Fundamenta Informaticae 104 (2010) 1–16
DOI 10.3233/FI-2010-330
IOS Press

A Communication Approach to the Superposition Problem

Farid Ablayev*†
Departmnt of Theoretical Cybernetics,
Kazan State University, Kazan, Russia
fablayev@gmail.com

Svetlana Ablayeva
Department of Differential Equations,
Kazan State University, Kazan, Russia

Abstract. In function theory the superposition problem is known as the problem of representing a continuous function $f(x_1, \ldots, x_k)$ in k variables as the composition of "simpler" functions. This problem stems from the Hilbert's thirteenth problem. In computer science good formalization for the notion of composition of functions is formula.

In the paper we consider real-valued continuous functions in k variables in the cube $[0,1]^k$ from the class $\mathcal{H}^k_{\omega_p}$ with ω_p a special *modulus of continuity* (measure the smoothness of a function) defined in the paper. $\mathcal{H}^k_{\omega_p}$ is a superset of Hölder class of functions. We present an explicit function $f \in \mathcal{H}^k_{\omega_p}$ which is hard in the sense that it cannot be represented in the following way as a formula: zero level (input) gates associated with variables $\{x_1, \ldots, x_k\}$ (different input gates can be associated with the same variable $x_i \in \{x_1, \ldots, x_k\}$), on the first level of the formula, arbitrary number $s \geq 1$ of t variable functions from $\mathcal{H}^t_{\omega_p}$ for $t < k$ are allowed, while the second (output) level may compute any s variable Hölder function.

We apply communication complexity for constructing such hard explicit function. Notice that one can show the existence of such function using the "non constructive" proof method known in function theory as Kolmogorov's entropy method.

Keywords: Hilbert 13th problem, superposition of continuous functions, communication complexity

*Supported by SNF grant 200021-107327/1 and RFBR grant 09-01-97004

†Address for correspondence: Department of Theoretical Cybernetics, Kazan State University, Kremlevskaia str. 18, Kazan, 42008, Russia

1. Preliminaries

In classic mathematic the problem of representation of functions by functions of "simpler" (in some sense) quality has a long history and is based on the following problem. It is known that a common equation $a_1x^n + a_2x^{n-1} + \cdots + a_nx + a_{n+1} = 0$ for $n \leq 4$ can be solved over radicals. In terms of the superposition problem this means that the roots of the equation can be represented by a superposition of arithmetic operations and one variable function of the form $\sqrt[n]{a}$ $(n = 2,\ 3)$ of coefficients of the equation. Galois and Abel proved that a common equation of the 5-th order can not be solved in radicals (can not be represented as a superposition of this special form). Hilbert [9] presented his 13-th problem as a problem of *representing a solution of a common equation of the 7-th order as a superposition of functions of two variables*. Hilbert's 13-th problem motivated an investigation of different aspects of the superposition problem. See [13, 15] for more information on the subject.

Arnold [3] and Kolmogorov [12] proved that any multivariate continuous real-valued function can be represented as a two-level superposition of continuous functions of only one variable and sum operation. This result is known as *Kolmogorov's superposition theorem*. More formally: denote $C([0,1]^k)$ a space of continuous real-valued functions in cube $[0,1]^k$. The superposition theorem states that for every function $f \in C([0,1]^k)$ there are $2k+1$ continuous functions ϕ_i, $i \in \{1,\dots,2k+1\}$, such that

$$f(x_1,\dots,x_k) = \sum_{i=1}^{2k+1} \phi_i \left(\sum_{j=1}^{k} h_{ij}(x_j) \right), \qquad (1)$$

where the functions $h_{ij} \in C([0,1])$ are universal for the given dimension k, i.e., are independent of the function f, while only the functions ϕ_i depend on f.

A generalization of Kolmogorov's superposition theorem presented in several papers. As it is proved in [15], the functions h_{ij} belong to the Hölder class.

