Nuclei of Monotonic Systems on a Semilattice of Sets [*]

I. B. Muchnik and L. V. Shvartser UDC 519.876

Algorithms developed in the method of monotonic systems for the identification of extremal subsets are modified for the analysis of the condition of an arbitrary finite semilattice of sets. We show how to generate these subsemilattices in a goal-directed manner in the form of solutions of systems of inequalities defined by a collection of monotonic systems. The modified technique is applied to construct new automatic classification algorithms.

1. Introduction

Monotonic systems are used as models of the identification of subsystems (nuclei) of a complex system [1] when any subset of elements of the given system may be potentially chosen. Formally, this means that the criterion of "nuclearity" (or autonomy) of a subsystem is defined on the entire family of subsets of a given set. Yet the system may be subject to a priori constraints that exclude the choice of some subsets as a nucleus. In particular, in the problem of aggregation of empirical data, the aggregates to be selected from the set objects being analyzed are often expected to satisfy inconsistent requirements. Such situations are studied using models that are capable of allowing for the given constraints.

In this context, we generalize the method of monotonic systems so as to be able to effectively find their nuclei on a part of the family of subsets of a given set. Only the families that form a semilattice [1] in the lattice of all subsets are considered feasible.

We provide a general formulation of the nucleus identification problem and some solution algorithms. The algorithms use "oracle" procedures to find the unity (the largest subset by inclusion) of the given semilattice. Two specific realizations of this procedure are described. We show that the proposed procedures can be used to construct applied algorithms that approach aggregation of empirical data as constrained extremum-seeking problems.

---

[*] Moscow. Translated from Avtomatica i Telemekhanika, No. 8, pp. 116 – 125, August, 1989. Original article submitted February 26, 1988.
005 – 1179/89/5008 – 1095 $12.50

[1] A lattice of subsets is a family of subsets closed with respect to union and intersections; a semilattice is a family closed with respect to one of these operations. In what follows, we consider semilattices closed with respect to union (upper semilattices).

2. Constrained Extremum Seeking on a Monotonic System

Consider a finite set W, $|W| = N$. Let 2^W be the set of its subsets, S some semilattice of the lattice 2^W $(S \subseteq 2^W)$, and $\langle W, \pi, F \rangle$ a monotonic system [1], i.e., each pair (i, H), $i \in H \subseteq W$, is assigned the weight $\pi(i,H)$ such that

$$\pi(i,H) \geq \pi(i,H'), \text{ if } H' \subseteq H, \tag{1}$$

and each $H \in 2^W$ is assigned the number

$$F(H) = \min_{i \in H} \pi(i,H). \tag{2}$$

Consider the problem

$$F(H) \to \max_{H \in S}, \tag{3}$$

which we call the problem of finding a constrained extremum (a constrained nucleus) on a monotonic system. It is a generalization of the problem of finding the nucleus of a monotonic system [1]. By the quasiconvexity property of $F(H): F(H_1 \cup H_2) \geq min(F(H_1), F(H_2))$ $\forall H_1, H_2 \subseteq W$ [2] it follows that the set of solutions of the problem (3), as well as the set of nuclei of a monotonic system, is a semilattice (given that S is a semilattice). Similarly to the problem of finding a nucleus of a monotonic system [1], which only considers algorithms that determine the largest (by inclusion) nucleus, we will examine algorithms that determine the largest constrained nucleus – the unity of the corresponding subsemilattice of the semilattice S.

The algorithms assume that a procedure is available for finding the unity E of the semilattice $S \cap [\varnothing, X]$, where $[\varnothing, X]$ is the set-theoretical interval $\varnothing \subseteq H \subseteq X$ for any $X \in 2^W$. We call this procedure P1.

First consider a generalization of the procedure $LAYER(u)$, which is the basis of the most efficient algorithm to find the largest nucleus of a monotonic system [3,4]. [2] The input infor-

[2] Actually, the $LAYER(u)$ algorithm was first introduced as a sequence of W elements in concord with the level u, see "Stable Coalition in Monotonic Games", Mullat, October, 1979, submitted in 1978, http://www.datalaundering.com/download/monogame.pdf, rem. by JM.

mation for this procedure includes the monotonic system $\langle W, \pi, F \rangle$, the semilattice S, the threshold $u \in \Re$ and the set $X \in 2^W$.

