Fault Identification Problems in Boolean Concepts Learning

Gabi Shafat
Afeka Tel Aviv Academic College of Engineering
AFEKA
218 Bney Efraim rd. Tel Aviv 69107, Israel
gabis@afeka.ac.il

Ilya Levin
School of Education, Tel Aviv University
TAU
Ramat Aviv, Tel Aviv, 69978, Israel
ilia1@post.tau.ac.il

ABSTRACT

This study examines the relationship between properties of Boolean functions and the structural complexity measure of Boolean functions. Boolean functions with linear, symmetrical, and monotone properties were examined from the point of the structural complexity measure. The impact of Boolean functions' properties on the structural complexity measure was examined on the example of solving fault identification type problems in digital circuits realized in NMOSFET technology.

INTRODUCTION

Boolean concepts are given a great deal of attention in Engineering studies in general and Software Engineering and Electronic Engineering studies in particular. Bachelors of engineering students are exposed to and acquire an understanding of Boolean concepts in the second semester Digital Systems course during their first year. Digital Systems course is based on Boolean algebra as its fundamental mathematical background. A large number systems naturally lend themselves to Boolean representation [1].

Various models are used for different types of content; the multitude of models creates a perceptual difficulty and complexity in understanding the Boolean concept when a transition is made from one content domain to another.

The standard approach in teaching the Digital Systems course is based on representing Boolean functions in a form of Truth tables and minimizing using Karnaugh maps. Understanding Boolean concepts has a huge influence on the students' cognitive and conceptual development. Acquiring Boolean concepts, representing them and using the concepts are the foundation for more advanced courses such as Digital Electronics, DSP (Digital Signal Processing), and so forth. Boolean concepts are the foundation for logical thinking, which the students use to cope with solving problems as part of their academic studies and later as engineers in the advanced technological world. This coping creates a fair amount of complexity from both a human and technological aspect. Human concepts learning is a discipline expressed in cognitive development in learning and problem solving processes. It receives a great attention in psychology studies. From the technological point of view, complexity is determined according to Occam's razor. This principle states that no more entities than is necessary should be used; when there are different explanations for the same phenomenon, the simplest one with the smallest number of concepts should be chosen. The principle is used as a guideline in the information doctrine regarding the minimal length of a message. Complexity is determined according to the resources required to create a specific object. In Computer Science, the complexity of a string of characters is measured by so-called Kolmogorov complexity, which is the length of a minimal computer program whose output is the string [2]. RISC processors are capable to perform the minimal set of simple instructions are based on these principles [3]. The hardware complexity is traditionally measured by a number two-input logic gates implementing the corresponding specification. "Logic gates" are the basic units through which each digital device can be realized. Our study is focused on a Structural Complexity. With respect to these complexity measures, difficulties in solving problems were examined, such as faults identification in digital systems. In addition, the relationship between symmetry, linearity and monotony properties of a Boolean concept and the Structural Complexity measures was examined.

STRUCTURAL COMPLEXITY

An important aspect of concept learning theory is the ability to predict the level of difficulty in learning different types of concepts. In this respect, the study of Boolean concepts, obviously, is an important topic of Engineering Education. Boolean concepts can be represented by Boolean expressions comprising basic logic operations: negation, disjunction ("or"), and conjunction ("and"). These types of Boolean concepts have been studied extensively by [4] SHJ. The SHJ studies are focused on Boolean concepts of three binary variables, where the concept is equal to "1" for 4 out of 8 possible combinations and "0" for the remaining 4 combinations. Since some of such 70 possible Boolean concepts are congruent, they can be categorized as the same type into six subcategories.

Structural complexity of a Boolean function should provide an answer to the question: What is it about the internal structure of Boolean function from any category that makes them harder to learn than Boolean function from a different category?

A known approach for answering to the above question is based on Boolean derivatives, which were introduced as part of the context of Boolean algebra, coding, error correction and detection electronic circuit [5]. For the purpose of quantitative measure of the structural complexity, the degree of categorical invariance must be calculated. At the foundation of the theory, the distances are explored between the combinations in which the function receives the value of "1" and the degree of invariance between the two Boolean function variables. The quantitative calculation of the complexity is defined as a structural complexity measure of the concept [6].