Although Kolmogorov's superposition theorem was intended for approximation of real functions, it is might be used for implementing Boolean functions. Such approach relies on analog circuity. With present advances of nanoelectronics, it is to be expected that hybrid CMOS analog/digital implementations appear to be efficient for information processing. In according to the relation (1) it is appeared that a use of Kolmogorov-inspired gates (i.e. analog gates) for circuits constructions might provide considerable saving in complexity for functions realization. For more information and discussions see [5] and survey [4]. The question on how cheap it would be possible to realize Kolmogorov's gates might be an interesting theoretical and practical problem.

2. Introduction

Generally the superposition problem can be expressed as follows: can continuous function f from certain class be expressed by a set of "simpler" class of functions. In terms of complexity theory the superposition problem is a problem of presenting f by a formula (by a circuit with 1 fan-out gates). It is generally not easy task to construct a hard functions explicitly — a functions that cannot be written as a superposition of a "simpler" functions.

Vitushkin and Kolmogorov proved that in certain classes of continuous functions hard functions exist and raised the question of constructing such functions explicitly. More precisely. Let $\mathcal{F}_p^k$ denote the class of all continuous functions of k variables which have restricted continuous partial derivatives up to the p-th order. Vitushkin (see a survey [18]) proved that there exists a function from $\mathcal{F}_p^k$ which cannot be represented by a superposition of functions from $\mathcal{F}_q^t$ if $\frac{k}{p} > \frac{t}{q}$. Later Kolmogorov gave a proof of this fact based on comparing complexity characteristics (entropy of discrete approximation of functional spaces) of classes $\mathcal{F}_p^k$ and $\mathcal{F}_q^t$. See the survey [18] and [15] for more information on the subject. In [13] Kolmogorov presented a problem of constructing an explicit continuous function that is hard to approximate in the sense of computational complexity of its approximation.

Since then Marchenkov [16] was only successful in defining a hard function f from $\mathcal{F}_p^k$ (function that cannot be represented by a superposition of functions from $\mathcal{F}_q^t$ if $\frac{k}{p} > \frac{t}{q}$) by using exponentially hard Boolean function.

In the recent paper [7] Hansen, Lachish, and Miltersen study a discrete analogue of Hilbert's 13th problem: Can all expicit (e.g. polynomial time computable) functions $f : (\{0,1\}^\omega)^3 \to \{0,1\}^\omega$ be computed by word circuits of constant size?

In the paper we consider real-valued continuous functions in k variables in the cube $[0,1]^k$ from the class $\mathcal{H}_{\omega_p}^k$ with ω_p a special modulus of continuity defined in (2). $\mathcal{H}_{\omega_p}^k$ is a superset of Hölder class of functions. Class $\mathcal{H}_{\omega_p}^k$ is itself subclass of the class $\mathcal{D}^k$ of functions known as Dini class. We present an explicit function $f \in \mathcal{H}_{\omega_p}^k$ which is hard in the sense that it cannot be represented in the following way as a superposition (as a formula): zero level (input) gates associated with variables $\{x_1, \ldots, x_k\}$ (different input gates can be associated with the same variable $x_i \in \{x_1, \ldots, x_k\}$), on the first level of the formula, arbitrary number $s \geq 1$ of t variable functions from $\mathcal{H}_{\omega_p}^t$ for $t < k$ are allowed, while the second (output) level may compute any s variable Hölder function.

We apply communication complexity for constructing such hard explicit function. Notice that one can show the existence of such function using the "non constructive" proof method known in function theory as Kolmogorov's entropy method. The initial version of this paper was presented in ECCC [1], the conference version of this result was presented in [2].

The paper is organized as follows. First we define continuous functions classes and present known hierarchy of classes of continuous functions based on the behavior of their modulus of continuity ω. Then using a sequence g of certain Pointer Boolean function from the AC^0 with the linear one-way communication complexity we define an explicit continuous function $f_{\omega,g} \in C([0,1]^k)$ from $\mathcal{H}_{\omega_p}^k$. The function $f_{\omega,g}$ is hard in according above discussion.

The proof method of the fact that $f_{\omega,g}$ does not presented by a superposition of the certain form use a discrete approximation of continuous functions and the communication complexity technique and is the following. We suppose that $f_{\omega,g}$ is presented by a superposition S of continuous functions. We consider their proper discrete approximations df and DS and compare the communication complexity C_{df} and C_{DS} of discrete functions df and DS respectively. By showing $C_{DS} < C_{df}$ we prove that $f_{\omega,g}$ cannot be presented by the superposition S.