Procedure $LAYER(u,S,X)$.

Step 0. $Y = X$.

Step 1. Using P1, find the unity E of the semilattice $S \cap [\varnothing, Y]$. If $E = \varnothing$, then stop, else determine the set $I_u = \{ i \mid i \in E, \pi(i,E) \leq u \}$. If $I_u = \varnothing$, then stop.

Step 2. Set $Y = E \setminus I_u$. If $Y = \varnothing$, set $E = \varnothing$ and stop; else go to 1.

Definition. A level-u set $(u \in \Re)$ (a nonstrict level-u set) of the semilattice $\Delta \in 2^W$ by $F(H)$ is the set

$$\Delta_u = \{ H \in \Delta \mid F(H) > u \}, \ (\overline{\Delta}_u = \{ H \in \Delta \mid F(H) \geq u \}).$$

The level set Δ_u $(\overline{\Delta}_u)$ for any $u \in \Re$ is obviously a semilattice.

Theorem 1. The result E produced by the procedure $LAYER(u,S,X)$ is unity of the semilattice $(S \cap [\varnothing, X])_u$.

Theorem 1 is proved in the Appendix.

Algorithm A1 to solve Problem (3).

Step 0. $X = W$. Find the unity E using P1 and compute $u' = \min_{i \in E} \pi(i,E)$ and $u'' = \max_{i \in E} \pi(i,E)$. Let

$$u = \frac{u' + u''}{2}. \quad (4)$$

Step j. Apply the procedure $LAYER(u,S,X)$ and obtain one of the following results.

1. $E = \varnothing$. The set $u'' = u$; transform u by (4) and again apply $LAYER(u,S,X)$; continue updating u by (4) and applying the procedure $LAYER(u,S,X)$ until $E \neq \varnothing$ is obtained.
2. $E \neq \varnothing$. Set $u = F(E)$ and apply $LAYER(u,S,X)$. If the result E' is an empty set $(E' = \varnothing)$, the last nonempty E is the sought result; stop. Otherwise $(E' \neq \varnothing)$, set $u' = u$, update u by (4), set $X = E'$, and go to step ($j+1$).

In each step we obtain either the unity of the level-u semilattice S or the signal $E = \varnothing$, which indicates that $u > \max_{H \in S} F(H)$. It thus obviously follows that Algorithm A1 will produce the largest (by inclusion) solution of the problem (3), i.e., the sought nucleus.

As shown in [5,6], in addition to the maxima of $F(H)$ on 2^W (the nuclei), local maxima (quasinuclei) are also of interest in applications. Special algorithms have been proposed for determining these local maxima [1,4]. [3] Below we construct two algorithms that use a similar technique to find all the local extrema of problem (3).

Algorithm A2.

Step 0. $X = W$. Apply P1 to find E_1 and set $u = F(E_1)$.

Step j ($j \geq 1$). Apply the procedure $LAYER(u,S,E_j)$ with $u = F(E_j)$. Its result is E_{j+1}. If $E_{j+1} = \varnothing$, then stop, else to step ($j+1$).

Theorem 2. Algorithm A2 constructs the sequence $\langle E_1,...,E_p \rangle$ of unities [4] of all level sub-semilattices of the semilattices S by $F(H)$.

Theorem 2 is a direct corollary of the Theorem 1 if we note that E_1 is the unity of the maximum level subsemilattice.

Corollary. E_p is the nucleus of the system $\langle W,\pi,F \rangle$ on S, i.e., the unity of the semilattice of solutions of problem(3).

Algorithm A3.

Step 0. $X = W$. Apply P1 to obtain $E_0 = E$. Order all the elements from $X \setminus E$ in the arbitrary way in the form of a sequence $J = \langle j_1,..., j_k \rangle$.

Step j ($j \geq 1$). Find an element $i \in E$ such that $\pi(i,E) = F(E)$. Fix the value of $F(E)$ and set $j_k = i$. Take $X = E \setminus \{i\}$ and apply P1 to find the next $E_j = E$. Arrange the elements from $X \setminus E$ in arbitrary order following ($j+1$).

As the value of the variable k, assign the number of elements in the sequence constructed so far. If $E = \varnothing$, the set E_p arg max $F(H)$, $p = j-1$, where the maximum is over all $H_k = \{j_k, j_{k+1},..., j_N\}$: $H_k \in S$, $k = \overline{1,N}$, stop. Otherwise, go to step ($j+1$).