Given that the Boolean function $F(x_1, x_2, ..x_i, ..x_n)$ for n variables from any category, the partial derivatives of the Boolean function are:

$$\frac{\partial F(x_1, x_2, ..x_i, ..x_n)}{\partial x_i} = F(x_1, x_2, ..x_i, ..x_n) \oplus F(x_1, x_2, ...\overline{x_i}, ...x_n)$$

The Logic manifold – Lm is defined as the number of combinations in which there is a change in the function's value compared to the original function for all the function's variables, as follows:

$$Lm_i = \frac{\left(\left|F_0 \cap F_{pi}\right|\right)}{\left|F_0\right|} \quad , \quad i = 1, 2, 3, ..., n$$

where:

$|F_0|$ - is the number of combinations where the original function is "1".

$$F_{pi} = \left(\frac{\tilde{\partial} F(x)}{\partial x_i}\right)$$

$|F_{pi}|$ - is the number of combinations where there is a change in the function's value relative to the original function for each of the variables.

The structural complexity variable is directly proportional to the number of combinations in which the original function's value is "1", and inversely proportional to the degree of invariance of each of the function's variables, according to the following formula:

$$SC = \frac{|F_0|}{\left[\left(\sum_{i=1}^{n}(Lmi)^2\right)^{\frac{1}{2}} + 1\right]}$$

Vigo's account of the invariance of Boolean function does not specify how individuals learn Boolean function. The approach is based on the assumption that cognitive processes could detect invariance by comparing a set of instances to the set yielded by the partial derivative of each variable; therefore this measure may provide an answer on the degree of difficulty of the Boolean function of a category compared to another category.

PROPERTIES OF BOOLEAN FUNCTIONS

Properties of functions are important for practical applications in a variety of fields, such as artificial intelligence, decentralized systems, content problems, and more.

Given that $F = \{f_1, f_2, ..., f_m\}$ is a set of Boolean functions, if every arbitrary function can be realized using the basic logical functions "or", "and" and "not" – then the collection of F functions will be called universal. Universal functions can be realized using "and" and "not" only or using "or" and "not" only. These trivial conclusions stem from Post's statement [7].

From a broad category of Boolean functions, the family of symmetric Boolean functions is one of the most important. Symmetric functions have independent values as large as the input, and the output depends only on the number of bits whose value is "1" in a vector, the input for which the value of the function is "1". The number of symmetric Boolean functions for n variables is 2^{n+1}. A family of symmetric functions contains many basic functions, such as screening functions, and includes functions known as dual functions [8],[9]. Symmetric functions were studied extensively and enabled boundaries to be drawn around the complexity of realizing digital circuits. A general result determines that a digital circuit of size $O\left(\sqrt{N}\right)$ with logical gates and depth 3 with polynomial weights is sufficient to calculate all the symmetric functions for a given number of variables [10]. Symmetric functions can be synthesized with fewer logical elements. Determining the symmetry property of a Boolean function is an important and difficult component in CAD – computer aided design problems.

Additional important families of Boolean functions are linear functions and monotone functions. Monotone functions are well known and highly important. They are applicable in numerous applications, decision-making system, synthesis of reliable systems [11].

FAULT PROBLEMS

Each of us occasionally finds ourselves trying to diagnose a fault or failure in some sort of system. For example, why is the car not starting? Why is the food not tasty? Why is the remote control not responding? Diagnosing failures and faults is ubiquitous in the professional lives of engineers, doctors, etc. It is inseparable part of engineering practice, were failures diagnostics of systems is required. Diagnosing failures is a type of problem solving and it is one of the cognitive skills that is learned and studied in psychology, computer science and artificial intelligence [12],[13]. Prior research has investigated fault finding in network tasks [14]. One research aim has been to automate the process by; for instance, devising automated systems that are capable of diagnosing faults in large-scale industrial systems, such as power plants [15].