Communication computations models are models of distributive computations was introduced by Yao [19]. A good source of information on communication computations are books [10, 14]. Communication complexity methods works productively in different areas of computer science and mathematics. Recently algebraic communication protocols (deterministic [6] and probabilistic [8]) were defined for certain algebraic problems. In papers [6, 8] presented several lower bounds on the algebraic communication complexity for computing real functions and recognizing algebraic varieties.

Recently in [11] Jukna introduced a complexity measure $entropy(f)$ entropy of an operator $f : \{0,1\}^n \to \{0,1\}^m$ and proved an interesting result that every depth-2 circuit for f requires at least $entropy(f)$ wires. Notice that the notion of entropy of f can be reformulated in terms of one-way communication complexity of f.

3. Functions Classes, Superpositions, and Function $f_{\omega,g}$

In this section we present definitions and known facts we need from function theory according to book [17].

3.1. Functions Classes Hierarchy

Following function theory for continuous function $f(x_1, \ldots, x_k) \in C([0,1]^k)$ its *modulus of continuity* $\omega_f(\delta)$ is the smallest upper bound of $|f(x) - f(x')|$, for all $x, x' \in [0,1]^k$ such that $|x - x'| = \max_{1 \le i \le k} |x_i - x'_i| \le \delta$. Modulus of continuity is a precise way to measure the smoothness of a function. The following condition is sufficient for a function to be a modulus of continuity (see [17], section 3.2).

Property 1. A continuous function $\omega(\delta)$ is a modulus of continuity if $\frac{\omega(\delta)}{\delta}$ is monotonously non-increasing in δ.

For a modulus of continuity ω the class $\mathcal{H}^k_\omega \subseteq C([0,1]^k)$ is defined as follows.

$$\mathcal{H}^k_\omega = \{f : \omega_f(\delta) = O(\omega(\delta))\}.$$

The following classes are known as Hölder classes in functions theory.

$$\mathcal{H}^k_\gamma = \{f : \omega_f(\delta) = O(\delta^\gamma)\}, \qquad (\gamma > 0).$$

The Hölder class $\mathcal{H}^k_1$ is also known as the Lipschitz class. Denote $\mathcal{H}_\gamma = \bigcup_{k \ge 1} \mathcal{H}^k_\gamma$ and $\mathcal{C} = \bigcup_{k \ge 1} C([0,1]^k)$. The following properties are known facts in functions theory.

- The class $\mathcal{F} \subset \mathcal{C}$ of continuous functions which have continuous derivatives is a proper subclass of $\mathcal{H}_1$.
- Hölder classes form proper hierarchy. For $\gamma < \gamma'$ it holds that $\mathcal{H}_{\gamma'} \subset \mathcal{H}_\gamma$.
- Class $\mathcal{H}_\gamma$ — is a class of constant functions if $\gamma > 1$.
- For a modulus of continuity $\omega(\delta)$ with $\lim_{\delta \to 0} \omega(\delta) \log \frac{1}{\delta} = 0$ or (using o-notation) with $\omega(\delta) = o\left(1/\log \frac{1}{\delta}\right)$ class $\mathcal{H}^k_\omega$ is known as Dini class and denoted $\mathcal{D}^k$. Dini class contains all Hölder classes properly.

Let $p > 1$, $a = 1/(e^{p+1})$, and

$$\omega_p(\delta) = \begin{cases} 1/(\ln 1/\delta)^p & \text{if } 0 < \delta \le a \\ 1/(\ln 1/a)^p & \text{if } \delta > a, \end{cases} \tag{2}$$

From the definitions it holds that class $\mathcal{H}^k_{\omega_p}$ is a subclass of Dini class $\mathcal{D}^k$ and is superset of Hölder classes.

3.2. Superpositions

Let Ω be a set of continuous functions. We call Ω a *basis*. We define a representation of a function $f \in C([0,1]^k)$ by superposition of functions from Ω in terms of circuits.