---

[3] In [1] only the largest nuclei/kernel has been detected, rem. by JM.

[4] The sets E_i, $i = 1,..., p$, are quasinuclei (local maxima) of $F(H)$ on S in the sense that $F(E_i) > F(H)\ \forall H \in S : E_i \subset H$ (see [4] and only they are quasinuclei.

It is easy to show that A3, like A2, produces the unities of all the level subsemilattices of the semilattice S by $F(H)$ $\left(\left\langle E_1,...,E_p \right\rangle\right)$. Unlike A2, which applies the procedure $LAYER(u,S,X)$, A3 finds solutions by constructing the sequence J, which is similar to the defining sequence introduced in [1] for the analysis of the monotonic system on the entire Boolean 2^W.

3. Constructive Procedures to find the Unity of a Semilattice of Sets

The efficiency of the procedure to find unities of semilattices essentially depends on how these semilattices are specified. In this section, we describe two particular specifications – by a system of inequalities of quasiconvex functions and by a family of systems of representatives of partitions generated by the cuts of the classification tree [7,8]. In the next section we will show that both these methods naturally arise in applied problems of empirical data aggregation.

First Technique [9].

Given are m monotonic systems $\langle W,\pi_1,F_1\rangle,...,\langle W,\pi_m,F_m\rangle$. By the first assertion of Theorem 1 and the fact that the intersection of a finite set of semilattices is a semilattice, we obtain that the solution set $\Theta \subseteq 2^W$ of the system of inequalities

$$\begin{aligned} &F_1(H) \geq u_1, \\ &\;.\;\;.\;\;.\;\;.\;\;.\;\;. \\ &F_m(H) \geq u_m. \end{aligned} \qquad (5)$$

is a semilattice. Its unity can be found by the following polynomial procedure.

Procedure P1-1.

Step 0. $Y = X$.

Step 1. Successively find the unities of all the semilattices, each defined by one of the inequalities in (5). To this end, apply the algorithm of [1] that construct the defining sequence on Y. We obtain the family of sets $\left(E^1,...,E^m\right)$.

Step 2. $Y = \bigcap_{i=1}^{m} E^i$. If Y satisfies all the inequalities (5), then stop, otherwise return to step 1.

That the procedure P1-1 finds the unity of the semilattice $\Theta \cap [\varnothing, X]$ follows from two easily proved facts:

1) the unity of $\Theta \cap [\varnothing, X]$ satisfies (5),

2) if $S_1,...,S_m$ are semilattices, $E^1,...,E^m$ their unities, and E_n is the unity of the semilattices $\bigcap_{i=1}^{m} S_i$ then $E_n \subseteq \bigcap_{i=1}^{M} E^i$.

Second Technique.

Let S be a sublattice of the lattice 2^W with known unity E_S. If S is specified so that for all $x_1, x_2 \in W$ the question (A): is true that

$$\exists H \in S : x_1 \in H \, , \; x_2 \notin H \, , \tag{6}$$

can be answered in polynomial time, then, as shown in [10], we can easily construct a polynomial procedure which, given any $X \subseteq E_S$, constructs the intersection $\overline{X}$ of all $H \in S$, such that $H \supseteq X$.

Procedure to Construct $\overline{X}$ [10].

Step 0. $\overline{X} = X$. Take $y \in E_S \setminus X$.

Step 1. Test (6) for $x_2 = y$, x_1 is any element from X . If the answer is yes for all $x_1 \in X$, then $y \notin \overline{X}$; if the answer is no foe at least one $x_1 \in X$, then set $\overline{X} = \overline{X} \cup \{y\}$.

Step 2. Take a previously unexamined $y \in E_S \setminus X$ and return to step 1; if no such y are found, then stop.

The set $\overline{X}$ in [10] is called the closure [5] of X . Using the closure construction procedure, we obtain the following effective procedure to find the unity E_X of $S \cap [\varnothing, X]$, $X \subseteq W$.

[5] Operation of the closure hereinafter will utilized, and for construction of the stability estimate classification of objects from W is set up.

Procedure P1-2.