A failure occurs in a circuit or system when there is a deviation from a specific defined behavior. A fault is a physical defect that may or may not cause a failure. Diagnosing failures is not a synthetic or analytic action and not studied in the field of complex digital systems. In digital circuits, a fault may be caused by a disconnection in the conductors though which the signal passes, a short in the reference potential or the source supplying the electrical voltage, a short or disruption between the signal processors. In general, the fault's effect is represented using a model that represents the change that the fault caused to the circuit signals. There are two fault models: Bridging fault and Stuck-open fault. A "Bridging fault" is caused by a short between two conductors in the digital system. A "Stuck-open fault" is caused by a disconnection between two terminals in the digital system. The tendency is usually to ascribe the fault to the unit adjacent to where the fault occurred [16].

This study will examine the relationship between the properties of Boolean functions and the structural complexity measure of the Boolean function in solving fault identification problems in digital systems using a Boolean function.

The research hypothesis was that Boolean function properties impact the problem solving success beyond the structural complexity measure of the Boolean function. Boolean functions with specific properties can lower the cognitive complexity in learning and problem solving processes. The properties of Boolean functions that were examined in the study are: Symmetric, linear, and monotone functions.

EXPERIMENT

Thirteen Boolean functions, F_1 to F_{13}, were selected for the research objectives and hypotheses (Table 1). The students were tested with fault identification problems using 2 questionnaires where thirteen digital circuits realized with NMOSFET technology. Each of the circuits represents one of the 13 Boolean functions tested accordingly. Fig. 1 shows a circuit consistent with F_5 in Table 1.