Circuit. A *circuit* over a set $\{x_1, \ldots, x_k\}$ of k variables, over a basis Ω is a dag with gates (nodes) either corresponding to functions in Ω or having in-degree 0 and called input gates (nodes). With the input gates we associate variables from $\{x_1, \ldots, x_k\}$. We allow different input gates to be associated with the same variable $x_i \in \{x_1, \ldots, x_k\}$.

The value at a gate is computed by applying the corresponding function to the values of the preceding gates. One node is distinguished as *output*.

Superposition. A *formula* is a circuit with all out-degrees ≤ 1, that is, with a tree structure. We denote C_Ω^k a formula over a set $\{x_1, \ldots, x_k\}$ of k real-valued variables, over a basis Ω — a set of real-valued continuous (discrete) functions. In the paper we consider only formulas. We call C_Ω^k a *superposition*.

We will also view on C_Ω^k as a function computed by this circuit. In this case we call C_Ω^k a (k, Ω)-superposition, or just a superposition.

Superposition $Sp^k[\mathcal{H}_{\omega_1}^t, \mathcal{H}_{\omega_2}^s]$. For modulus of continuity $\omega_1(\delta)$ and $\omega_2(\delta)$ denote $Sp^k[\mathcal{H}_{\omega_1}^t, \mathcal{H}_{\omega_2}^s]$ a set of functions from $C([0,1]^k)$, that can be represented by a superposition of the following form

$$F\left(h_1(x_1^1, \ldots, x_t^1), \ldots, h_s(x_1^s, \ldots, x_t^s)\right), \tag{3}$$

where F is a function from class $\mathcal{H}_{\omega_2}^s$, and $\{h_i : 1 \leq i \leq s\} \subseteq \mathcal{H}_{\omega_1}^t$.

Informally (3) is the following two-level formula over a set $\{x_1, \ldots, x_k\}$ of k variables, over the basis $\Omega = \{\mathcal{H}_{\omega_1}^t, \mathcal{H}_{\omega_2}^s\}$. First level of (3) contain functions from $\mathcal{H}_{\omega_1}^t$ while the second level contain function from $\mathcal{H}_{\omega_2}^s$. In general it might be $s + t > k$ since we allow different input gates to be associated with the same variable $x_i \in \{x_1, \ldots, x_k\}$.

Property 2. For modulus of continuity $\omega_1(\delta)$ and $\omega_2(\delta)$ it holds that the function $\omega(\delta) = \omega_2(\omega_1(\delta))$ is a modulus of continuity and $Sp^k[\mathcal{H}_{\omega_1}^t, \mathcal{H}_{\omega_2}^s] \subseteq \mathcal{H}_\omega^k$. In particular we have that $Sp^k[\mathcal{H}_{\omega_1}^t, \mathcal{H}_1] \subseteq \mathcal{H}_{\omega_1}^k$.

3.3. The function $f_{\omega,g}$

We define an explicit continuous function $f_{\omega,g} \in \mathcal{H}_\omega^k$ by a sequence $g = \{g_n\}$ of explicit Boolean functions (known as Pointer functions). We consider $n = 2^j - 1,\ j \geq 1$ through the paper. Informally speaking our construction of $f_{\omega,g}$ can be described as follows. We fix a subset I^k of cube $[0,1]^k$. We define $f_{\omega,g}(x)$ for $x \in I^k$ as follows. We start with the informal description.

Cubes I_n^k and a set I^k. Let $I_n = [\frac{1}{n+1}, \frac{2}{n+1}]$ be a closed interval and let $I_n^k = \underbrace{I_n \times \cdots \times I_n}_{k}$. Clearly we have that $I_n^k \cap I_{n'}^k = \emptyset$ for $n \neq n'$. We define

$$I^k = \bigcup_{n \geq 1} I_n^k.$$

1. We partition each cube I_n^k into 2^{kn} cubes of the same size. We encode each center b of the 2^{kn} cubes by a binary sequence v of length kn. We write $b = b(v)$ and denote $I_n^k(b(v))$ a corresponding sub cube of I_n^k. So we have
$$I_n^k = \bigcup_{v\in\{0,1\}^{kn}} I_n^k(b(v)).$$
2. We define simple continuous function $\Psi^k_{n,b(v)}(x)$ in each cube $I_n^k(b(v))$, $(n \geq 1, v \in \{0,1\}^{kn})$.
3. We associate with each cube $I_n^k(b(v))$ Pointer Boolean function $g_n : \{0,1\}^{kn} \to \{0,1\}$ over kn variables.
4. Our function $f_{\omega,g}$ is determined in each cube $I_n^k(b(v))$ by function g_n and continuous functions $\Psi^k_{n,b(v)}(x)$. Boolean value $g_n(v)$ determines the behavior of function $\Psi^k_{n,b(v)}(x)$ in each cube $I_n^k(b(v))$.