Step 0. Choose $i_0 \in X \cap E_S$ such that $\overline{\{i_0\}} \subseteq X$. If no such i_0 exists, then $E_X = \varnothing$ (E_X is the unity of the semilattice $S \cap [\varnothing, X]$) and stop. Otherwise, set $H_0 = \{i_0\}$.

Step 1. Take $i_0 \in (X \setminus H_0) \cap E_S$ such that $\overline{H_0 \cup \{i_0\}} \subseteq X$. If no such i_0 exists, then $E_X = H_0$ and stop. Otherwise, set $H_0 = \overline{H_0 \cup \{i_0\}}$ and go to 1.

For the special case when S is a lattice of systems of representatives of a hierarchical classification of elements of the set W, we will construct a polynomial procedure that answers the question (A). It obviously ensures that the proposed procedure P1-2 runs in polynomial time.

Let T be some hierarchical classification of elements of the set W defined by a tree of classes (taxons), where W is the tree root and the leaves are all the single-element sets $\{i\}$, $i \in W$. Among all the cuts of this tree separating the root from the leaves, identify those cuts, which ensure that the dominating classes (taxons) located on the same side of the cut as the leaves, form a partition of the set W. Denote the set of these partitions by $\tilde{R}$. Adjoin to $\tilde{R}$ the partition consisting of the single class W.

Number the original sets W in an arbitrary sequence. The mapping that associates to each $R \in \tilde{R}$, a set of numbers from W by the rule: to each class from R associate the minimum number in W contained in this class, is easily shown to be one-to-one. It is easy to show that the image of the set $\tilde{R}$ in 2^W is the sublattice S of the Boolean 2^W with $E_S = W$ and $0_S = \{1\}$. For this sublattice, we can construct an answer to question (A) in the following way.

Starting with the vertex $\{x_1\}$ move up through the tree along an appropriate chain (we denote it by J_{x_1}) and stop when we reach the taxon included in a similar chain originating from the vertex $\{x_2\}$ (J_{x_2}). If the taxon obtained in this way or the preceding taxon in the chain J_{x_2} has the number x_2, then the answer to question (A) is no, because in this case we cannot construct a cut that contains the taxon with the number x_1 and does not contain the taxon with the number x_2. Otherwise, the answer to question (A) is yes, because the cut separating the vertex obtained in this way from the preceding vertex in J_{x_2} and the last vertex with the number x_1 from its successor in J_{x_1} satisfies the give requirements.

4. Aggregation of Empirical Data by the Algorithm that finds the Nucleus of a Monotonic System on a Semilattice

Consider two types of aggregation problems, which reduce to finding the constrained extremum on a monotonic system. The first type is characteristic of the case when the same set of objects is described by several different groups of parameters or by several association matrices determining different forms of interaction between the parameters. The second type involves a single association matrix between the objects. This matrix is used to construct some hierarchical classification that generates the family of classifications $\tilde{R}$. It is required to choose a classification, which is the best by a given criterion and at the same time satisfies some stability test.

The first type will be constructed in the framework of the method of linguistic data analysis [11]. The space of parameters, where the set of objects being analyzed is represented by a set of points, is partitioned into $(k+1)$ subspaces. The set of objects being analyzed is considered separately in each of these subspaces.

Let $X^q = \| x^q_{i\,j} \|$ be the q-th data matrix, where $x^q_{i\,j}$ is the value of parameter j of the q-th group A_q ($q = 1,\ldots,k+1$) on the object i ($x^q_{i\,j} \geq 0$). Define a monotonic system on the direct product of the set of all objects and the set A_q of the parameters of the q-th group. To this end, we introduce the weighting function

$$\pi(\,(i,j),\left(H_0, H^q_n\right)) = \sum_{k\in H_0} x^q_{i\,j} + \sum_{p\in H^q_n} x^q_{i\,p}\ , \tag{7}$$

where $\left(H_0, H^q_n\right)$ is some subset of pairs such that $i \in H_0 \subseteq W$, $j \in H^q_n \subseteq A_q$. Then

$$F_q(\left(H_0, H^q_n\right)) = \min_{\substack{i\in H_0 \\ j\in H^q_n}} \pi_q(\,(i,j),\left(H_0, H^q_n\right)) \tag{8}$$

is an estimate of the level of values of the elements $x^q_{i\,j}$ of the submatrix of the matrix X^q defined by the set of rows H_0 and the set of columns A_q.