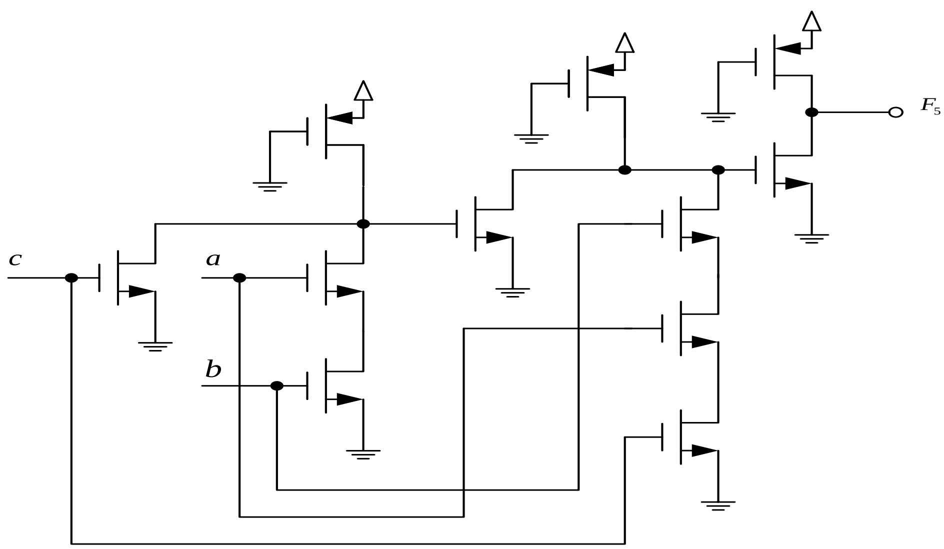

Fig. 1

Boolean Function	Structural Complexity	Fault problems		Properties
		Accuracy (%) Stuck-open	Accuracy (%) Bridging	
$F_1(a,b,c)=\bar{b}(a+c)$	1.54			-
$F_2(a,b,c)=\bar{a}c+b\bar{c}$	2.14			-
$F_3(a,b,c)=\bar{a}\,\bar{b}+ab=\overline{a\oplus b}$	2.00			S+L
$F_4(a,b,c)=a(b+c)+bc$	2.14	100	93	S+M
$F_5(a,b,c)=(\bar{a}+\bar{b})\bar{c}+abc=\overline{c\oplus(ab)}$	2.34			-
$F_6(a,b,c)=a+bc+\bar{b}\,\bar{c}=a+\overline{b\oplus c}$	2.79			-
$F_7(a,b,c)=a\,b\,\bar{c}+\bar{a}\,b\,c+a\,\bar{b}\,c=b(a\oplus c)+a\,\bar{b}\,c$	3	100	98	S
$F_8(a,b,c)=\bar{a}(b+c)+bc$	2.14			-
$F_9(a,b,c,d)=a\,c\,\bar{d}+a\,\bar{b}\,\bar{c}+\bar{a}\,\bar{b}\,c\,d$	2.95			-

Boolean Function	Structural Complexity	Fault problems: Accuracy (%) Stuck-open	Fault problems: Accuracy (%) Bridging	Properties
$F_{10}(a,b,c)=a(\bar{b}\ c+\bar{c}\ b)+\bar{a}(\bar{b}\ \bar{c}+b\ c)=\overline{a\oplus b\oplus c}$	4.00			S+L
$F_{11}(a,b,c)=a(b\ c+\bar{c}\ \bar{b})+\bar{a}(\bar{b}\ c+b\ \bar{c})=a\oplus b\oplus c$	4.00			S+L
$F_{12}(a,b,c,d)=a(b+c+d)+b(d+c)+cd$	4.48	99	97	S+M
$F_{13}(a,b,c,d)=\bar{a}(\bar{b}+\bar{c}+\bar{d})+\bar{b}(\bar{d}+\bar{c})+\bar{c}\ \bar{d}$	4.48	99	97	S+M

Table 1. The 13 concepts were tested during the experiment and their descriptions according to Structural Complexity, Property of Boolean functions Symmetry (S), Linearity (L) and Monotone (M) and fault problem accuracy.

In Part A, for each circuit the subjects were asked to find the Boolean function of the circuit. In Part B, they received a Boolean function that the circuit conducts as a result of a short type fault and were asked to discover the location of the fault and explain their answer. In Part C, they received a Boolean function that the circuit conducts as a result of a short type fault and were asked to discover the location of the fault and explain their answer. The questionnaire was divided into two. The first part had circuits corresponding to Boolean functions F_1, F_2, F_4, F_5, F_7, F_{10}, F_{12} in Table 1 and the second questionnaire had circuits corresponding to Boolean functions F_3, F_6, F_8, F_9, F_{11}, and F_{13} in Table 1. The questionnaires were administered same day and 90 minutes were allotted to solve each questionnaire. The success of solving the problems was measured based on the correct solution during the allotted time. All the solutions were examined compared to Structural complexity measures.

An example for calculating Structural Complexity for concept 7 (F_7 in Table 1):

$$F_7(a,b,c)=\sum(3,5,6)=a\ b\ \bar{c}+\bar{a}\ b\ c+a\ \bar{b}\ c=b(a\oplus c)+a\ \bar{b}\ c$$

$$\frac{\partial F_7(a,b,c)}{\partial a}=(a\ b\ \bar{c}+\bar{a}\ b\ c+a\ \bar{b}\ c)\oplus(\bar{a}\ b\ \bar{c}+a\ b\ c+\bar{a}\ \bar{b}\ c)=\sum(1,2,3,5,6,7)$$

$$F_{7ap}=\frac{\bar{\bar{\partial}}F_7(a,b,c)}{\bar{\bar{\partial}}a}=\sum(0,4)$$

$$\frac{\partial F_7(a,b,c)}{\partial b}=(a\ b\ \bar{c}+\bar{a}\ b\ c+a\ \bar{b}\ c)\oplus(a\ \bar{b}\ \bar{c}+\bar{a}\ \bar{b}\ c+a\ b\ c)=\sum(1,3,4,5,6,7)$$

$$F_{7bp}=\frac{\bar{\bar{\partial}}F_7(a,b,c)}{\bar{\bar{\partial}}b}=\sum(0,2)$$

