

ON STRUCTURAL RESOURCE OF MONOTONE RECOGNITION¹

Hasmik Sahakyan, Levon Aslanyan

Abstract: Algorithmic resources are considered for elaboration and identification of monotone functions and some alternate structures are brought, which are more explicit in sense of structure and quantities and which can serve as elements of practical identification algorithms. General monotone recognition is considered on multi-dimensional grid structure. Particular reconstructing problem is reduced to the monotone recognition through the multi-dimensional grid partitioning into the set of binary cubes.

1. Introduction

Monotone Boolean functions have an important role in research area since they arise in various application models, such as design of schemes, pattern recognition, etc.

Monotone Boolean functions are studied in different viewpoint and are known as objects of high complexity. First results, obtained by Miceev [M, 1959] and Korobkov [K, 1965], characterize Sperner families in unit cube. After enormous investigations and overcoming difficulties, Korshunov [K, 1981] obtained the asymptotical estimate of the number of Monotone Boolean functions. It is characteristic that analytical formulas are not known at this point.

Another cluster of research work solves problems of algorithmic identification of monotone Boolean functions. Hansel [H, 1966] constructed the best algorithm in sense of Shannon criterion, then Tonoyan [T, 1979] constructed a similar algorithm with minimal use of memory. Later on there obtained some generalizations for multi-valued cube. Alekseev [A, 1976] generalized Hansel's result, Katerinotchkina [K, 1978] gave precise description of structure of Sperner families.

It is typical that for multi-valued cube there is no explicit formula not only for the number of monotone functions, but also for the cardinality of middle layer. It makes difficult choice of algorithms for a concrete problem and estimation of their complexity.

Below in this paper some algorithmic resources are considered for elaboration and identification of grid defined monotone functions and some alternate structures are brought, which are more explicit in sense of structure and quantities and which can serve as elements of practical identification algorithms.

2. Learning monotone functions on multi-valued cube

Let Ξ_{m+1}^n denotes the grid of vertices of n dimensional, $m+1$ valued cube, i.e. the set of all integer-valued vectors $S = (s_1, s_2, \dots, s_n)$ with $0 \leq s_i \leq m$, $i = 1, \dots, n$. For any two vertices $S' = (s'_1, s'_2, \dots, s'_n)$ and $S'' = (s''_1, s''_2, \dots, s''_n)$ of Ξ_{m+1}^n we say that S' is greater than S'' , $S' \geq S''$ if $s'_i \geq s''_i$, $i = 1, \dots, n$. We call pair of vectors S', S'' comparable if $S' \geq S''$ or $S' \leq S''$, otherwise these vectors are incomparable. Set of pair wise incomparable vectors composes a Sperner family.

Usually vertices of Ξ_{m+1}^n are placed schematically among the $m \cdot n + 1$ layers of Ξ_{m+1}^n according to their weights, – sums of all coordinates. Vector $\tilde{0} = (0, \dots, 0)$ is located on the 0-th layer; then the i -th layer consists of all vectors, with the weight i . An element of i -th layer might be greater than some vector from the $i-1$ -th layer, exactly by one component and exactly by one unit of value (such vector pairs are called neighbors and are connected by an edge). The vector $\tilde{m} = (m, \dots, m)$ is located on the $m \cdot n$ -th layer.

Consider a binary function f on Ξ_{m+1}^n , $f : \Xi_{m+1}^n \rightarrow \{0,1\}$. We say that f is monotone if for any two vertices S', S'' notion $S' \geq S''$ implies $f(S') \geq f(S'')$. The vector $S^1 \in \Xi_{m+1}^n$ is a lower unit of monotone function

¹ The research is supported partly by INTAS: 04-77-7173 project, <http://www.intas.be>

f if $f(S^1) = 1$ and for arbitrary $S \in \Xi_{m+1}^n$, such that $S < S^1$ it is true that $f(S) = 0$. The vector $S^0 \in \Xi_{m+1}^n$ is an upper zero of monotone function f if $f(S^0) = 0$ and for arbitrary $S \in \Xi_{m+1}^n$, such that $S > S^0$ it is true that $f(S) = 1$.

