), X_M corresponds to the parameters partition conforming to the local maximum of functional I_1, i.e., the transferring of any parameter from its group to another one would not increase I_1 value.

Proof. The algorithm SA3 is constructed in such a manner that the values of $f(\cdot)$ will be calculated on all points of

S during the $(M+1)$-st iteration if $X_{M+1} = X_M$. The unoccurance of transition to any point of S during the $(M+1)$-st iteration indicates that $f(\cdot)$ values on the points of S are less than $f(X_M)$. Thus, X_M corresponds to the locally optimal partition and the Note is proved.

Note 7. Let X_M be the current point of SA2 after M iterations. If it remains the same during consequent iterations,the probability of calculation of $f(\cdot)$ values on all points of S in consequent iterations grows to 1.

Proof. Let the condition of Note 7 be met. Let us examine any point from S. The algorithm SA2 is constructed in such a manner, that the probability of calculation of $f(\cdot)$ value in this point during the $(M+1)$-st iteration is equal:

$$p_1 = 1 - \left(\frac{p-2}{p-1}\right)^{p-1},$$

and during $M+1, \ldots, M+t$ iterations is equal (if the current point in these iterations remains as X_M):

$$p_t = 1 - (1-p_1)^t.$$

$$\lim_{t\to\infty} p_t = 1.$$

The probability of calculation of values on all points of S in consequent iterations grows to 1 because the probability of calculation of $f(\cdot)$ value in any point of S grows to 1 by increasing of iteration number. The Note is proved.

Note 8. Let X_M be the current point of SA2 after M iterations. If it remains the same during some consequent iterations and if $f(\cdot)$ values are calculated on all points of S during these iterations, X_M corresponds to the parameters partition, conforming to the local maximum of functional I_1.

Proof. From the conditions of Note 8 it follows,that $f(\cdot)$ values on the points of S are less than $f(X_M)$. Thus, X_M corresponds to the locally optimal partition and the Note is proved.

3.2. Experimental investigation. Investigations were performed on the EC-1045 computer . The algorithms are written in FORTRAN. The presented results are obtained for the third mode of m_0 and m selection because the optimal partitioning quality of all modes was found to be similar (only the optimal value of δ differs). The termination condition of all investigated algorithms was the limited number of iterations (the more complex termination condition for practical problems may be selected). 20 random correlation matrices were generated. $n = 20$, $p = 4$, the following initial parameter partition was chosen: $A_1 = \{x_1, \ldots, x_5\}$, $A_2 = = \{x_6, \ldots, x_{10}\}$, $A_3 = \{x_{11}, \ldots, x_{15}\}$, $A_4 = \{x_{16}, \ldots, x_{20}\}$. The averaged results of the maximization of $f(\cdot)$ are presented in Fig.1, Fig.2 and Table 1.

In Table 1 IT is the averaged number of iterations, where the maximum of $f(\cdot)$ was obtained, and NL is the number of performed iterations by the SA type algorithms. IE is the averaged number of calculations of maximal eigenvalues during one iteration (calculations of such a type take the great part in computational expenditures of the algorithms investigated). The results of SA1, SA2 and SA3 performance are presented in Table 1 for the found optimal δ. The performance of the deterministic algorithms is illustrated on the basis of A1 and A2 (Dzemyda,1987a). The presented IT values for A1 and A2 lines mean the averaged number of performed runnings through all parameters. The last two lines of Table 1 correspond to the case when SA1 and SA2 were used for the initial partition of parameters and then the result was made more precise by A1.