$$\frac{\partial F_7(a,b,c)}{\partial c}=(a\ b\ \bar{c}+\bar{a}\ b\ c+a\ \bar{b}\ c)\oplus(a\ b\ c+\bar{a}\ b\ \bar{c}+a\ \bar{b}\ \bar{c})=\sum(2,3,4,5,6,7)$$

$$F_{7cp} = \frac{\hat{\partial} F_7(a,b,c)}{\hat{\partial} c} = \sum(0,1)$$

$$LM_{F_7}(a,b,c) = \left| \begin{array}{l} \frac{\left|\sum(3,5,6) \cap \sum(0,4)\right|}{6}, \frac{\left|\sum(3,5,6) \cap \sum(0,2)\right|}{6}, \\ \frac{\left|\sum(3,5,6) \cap \sum(0,1)\right|}{6} \end{array} \right|$$

$$LM_{F_7}(a,b,c) = (0,0,0)$$

$$SC_{F_7} = \frac{3}{\sqrt{3 \times (0)^2 + 1}} = 3$$

METHOD

The research population includes 116 first year students studying for a Bachelor of Electrical Engineering degree at the college. All the students took the Digital Design course in the same study group and with the same lecturer in the second semester of year one. All the students finished the course successfully after the first exam with an average of 81 percent. At the end of the second semester of year one, the first and second stages of the study were conducted. Those same students took a Digital Electronic Circuits course in the first semester of their second year. The students completed the course successfully with an average of 83 percent. The Digital Systems and Digital Electronics courses were taught by different lecturers. All the students also took the Digital Electronics course with the same lecturer. A two-part questionnaire was developed for the fault problems.

RESULTS AND CONCLUSION

Since there are families of Boolean functions with specific properties, not all the Boolean functions can be addressed to the same degree. According to the study hypotheses, the structural complexity of a Boolean function alone as a measure of complexity in content tasks does not fully address learning difficulties. In addition to a logical complexity measure of a Boolean function, also the properties of Boolean functions and the type of problem must be considered to characterize difficulties in learning and problem solving processes in digital systems.

The research hypothesis was that Boolean functions with specific properties can reduce the cognitive complexity in learning and problem solving processes. The estimates chosen in the study to examine the research hypothesis are success in solving short or disconnection type fault identification problems for functions with symmetric properties or for functions with both properties – symmetry and monotone together, compared to the success in solving problems for functions without properties.

Learning processes are at the center of cognition science. The theories take into account relative difficulties of learning various concepts. There are quantitative complexity measures to measure the logical complexity of Boolean functions. These complexity measures are determined according to the degree of the solution's succinctness, represented by the described Boolean function, using basic rules of Boolean algebra. However, logical complexity of Boolean functions is insufficient in a broader perspective in assessing the complexity of problems in digital systems.

Every arbitrary function can be realized using a system of universal functions. Properties of functions enable us to examine whether the collection of Boolean functions can be realized as a digital system or not. They are also criteria for the optimization of digital systems.

The functions F_4, F_{12}, F_{13} have two simultaneous properties – symmetry and monotone. The F_7 function is symmetrical only. According to the study findings, the subjects succeeded to a high degree in solving fault type problems for the functions F_4, F_7, F_{12}, and F_{13} compared to functions without properties with the same or lower logical complexity measure. The four functions and the success rates in solving three types of problems for these functions are summarized in Table 1. The structural complexity measure does not predict difficulty in solving fault problems for functions with the properties examined in the study.

Although the structural complexity measure indicates that the functions F_{12}, F_{13} are the most difficult functions to learn, the success rates for these functions in solving short or disconnection type problems was high.

Technologically, symmetric functions are the most complex with regards to the number of logical elements required for realizing the functions for application. In contrast, cognitively, they are the simplest to give verbal explanations to detect rules. The study findings reinforce this claim for function F_7, which is a symmetric function. The success rates for this function in solving short or disconnection type fault problems were high.

In contrast, functions with two simultaneous properties – symmetry and monotone – is a small plurality of all the Boolean functions and prevalent in practical applications in digital systems.