Now we turn to the formal definitions.

Partition of I_n^k. Let $\Sigma = \{0,1\}$. We consider the following mapping $a : \Sigma^n \to [0,1]$. For a word $v = \sigma_1 \ldots \sigma_n \in \Sigma^n$ we define real number
$$a(v) = \frac{1}{n+1}\left(1 + \sum_{i=1}^{n} \sigma_i 2^{-i} + \frac{1}{2^{n+1}}\right).$$
Let $A_n = \{a(v) : v \in \Sigma^n\}$. Let $\Delta_n = \frac{1}{2(n+1)2^n}$. For a number $a(v) \in A_n$ denote
$$I_n(a(v)) = [a(v) - \Delta_n,\ a(v) + \Delta_n]$$
the closed interval of real numbers of length $2\Delta_n = \frac{1}{(n+1)2^n}$ and a center $a(v)$. From the definitions of A_n and $I_n(a(v))$ it holds that:

1. For $a(v),\ a(v') \in A_n$ and $a(v) \neq a(v')$ the segments $I_n(a(v))$ and $I_n(a(v'))$ can intersect only in their boundary.
2. $\bigcup_{a(v)\in A_n} I_n(a(v)) = I_n$

For a tuple $v = (v_1, \ldots, v_k)$, where $v_i \in \Sigma^n$, $1 \leq i \leq k$, denote
$$I_n^k(b(v)) = I_n(a(v_1)) \times \cdots \times I_n(a(v_k))$$
the k-dimensional cube of size $2\Delta_n$ with a center in a point $b(v) = (a(v_1), \ldots, a(v_k))$. From the definition of $I_n^k(b(v))$ we have the following property.

Property 3. It is true that
$$I_n^k = \bigcup_{v\in\Sigma^{kn}} I_n^k(b(v)) \qquad \text{and} \qquad I_n^k(b(v)) \cap I_n^k(b(v')) = \emptyset \quad \text{for} \quad v \neq v'$$

Now we define continuous function Ψ and Pointer Boolean function g.

Functions Ψ. We define a one variable continuous function $\Psi^1_{n,a(v)}(x)$ on the segment $I_n(a(v))$, $a(v) \in A_n$ as follows:

$$\Psi^1_{n,a(v)}(x) = \begin{cases} 1 + \frac{x-a(v)}{\Delta_n}, & a(v) - \Delta_n \le x \le a(v) \\ 1 - \frac{x-a(v)}{\Delta_n}, & a(v) \le x \le a(v) + \Delta_n \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

From the definition we have the following property.

Property 4. For the function $\Psi^1_{n,a(v)}(x)$ the following is true:

- $\Psi^1_{n,a(v)}(x)$ reaches the maximum value 1 in the center $a(v)$ of the segment $I_n(a(v))$ and has value 0 in the border points of this segment.
- For $x, x' \in I_n(a(v))$, for $\delta = |x - x'|$ Modulus of continuity $\omega_{\Psi^1_n}(\delta)$ of $\Psi^1_{n,a(v)}(x)$ satisfies

$$\omega_{\Psi^1_n}(\delta) = \begin{cases} \frac{\delta}{\Delta_n}, & \text{if } \; 0 < \delta \le \Delta_n \\ 1, & \text{if } \; \Delta_n \le \delta \le 2\Delta_n. \end{cases} \tag{5}$$

Consider the following continuous k variable function $\Psi^k_{n,b(v)}(x)$ inside each cube $I^k_n(b(v))$

$$\Psi^k_{n,b(v)}(x) = \prod_{i=1}^{k} \Psi^1_{n,a(v_i)}(x_i).$$

Property 5. Function $\Psi^1_{n,b(v)}(x)$ reaches the maximum value 1 in the center $b(v)$ of the cube $I^k_n(b(v))$ and has value 0 in the border points of this cube.