From the set of submatrices $\{X^q, q = 1,...,k+1\}$, choose one as the goal submatrix. For definiteness, let this be X^{k+1}. Then the sought aggregation problem can be stated as the constrained extremum problem

$$F_{k+1}((H_0, H^{k+1})) \to \max_{H_0 \subseteq W, H^{k+1} \subseteq A_{k+1}},$$
$$F_i((H_0, H^i)) \geq u_i, \; i = \overline{1,k}. \tag{9}$$

This problem coincides with (3) up to the notation. Indeed, the constraints in (9) define a semilattice S on the set 2^W of feasible first elements in the pair (i, j), which in turn induces the semilattice $S \times 2^{A_{k+1}}$ on the set of subsets of the pair (i, j) such that $i \in H_0 \in S$, $j \in H^{k+1} \in 2^{A_{k+1}}$. In other words, problem (9) may be viewed as problem (3) on the set $W = W \times A_q$. The algorithm to find the unity of the constraint semilattice in this case is an obvious modification of P1-1.

In this case, the solution of system (9) involves simultaneous search for $(k+1)$ parameter subsets $(H_n^q)^{max}$ in the submatrices $X^1,...,X^{k+1}$ such that the subset H_0^{max} of objects satisfying (3) comprises objects with the highest values of their characteristics on $(H_n^q)^{max}$.

The analysis of an organization defined by $(k+1)$ association matrices $B_q = \|b_{ij}^q\|$, $q = \overline{1,(k+1)}$ that represent various interactions on the same set W of organizational functions [5,6] is similarly reduced to the solution of problem (3). Consider each matrix A_q separately and define on these matrices monotonic systems with the weighting functions

$$\pi_q(i, H) = \max_{j \in H} b_{ij}^q \tag{10}$$

and the criterion

$$F_q(H) = \min_{i \in H} \pi_q(i, H). \tag{11}$$

Then the sought problem is obtained as soon as we choose the goal ("principal") matrix A_{k+1}.

Let us consider the second type of aggregation problems. Let $P = \|p_{ij}\|$ be the matrix of distances between objects of the set W, and T some hierarchical classification of the objects W, based on the matrix P. The classification can be obtained by examining simultaneously several partitions formed by constructing the graphs $G(W,P,u_q)$ ($u_1 < u_2 < \ldots$ is a monotone increasing collection of thresholds) on the vertex set W and then partitioning them into connected components. The adjacency matrix $M_q(G) = \left\| m^q_{ij} \right\|$ of the graph $G(W,P,u_q)$ is defined as

$$m^q_{ij} = \begin{cases} 1, & \text{if } \rho_{ij} \geq u_q, \\ 0 & \text{otherwise.} \end{cases} \tag{12}$$

A hierarchical classification arises also in taxonomy algorithms that identify a tree of minimum length on the complete weighted graph $G(W,P)$. Finally, agglomerative procedures [7,8] directly construct a hierarchical classification.

The problem is to select a classification R from the set $\tilde{R}$ of all possible classifications generated by the given hierarchy T which is, first, the best by some criterion and, second, sufficiently stable in the previously defined sense.

As the choice criterion of the sought classification R, we use the functional from [13]

$$F(H_R) = \min_{i \in H_R} \sum_{j \in H_R} \rho_{ij}, \tag{13}$$

where H_R is a system of representatives of the partition R. Unlike [13], we maximize the criterion (13) not on the entire set 2^W, but only on the semilattice S of the systems of representatives of the partitions from $\tilde{R}$. As a stability estimate of the classification R, we use a function that reflects the variability of the classification R in the process of agglomeration (this is consistent with the accepted procedure of hierarchical analysis [7]. Denote by E_X the unity of the lattice $S \cap [\varnothing, X]$.

Introduce the characteristic $J(R) = \sum_{S=1}^{m} \sum_{i,j \in R_S} \rho_{ij}$ (m is the number of classes in R) and define on S the function $I(H_R) = J(R)$. It is easy to see that this function has the following property: for all $H, H' \in S$ such that $H' \subseteq H$,

$$I(H) - I(H') \leq I(E_{H \setminus \{i\}}) - I(E_{H' \setminus \{i\}}). \tag{14}$$

Then construct on 2^{E_S} the monotonic system

$$\pi(i,X) = I(E_{\overline{X}\setminus\{i\}}) - I(\overline{X}),\; i \in X \subseteq 2^{E_S}.$$

Take $\lambda(H_R) = \min_{i \in H_R} \pi(i, H_R)$ as the stability estimate of the classification R.