Boolean functions with both properties – symmetry and monotone – are easier to understand and make it easier to detect rules in learning and problem solving processes.

This manifests in the success rates in this study for the functions F_4, F_{12}, F_{13}, which are simultaneously symmetrical and monotonous. For Boolean functions with the symmetry and monotone property, a different mechanism works from the Boolean complexity of the function, which impacts the problem solving success. Detecting the rules for solving the problem is intuitive for functions with these properties, and explicit knowledge gives way for personal intuition. When the function has two properties – symmetry and monotone – they are simple to solve and comprehend. In reconstruction and fault identification problems, the solutions were based more on formal knowledge than explicit information that was learned in the Digital Systems course.

The functions F_{10}, F_{11} are both linear and symmetric functions. The success rates in solving short or disconnection type fault problems are the lowest compared to the other functions in the study. Compared to symmetric and monotone functions, linear functions that are also necessarily symmetrical for three and more variables are the most difficult to learn and solve.

The difficulties in learning processes for functions with properties can be explained by the logical structure of the functions in their graphic representation. The structure of symmetrical functions is a layer of vertices (numbers). All the combinations are in one layer and the transition between the combinations is continuous (Fig 2). This property makes it easy to detect the rules in the various tasks. The structure of functions that are both symmetrical and monotone is a collection of continuous layers (Fig 2) and therefore they are more intuitive and learned more quickly. Since of the structure of symmetric and monotone functions, the rule structure is revealed intuitively, the tacit knowledge overcomes the explicit knowledge. In contrast, in linear functions that are a collection of layers with a separating layer that breaks the continuity in the transition between the layers (Fig 2), the non-continuity between the layers makes it difficult to detect the rules intuitively and subsequent to this difficulty the subjects turn to analytical tools in solving the various tasks. Linear functions are an individual case of symmetric functions.

For each of the functions F_4, F_7, F_{12}, F_{13} in solving the three types of problems, the interviewees described the problem solving path as intuitive.

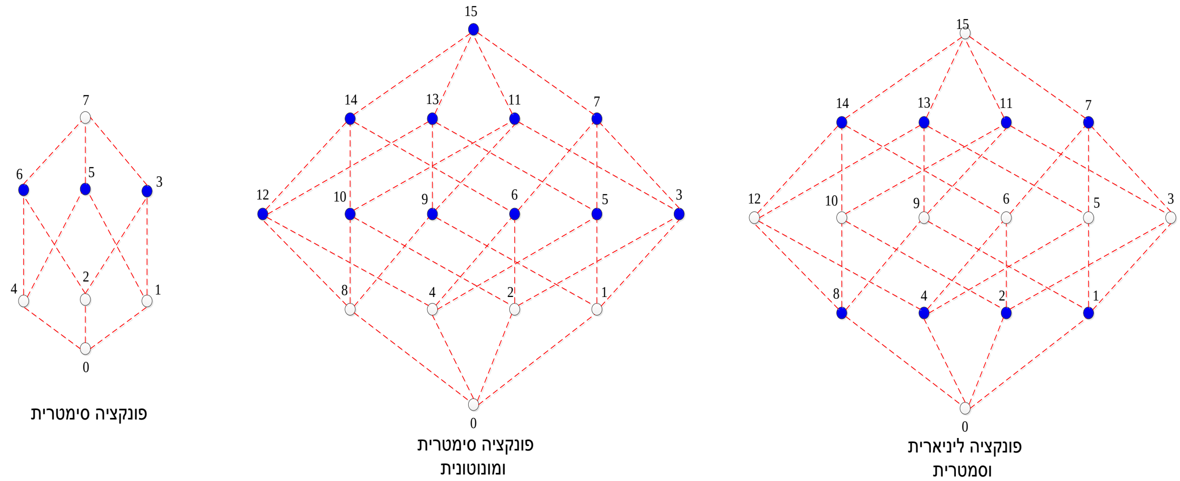

Fig 2: Graphic depiction of functions with the properties: linearity and symmetry, symmetry and monotone, and symmetry

In a broader perspective and in light of the study's conclusions to characterize the cognitive complexity of digital systems in addition to the logical complexity of the function, the type of problem and the properties of the functions are also important. In the processes of learning Boolean concepts and solving problems in digital systems as part of the Digital Systems course, the three components must be taken into account – structural complexity measure as Boolean complexity of the function, complexity of the type of problem, and properties of the function if there are any, according to the depiction in Fig 3 below.