Above defined monotone functions are known also as increasing monotone in contrast with a decreasing monotone function. A function f is decreasing monotone if for any two vertices S', S'' , $S' \geq S''$ implies $f(S') \leq f(S'')$. For f decreasing monotone, the vector $S^1 \in \Xi_{m+1}^n$ is an upper unit if $f(S^1) = 1$ and for any $S \in \Xi_{m+1}^n$, such that $S > S^1$ we get $f(S) = 0$. The vector $S^0 \in \Xi_{m+1}^n$ is a lower zero of function f if $f(S^0) = 0$ and for any $S \in \Xi_{m+1}^n$, such that $S < S^0$ we get $f(S) = 1$.

In case when $m = 1$ the definitions above lead to ordinary monotone Boolean functions defined on binary cube E^n .

Let a monotone function f be defined with the help of an oracle which, receiving any vector $S \in \Xi_{m+1}^n$, gives the value $f(S)$. The problem is in identification of arbitrary monotone function f by as far as possible small number of accesses to the oracle. Similar problems are interested in finding all or the maximal/minimal upper zeros or alternatively the minimal/maximal lower 1's of the given Boolean function. Consider an example. Let, it is given a set of n linear inequalities. A consistent subset of inequalities is coded by a vertex of E^n , where we define f as 0. The problem of finding the maximal consistent subset of inequalities is a known hard problem and the use of oracle reduces the problem to solving several subsystems of inequalities, which is just an alternative way of solving the main problem. The monotone binary function recognition on Ξ_{m+1}^n is the weighted inequalities version of the above given example model.

In [A, 1976] an algorithm of complexity $\leq |M| + \lfloor \log_2 m \rfloor \cdot |N|$ is constructed to learn the binary monotone functions above the multi-valued discrete grid, which generalizes the Hansel's method ([H, 1]) for the case of monotone Boolean functions, here M and N denote the sets of vertices of middle layers of multi-valued grid/cube, i.e. layers which contain vectors with sums of coordinates equal to $\lfloor (m \cdot n) / 2 \rfloor$ and $\lfloor (m \cdot n) / 2 \rfloor + 1$ respectively. It is also proven that the complexity of the algorithm is approximately \sqrt{n} time less than the whole number of vertices of the grid.

3. Ξ_{m+1}^n partitioning through binary cubes

In this section an alternate approach to traditional means is considered for identification of monotone functions defined on Ξ_{m+1}^n . First Ξ_{m+1}^n is partitioned into binary cube like structures and then Hansel's method is applied for identification of monotone Boolean functions. This approach may serve as a separate element of practical identification algorithms.

In Ξ_{m+1}^n we distinguish several classes of vectors.

Upper and lower homogeneous vectors. A vector of Ξ_{m+1}^n is called an upper h -vector (upper homogeneous) if the values of all its coordinates are at least $m/2$ for even m , and are at least $(m+1)/2$ for odd m .

Similarly, a vector of Ξ_{m+1}^n is called a lower h -vector (lower homogeneous) if the values of all its coordinates are at most $m/2$ for even m , and are at most $(m-1)/2$ for odd m .

We denote by \hat{H} the set of all upper h -vectors and by \check{H} the set of all lower h -vectors. The cardinalities of sets \hat{H} and \check{H} are equal to $((m+1)/2)^n$ for odd m and to $(m/2+1)^n$ - for m even.