The results of SA1, SA2 and SA3 performance are

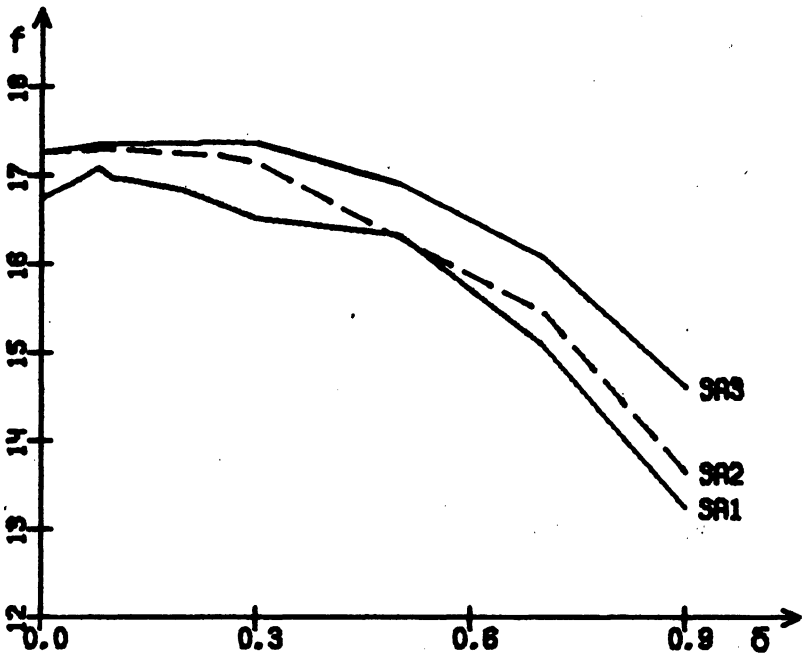

Fig.1. The dependence of SA1, SA2 and SA3 performance on δ

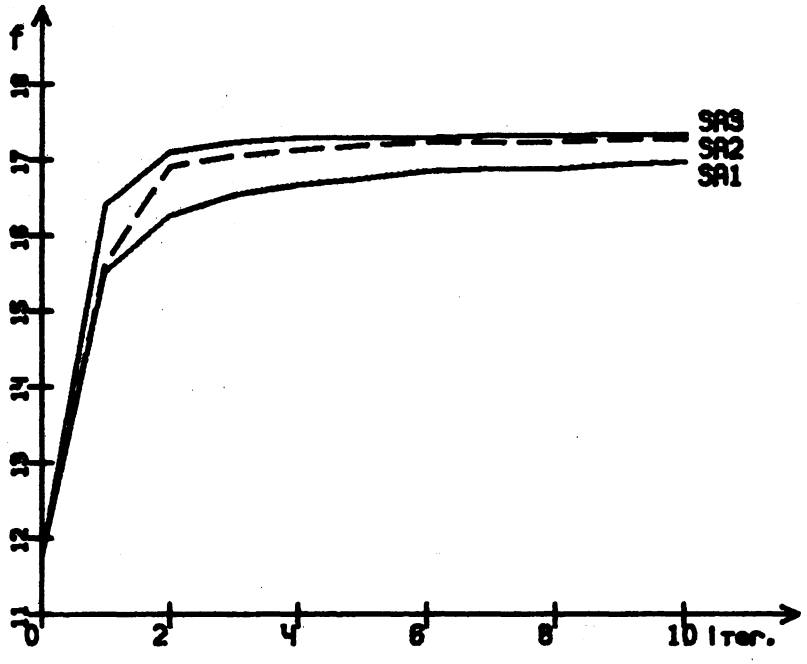

Fig.2. The dependence of SA1, SA2 and SA3 performance on iteration number

presented in Fig.1 (in dependence on δ) and in Fig.2 for the found optimal values of δ (in dependence on iteration number).

Table 1. The averaged results of test problems solving

Algorithm	δ	NL	IT	IE	f(.)
SA1	0.08	20	7.2	55.4	17.09
SA2	0.10	20	6.5	61.9	17.29
SA3	0.25	20	6.25	80.0	17.37
A1	——	——	2.8	7.6	17.02
A2	——	——	3.5	80.0	17.26
SA1+A1	0.08	2	2.0	——	17.11
SA2+A1	0.10	2	2.0	——	17.12

4. Conclusions. The experimental investigation showed that by using simulated annealing it is possible to find the better partition of parameters in comparison with that found by the deterministic algorithms. We observed the tendency to a significant improvement of the partition during the initial iterations. Later the results were specified. Thus some iterations of simulated annealing algorithms also can be used for the initial partition of parameters. Then the deterministic algorithms, which are faster but require a good initial partition, can be used. The practical use of SA1 (not for the initial partition) is doubtful because its partitioning quality is similar to that of A1, but A1 requires considerably less computational expenditures. The partitioning quality of SA3 is better than of SA2. The improvement of partition stops after the greater number of SA2 iterations in comparison with SA3. However, one iteration of SA3 performs more calculations of the maximal eigenvalues of symmetrical matrices. The algorithms proposed can be modified for the solving of any clustering problem.

REFERENCES

Anderberg, M.R. (1973). *Cluster Analysis for Applications.* Academic Press, New York. 359pp.