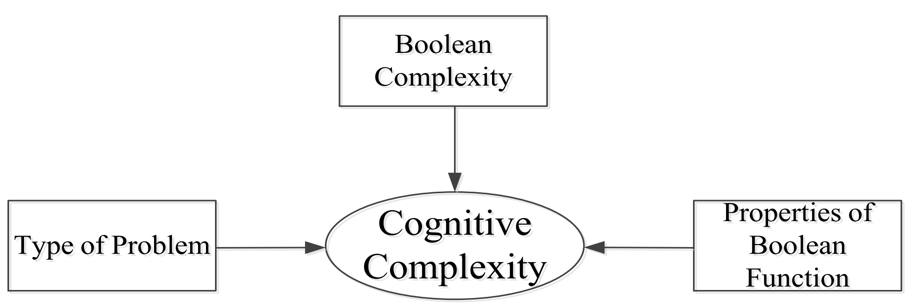

Fig 3. Cognitive complexity characterizes.

By considering the three components, it is possible to characterize the cognitive complexity measure in the process of learning Boolean concepts that can be described by Boolean functions and in learning processes through problem solving.

REFERENCES

[1] Minsky, M.L. (1967). *Computation: Finite and Infinite Machines*. Englewood-Cliffs, NJ: Prentice-Hall.

[2] Kolmogorov, A. N. (1965). Three approaches to the quantitative definition of information'. *Problems of information transmission, 1*(1), 1-7.

[3] Li, M. & Vitanyi, P. An Introduction to Kolmogorov Complexity and its Applications (Springer, New York, 1997).

[4] Shepard, R. N., Hovland, C. I., & Jenkins, H. M. (1961). Learning and memorization of classifications. Psychological Monographs: General and Applied 75(13), 1-42.

[5] Reed, I. S. (1954). A class of multiple error-correcting codes and the decoding scheme. IRE Trans. on Information Theory, 4, 38-49.

[6] Vigo, R., 2009. Categorical invariance and structural complexity in human concept learning. Journal of Mathematical Psychology, 53, 203–221.

[7] Post, E. L. (1941). The two-valued iterative systems of mathematical logic, annals of mathematics series #5. Princeton, NJ: Princeton University Press.

[8] Wegener, I. (1987). The complexity of Boolean functions. Chichester: Wiley.

[9] Parberry, I. (1994). Circuit complexity and neural networks. MIT Press.

[10] Siu, K. Y., Roychowdhury, V. P., & Kailath, T. (1995). Discrete Neural Computation - A Theoretical Foundation. Prentice Hall.

[11] Korshunov, A.D. (2003). Monotone Boolean functions. Russian Mathematical Surveys 58.

[12] Brooke, J. B., & Duncan, K. D. (1981). Effects of system display format on performance in a fault location task. Ergonomics, 24, 175-189.

[13] Kostopolou, O., & Duncan, K. D. (2001). Abstract and reduced-context representations in Fault-finding training. Ergonomics, 44, 175-201.

[14] Morrison, D. L. & Duncan, K. D. (1988). Strategies and tactics in fault diagnosis. Ergonomics, 31, 761-784.

[15] Leake, D. B. (1995). Abduction, experience, and goals: A model of everyday abductive explanation. Journal of Experimental and Theoretical Artificial Intelligence, 7, 407-428.

[16] Goodwin, G. P. (2006). How individuals learn simple Boolean systems and diagnose their faults (Doctoral dissertation, Princeton University, 2006). Dissertation Abstracts International: Section B: The Sciences and Engineering, 67, 1176.