Middle vectors \tilde{m}_{mid+} and \tilde{m}_{mid-}

$\tilde{m}_{mid+} = ((m+1)/2, \dots, (m+1)/2)$ and $\tilde{m}_{mid-} = ((m-1)/2, \dots, (m-1)/2)$ for odd m and $\tilde{m}_{mid+} = \tilde{m}_{mid-} = (m/2, \dots, m/2)$ for even m . \tilde{m}_{mid+} is located on the $n \cdot (m+1)/2$ -th layer of Ξ_{m+1}^n (the lowest layer that contains vector from \widehat{H}) and \tilde{m}_{mid-} is located on the $n \cdot (m-1)/2$ -th layer of Ξ_{m+1}^n (the highest layer that contains vector from \check{H}) for odd m ; for even m the vector $\tilde{m}_{mid+} = \tilde{m}_{mid-}$ is located on the layer $n \cdot m/2$ and this is the only common vector of \widehat{H} and \check{H} .

Vertical equivalent vectors. Let $S', S'' \in \Xi_{m+1}^n$. S' and S'' are called ν -equivalent (vertically equivalent) if one of them is obtained from the other by inverting some coordinates (that is replacing some coordinates by their complements up to the m).

For a given vector S denote by $V(S)$ the class of all ν -equivalent vectors to S and call it the ν -equivalency class of S . This structure $V(S)$ is congruent to a cube E^k , where k is the number of coordinates of S not equal to $m/2$ (this is valid for even m). For odd m $k = n$. It is also evident, that $V(S') = V(S)$ for an arbitrary $S' \in V(S)$.

In $V(S)$ we distinguish two vectors $\widehat{S} = (\widehat{s}_1, \dots, \widehat{s}_n)$ and $\check{S} = (\check{s}_1, \dots, \check{s}_n)$ - upper and lower vectors, which coordinates are defined as follows:

$$\widehat{s}_i = \begin{cases} s_i, & s_i \geq m - s_i \\ m - s_i, & s_i < m - s_i \end{cases} \text{ and } \check{s}_i = \begin{cases} m - s_i, & s_i \geq m - s_i \\ s_i, & s_i < m - s_i \end{cases}, i \in \overline{1, n}.$$

These are the only vectors of $V(S)$ that belong to sets \widehat{H} and \check{H} respectively.

Consequently, for any S the class of its ν -equivalency can be constructed by the upper vector \widehat{S} and/or by the lower vector \check{S} of that class by coordinate inversions. ν -equivalency classes of different upper homogeneous vectors are none intersecting.

This proves partitioning of the whole structure Ξ_{m+1}^n through binary cube like vertical extensions of elements of \widehat{H} or \check{H} . The following formula shows the picture of factorization of structure of Ξ_{m+1}^n through these cubical elements: $(m+1)^n = \sum_{k=0}^n (C_n^k \cdot 2^k \cdot (m/2)^k) = \sum_{k=0}^n (C_n^k \cdot m^k)$ for even m and $(m+1)^n = ((m+1)/2)^n \cdot 2^n$ for odd m .

Thus, we get $|\widehat{H}|$ disjoint subsets, congruent to binary cubes, which cover Ξ_{m+1}^n . Notice that if we construct the corresponding binary cubes, then a pair of vertices, comparable in a binary cube, is comparable also in Ξ_{m+1}^n . Therefore monotonicity in Ξ_{m+1}^n implies monotonicity in all received binary cubes and starting by a monotone function in Ξ_{m+1}^n and reconstructing the implied functions on cubes the initial function will be reconstructed in a unique way.

We recall now the problem of identification of monotone binary functions defined on Ξ_{m+1}^n .

By Hansel's result [H, 1966] an arbitrary monotone Boolean function with k variables can be identified by $C_k^{\lfloor k/2 \rfloor} + C_k^{\lfloor k/2 \rfloor + 1}$ accesses to the oracle.