Bhanot, G. (1988). The Metropolis algorithm. *Reports on Progress in Physics,* **51**(3), 429–457.

Braverman, E.M. (1970). Methods of extremal grouping of parameters and problem of apportionment of essential factors. *Avtomatika i Telemechanika,* **1**, 123–132 (in Russian).

Braverman, E.M., and I.B.Muchnik (1983). *The Structural Methods for the Empiric Data Processing.* Nauka, Moscow. 464pp. (in Russian).

Chiang, T.S., and Y.Chow (1988). On the convergence of annealing processes. *SIAM J.Contr. and Optim.,* **26**(6), 1455–1470.

Dzemyda, G. (1987a). On the extremal parameter grouping. In A.Žilinskas (Ed.), *Teorija Optimaljnych Reshenij,* Vol.12. Inst. Math. Cybern. Lith.Acad.Sci., Vilnius. pp. 28–42 (in Russian).

Dzemyda, G. (1987b). LP–search with extremal problem structure analysis. In N.K.Sincha and L.A.Telksnys (Eds.), *Stochastic Control, Proceedings of the 2nd IFAC Symposium.* Pergamon Press, Oxford. pp. 499–502.

Dzemyda, G. (1988). The algorithms of extremal parameter grouping. In A.Sydow, S.G.Tzafestas and R.Vichnevztsky (Eds.), *Mathematical Research,* Band 46. Akademie–Verlag, Berlin. pp. 133–136.

Dzemyda, G. (1990). Grouping on the sphere. In *Teorija Optimaljnych Reshenij,* Vol.14 (to appear, in Russian).

Dzemyda, G., V.Tiešis and J.Valevičiene (1989). The composition of the algorithms of minimization and of objective function calculation. In *Wissenschaftliche Berichte der Technischen Hochschule Leipzig, 14th IFIP Conference on System Modelling and Optimization,* Heft 7. Technischen Hochschule, Leipzig. pp. 60–62.

Dzemyda, G., and J.Valevičienė (1988). The extremal parameter grouping in cluster analysis. In *Teorija Optimaljnych Reshenij,* Vol.13. pp. 36–53 (in Russian).

Gidas, B. (1985). Non–stationary Markov chains and convergence of the annealing algorithm. *J.Stat.Phys.,* **39**(1), 73–131 .

Harman, H.H. (1976). *Modern Factor Analysis,* 3rd ed. University of Chicago Press, Chicago .

Lundy, M., and A. Mees (1986). Convergence of an annealing algorithm. *Mathematical Programming,* **34**(1), 111–124 .

Mitra, D., F.Romeo and A.Sangiovanni–Vincentelli (1986). Convergence and finite–time behaviour of simulated annealing. *Adv. Appl. Prob.,* **18**(3), 747–771.

SAS (1982). *SAS User's Guide: Statistic,* 1982 ed. SAS Institute Inc., Cary. 584pp.

Trzebiatowski, G.V. (1985). Thermodynamic simulation procedure for partitioning problems. In: H.–M.Voigt (Ed.), *Informatic: Informationen–Reporte,* Vol.1,No.12. Institut für Informatik und Rechentechnik, Berlin. pp. 86–96.

Valevičienė, J., G.Dzemyda, V.Tiešis and A.Karminskij (1988). The combination of minimization and the objective function calculation. In *Teorija Optimaljnych Reshenij,* Vol.13. pp. 17–35 (in Russian).

Received November 1989

G. Dzemyda received the Degree of Candidate of Technical Sciences from the Kaunas Polytechnic Institute, Kaunas, Lithuania, in 1984. He is a senior researcher at the Department of Optimal Decision Theory, Institute of Mathe-

matics and Cybernetics, Lithuanian Academy of Sciences. His research interests include interaction of optimization and data analysis.

E. Senkienė received the Degree of Candidate of Physical and Mathematical Sciences from the Vilnius University, Vilnius, Lithuania, in 1974. She is a senior researcher at the Department of Optimal Decision Theory, Institute of Mathematics and Cybernetics, Lithuanian Acad. Sci. Her research interests include stochastic processes.

J. Valevičienė is a researcher at the Department of Optimal Decision Theory, Institute of Mathematics and Cybernetics, Lithuanian Acad. Sci. She is interested in programming systems and their application to optimization problems solving.