We thus obtain the following problem. On the lattice S of the system of representatives of a classification from $\widetilde{R}$ generated by the hierarchy T find a partition R such that

$$F(H_R) \to \max,\; \lambda(H_R) \geq u.$$

This problem obviously can be solved by any of the algorithms A1, A2, or A3, applying the procedure P1-2 to find the unity of $S \cap [\varnothing, X]$.

In conclusion, two remarks.

1. On the application level, the thresholds u_i, $i = 1,...,m$, in problem (3) with constraints of the form (5) obviously should be made as large as possible. However, they cannot be taken arbitrarily large because, first, this may reduce the sought maximum of F and, second, not every choice of thresholds is consistent. For some collection of thresholds, the intersection of the semilattices defined by corresponding inequalities is empty, i.e., no element in 2^W is common to all these semilattices. We therefore propose a scheme that refines the choice of threshold in this sense and guarantees a nontrivial solution of (5).

Assume that the collection of thresholds $\{u_1,...,u_m\}$, once substituted in (5), produces a solution of problem (3) H^*, such that $F(H^*) = u_0$. Denote by A the set of all vectors $\{u_1, u_2, ..., u_m\}$ such that the collection $\{u_1, u_2, ..., u_m\}$ generates a nontrivial (in sense of (5)) semilattice. A vector from the set A, which is Pareto-unimprovable on A satisfies the above requirements. The following procedure, starting with an arbitrary vector from A, produces unimprovable vector on A.

Procedure P. Enumerating $i = 1,...,m$, successively solve the problems $F_i(H) \to max$, $F_{j \neq i}(H) \geq u_i$, $F(H) \geq u_0$. When each solution $\widetilde{H}_i$ is found, set $u_i = F(\widetilde{H}_i)$.

Theorem 3. The point H^* produced by procedure P is an equilibrium point in the following sense: there is no $H \subseteq W$ such that for some $i \in \{0,...,m\}$ we have $F_i(H) > F_i(H^*)$ and $F_j(H) \geq F_j(H^*)$ for $j \in \{0,...,m\}$ $j \neq i$ (here $F_0(H) = F(H)$).

The proof follows obviously from the fact that H^* is the unity of the corresponding semilattice defined by the constraints.

The constraints in the procedure P can be selected in a different order, leading in general to different equilibrium points. Moreover, the equilibrium point produced by the procedure obviously depends on the initial choice of the threshold vector $\{u_1,...,u_m\}$ for the procedure P.

2. Our analysis has mainly focused on the relationship between two objects – semilattices of sets and monotonic systems. Let us consider this relationship in detail.

Let $S \subseteq 2^W$ be some family of subsets. Consider the hull $\widehat{S}$ of the set S, i.e., a semilattice $\widehat{S}$ such that $S \subseteq \widehat{S}$ and there is no semilattice S' such that $S \subset S' \subset \widehat{S}$. Note that the hull-construction procedure adjoins to the set S all the possible unions of its subsets.

Theorem 4. Assume that the procedure P1 for the semilattice $\widehat{S}$ is polynomial. Then the following two statements are equivalent:

1) S is a semilattice,
2) for any monotonic system algorithm A3 P1 to S produces a solution of problem (3).

Theorem 4 is proved in the Appendix. It is similar to the well-known theorem on the correspondence of the matroid and the greedy procedure [14]. Note that algorithm A3 is also superficially similar to the greedy procedure.