Hence an arbitrary monotone function defined on Ξ_{m+1}^n can be identified by $\sum_{k=0}^n (C_n^k \cdot (m/2)^k \cdot (C_k^{\lfloor k/2 \rfloor} + C_k^{\lfloor k/2 \rfloor + 1}))$ accesses for even m and by $((m+1)/2)^n \cdot (C_n^{\lfloor n/2 \rfloor} + C_n^{\lfloor n/2 \rfloor + 1})$ - for odd m .

4. Characteristic vectors of subsets partitions of E^n and identification of monotone functions in \hat{H}

For a given $m, 0 \leq m \leq 2^n$ let ψ_m denote the set of all **characteristic vectors of partitions** of m -subsets of E^n . A nonnegative integer-valued vector $S = (s_1, s_2, \dots, s_n)$ is called characteristic vector of partitions of a vertex subset $M, M \subseteq E^n$ if its coordinates are the sizes of partition-subsets of M by coordinates x_1, x_2, \dots, x_n , which are the Boolean variables composing E^n . s_i is the size of one of partition-subsets of M in the i -th direction and $m - s_i$ is the complementary part of partition. For simplicity we will later assume that s_i is the size of the partition with $x_i = 1$.

If $m \neq 0$ then ψ_m is not empty. It is also evident that $\psi_m \subseteq \Xi_{m+1}^n$. As other exceptions distinguish between the 2 boundary cases: if $m=1$ then $\psi_m = \Xi_{m+1}^n$ and so $|\psi_m| = |\Xi_{m+1}^n| = 2^n$; if $m=2^n$ then $|\psi_m| = 1$ and the vector with all coordinates 2^{n-1} indeed belongs to Ξ_{m+1}^n .

In [S, 2006] the entire description of ψ_m is given in terms of Ξ_{m+1}^n geometry. It is particularly proven that the main problem of describing characteristic vectors can be moved from the Ξ_{m+1}^n to the area of \hat{H} (\check{H}), where the vector set ψ_m has monotonous structure, - it corresponds to the units of some monotone decreasing binary function defined on \hat{H} (monotone increasing binary function defined on \check{H}).

Figure 1 illustrates the sets $\hat{H} \cap \psi_m$ and $\check{H} \cap \psi_m$ for even and odd m values, correspondingly.

Thus for entire description of ψ_m it is sufficient to consider all monotone functions defined on \hat{H} or \check{H} . We shall restrict ourselves to the consideration of decreasing monotone functions defined on \hat{H} . Let $\hat{\psi}_m$ be the subset of $\hat{H} \cap \psi_m$ consisting of all upper units of corresponding monotone function.

In [AS, 2001] additional resource is introduced: L_{min} and L_{max} , - minimal and maximal numbers of layers of \hat{H} , - are calculated, such that all vectors of $\hat{\psi}_m$ are located between them. It importantly follows that the entire description of ψ_m is reduced to the identification of monotone functions with upper units between the layers L_{min} and L_{max} .

Summarizing all the above consideration we come to the conclusion:

1) Algorithmic resource of learning monotone binary functions defined on Ξ_{m+1}^n includes structures such as:

- generalized Hansel's method and constructions, for monotone binary functions defined on multi-valued cube,
- Ξ_{m+1}^n partitioning through binary cube like vertical extensions of the elements of \hat{H} together with applying Hansel's result for monotone Boolean functions defined on that cubes,

2) For the entire description of ψ_m we reduce the problem to \hat{H} becoming able to possess with additional resources:

- learning monotone binary functions defined on \hat{H} by means of generalized Hansel's method,