APPENDIX

Proof of Theorem 1. Let $E \neq \varnothing$. Since $I_u = \varnothing$, then $F(E) > u$, i.e., $E \in (S \cap [\varnothing, X])_u$. It is easy to show by induction that the sets I_u in each step consist of elements that do not belong to any of $H \in (S \cap [\varnothing, X])_u$. Thus there exists no $H \in (S \cap [\varnothing, X])_u$, such that $E \subset H$. Therefore, E is the unity of the semilattice $(S \cap [\varnothing, X])_u$. The case $E = \varnothing$ obviously corresponds to $(S \cap [\varnothing, X])_u = \varnothing$. ■

Proof of Theorem 4. 1) → 2) was proved above. Let us prove 2) → 1) by contradiction. Let $H_1,H_2 \in S$; $H_1 \cup H_2 \notin S$. Take $i^* \in H_1$, $i^* \notin H_2$. Construct a monotonic system in the following way:

1) $\forall H : H \subseteq H_1 \cup H_2,\ i^* \in H$,

$$\pi(i,H)=\begin{cases}0 \text{ for } i \notin H_1 \cup H_2,\\ 1 \text{ for } i = i^*,\\ 2 \text{ for } i \in H_1 \cup H_2,\ i \neq i^*,\end{cases}$$

2) $\forall H : H \subseteq H_1 \cup H_2 \setminus \{i^*\}$

$$\pi(i,H)=\begin{cases}0 \text{ for } i \notin H_2,\\ ½ \text{ for } i \in H_2.\end{cases}$$

The subset H_1 with the value $F(H_1)=1$ is clearly a solution of problem (3). at the same time, algorithm A3 will generate the subset H_2 with the value $F(H_2)=½$. A contradiction. ■

LITERATURE CITED

1. J. E. Mullat, "Extremal subsystems of monotonic systems. I-III," *Avtomat. Telemekh., No. 5*, 130-139 (1976); *No. 8*, 169-178 (1976); *No. 1*, 143-152 (1977), http://www.datalaundering.com/download/extrem01.pdf, http://www.datalaundering.com/download/extrem02.pdf, http://www.datalaundering.com/download/extrem03.pdf.
2. A. V. Malishevskii, "On the properties of order functions of sets," Abstracts of Papers of All-Union Seminar on Optimization and Its Applications [in Russian], *Akad. Nauk TadzhSSR*, Dushanbe (1986), pp. 151-152, http://www.datalaundering.com/download/order.pdf.
3. J. E. Mullat and L. K. Võhandu, "Monotonic systems in scene analysis," Symp. Math. Processing of Cartographic Data, *TPI, Tallinn* (1979), pp. 63-66.
4. E. N. Kuznetsov, I. B. Muchnik, and L. V. Shvartser, "Monotonic Systems and their properties, "in: Analysis of Nonnumerical Information in Social Sciences [in Russian], *Nauka*, Moscow (1985), pp. 29-56.
5. E. N. Kuznetsov and I. B. Muchnik, "Monotonic systems for the analysis of organization structures," in: Methods of Analysis of Multivariate Economic Information [in Russian], *Nauka, Novosibirsk* (1981), pp. 71-84.
6. E. N. Kuznetsov and I. B. Muchnik, "Analysis of the distribution of management functions in organizations," *Avtomat. Telemekh., No. 10*, 119-127 (1982), http://www.datalaundering.com/download/organiza.pdf.

7. S. C. Johnson, “Hierarchical clustering schemes,” *Psychometrica*, *32*, 241-254 (1967).

8. G. A. De Soete, “Least squares algorithm for fitting additive trees to proximity data,” *Psychometrica*, *48*, 621-626 (1983).

9. I. B. Muchnik and L. V. Shvartser, “Algorithms to find constrained extrema on monotonic systems,” Abstracts and Papers at All-Union Seminar on Optimization and Its Applications [in Russian], *Akad. Nauk. TadzhSSR*, Dushanbe (1986), pp.161-162.

10. M. Grotschel, L. Lovasz, and A. Schriver, “The ellipsoid method and its consequences in combinatorial optimization, “ *Combinatorica*, *1*, *No. 2*, 169-197 (1981).

11. E. M. Braverman and I. B. Muchnik, Structural Methods of Empirical Data Analysis [in Russian], *Nauka*, Moscow (1983).

12. B. G. Mirkin, Analysis of Qualitative Features and Structures [in Russian], *Finan. Stat.*, Moscow (1980).

13. E. N. Kuznetsov, “Analysis of the structure of the association matrix by constructing a monotonic system,” *Avtomat. Telemekh.*, *No. 7*, 128-136 (1980), http://www.datalaundering.com/download/couplem.pdf.

14. Ch. Papdimitriou and K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity, *Prentice Hall*, Englewood Cliffs, N. J. (1982).