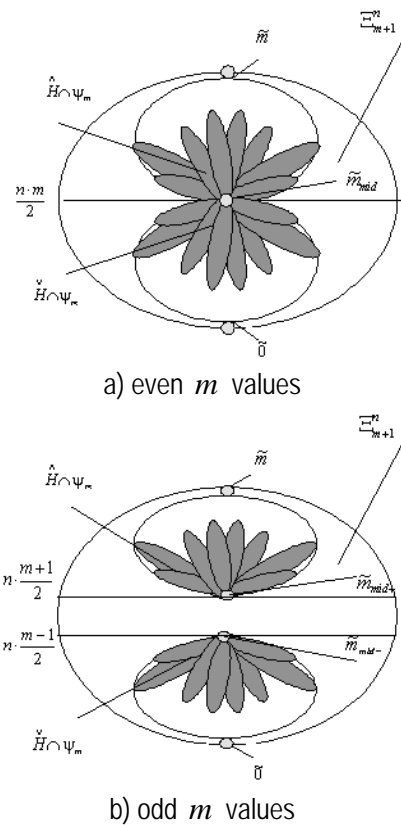


Figure 1

- partitioning \hat{H} into binary cube like vertical extensions of its upper homogeneous elements and applying Hansel's method for them,
- identifying monotone functions defined on \hat{H} with use of additional information on location of their upper units through L_{min} and L_{max} .

The choice of concrete resource set depends on requirements of certain applications.

Conclusion

Algorithmic resources are considered for elaboration and identification of monotone functions. Current research proposes two new components - partitioning the multi-valued cube through binary cube like vertical extensions of its upper homogeneous elements; and learning upper homogeneous area through the analogous partitioning. The choice of concrete resource depends on requirements of certain application.

Bibliography

- [M, 1959] V. Mickeev. On sets, containing maximal number of pair wise incomparable Boolean vectors, Prob.Cyb, 2, 1959.
- [K, 1965] V. Korobkov. On monotone functions of algebra of logic, Prob.Cyb, 13, 1965.
- [K, 1981] A. Korshunov. On the number of monotone Boolean functions, Prob.Cyb, 38, 1981.
- [H, 1966] G. Hansel. Sur le nombre des fonctions booleennes monotones de n variables, C.R.Acad.Sci., Paris, 262, N.20, 1966.
- [T, 1979] G. Tonoyan. Cain partitioning of n-cube vertices and deciphering of monotone Boolean functions, Journal of Computational Mathematics and Math. Physics, 1979, vol. 19, N.6.
- [A, 1976] V. Alexeev. On deciphering of some classes of monotone many valued functions, Journal of Computational Mathematics and Math. Physics, 1976, vol. 16, N.1.
- [K, 1978] Katerinokhina N., On sets, containing maximal number of pair wise incomparable n-dimensional k-valued vectors, Mathematical notes, v. 24, no, 3, Sept. 1978.
- [AS, 2001] L. Aslanyan, H. Sahakyan. On the boundary cases of partitioning of subsets of the n-dimensional unit cube, "Computer Science & Information Technologies" Conference, Yerevan, September 17-20, 2001
- [S, 2006] H. Sahakyan. Numerical characterization on n-cube subset partitioning, submitted to Discrete Applied Mathematics.

Authors' Information

Hasmik Sahakyan – e-mail: hasmik@jpia.sci.am

Levon Aslanyan – e-mail: asl@sci.am

Institute for Informatics and Automation Problems, NAS Armenia, P.Sevak St. 1, Yerevan-14, Armenia

DYNAMIC DISTRIBUTION SIMULATION MODEL OBJECTS BASED ON KNOWLEDGE

Alexander Mikov, Elena Zamyatina, Konstantin Osmehin

Abstract: *This paper presents the process of load balancing in simulation system Triad.Net, the architecture of load balancing subsystem. The main features of static and dynamic load balancing are discussed and new approach, controlled dynamic load balancing, needed for regular mapping of simulation model on the network of computers is proposed. The paper considers linguistic constructions of Triad language for different load balancing algorithms description.*

Keywords: *Distributed calculations, distributed simulation, static load balancing, dynamic load balancing, expert systems*

ACM Classification Keywords: *1.6 Simulation and Modeling 1.6.8 Types of Simulation - Distributed: 1.2 Artificial Intelligence 1.2.5 Programming Languages and Software - Expert system tools and techniques*