Sequence g of Pointer Boolean functions. Let $g = \{g_n(v)\}$ is the sequence of the following Pointer Boolean functions:

$$g_n : \underbrace{\Sigma^n \times \cdots \times \Sigma^n}_{k} \to \{0,1\}.$$

For inputs $v = (v_1, \dots, v_k)$, where $v_i \in \Sigma^n$, $1 \le i \le k$, we consider the following partition $pat(n,k)$. Each word v_i of v is divided into two parts the beginning u_i and the end w_i of length $l(n,k) = n - d(n,k)$ and $d(n,k) = \lceil (\log kn)/k \rceil$ respectively. We will write $v = (u, w)$ and call $u = (u_1, \dots, u_k)$ the first part of v and $w = (w_1, \dots, w_k)$ the second part of v.

Function $g_n(u,w) = 1$ iff the $(ord(w_1 \dots w_k) + 1)$-th bit in the word $u_1 \dots u_k$ is one. Here $ord(\sigma)$ denotes the integer whose binary representation is σ. The numeration of bits in the words starts from 1. We will use both notation $g_n(v)$ and $g_n(u,w)$ for the Boolean function g_n.

Notice that the function g_n is in the uniform class AC^0 and formally described by the following DNF formula:

$$g_n(u,w) = \bigvee_{\substack{\sigma \\ 0 \le ord(\sigma) \le |u|-1}} \bigwedge_{i=1}^{kd(n,k)} y_i^{\sigma_i} \wedge x_{ord(\sigma)},$$

where y_j (x_j) is the j-th symbol of the sequence w (u) in the common numeration of its elements.

Function $f_{\omega,g}$. Let $\omega(\delta)$ be a continuous function such that $\lim_{\delta\to 0}\omega(\delta) = 0$. Define continuous k variable function $f_{\omega,g}$ on cube $[0,1]^k$ as follows:

$$f_{\omega,g}(x) = \sum_{\substack{n=2^j-1,\\ j\geq k}} \sum_{v\in\Sigma^n} (2g_n(v)-1)\omega(\Delta_n)\Psi^k_{n,b(v)}(x). \tag{6}$$

Function $f_{\omega,g}$ has the following property.

Property 6. For function $f_{\omega,g}$ the following holds:

1. $f_{\omega,g}$ is continuous function on cube $[0,1]^k$.
2. In each cube $I^k_n(b(v)) \in I^k$ function $f_{\omega,g}$ reaches the maximum (minimum) value $\omega(\Delta_n)$ $(-\omega(\Delta_n))$ in the center $b(v)$ and has value 0 in the border points of the cube.

4. The Result

The result of the paper (for the special modulus of continuity that determines the subclass $\mathcal{H}^k_{\omega_p}$ of Dini class) is presented in Theorem 4.2. We state and prove slightly more general Theorem 4.1.

Theorem 4.1. Let $t < k$. Let $\omega_1(\delta)$ be a monotonously non-decreasing in δ function such that $\frac{\omega_1(\delta)}{\delta}$ is monotonously non-increasing in δ and

$$\log\frac{1}{\omega_1(\delta)} = o\left(\left(\log\frac{1}{\delta}\right)^{1-t/k}\right). \tag{7}$$

Then for $s \geq 1$ and $\gamma \in (0,1]$, for $\omega_2(\delta) = O(\delta^\gamma)$, $\omega(\delta) = \omega_2(\omega_1(\delta))$ function $f_{\omega,g}(x)$ is in $\mathcal{H}^k_\omega \backslash Sp^k[\mathcal{H}^t_{\omega_1}, \mathcal{H}^s_{\omega_2}]$. That is, $f_{\omega,g}(x)$ cannot be presented as the superposition

$$F\left(h_1(x^1_1,\ldots,x^1_t),\ldots,h_s(x^s_1,\ldots,x^s_t)\right),$$

where $F \in \mathcal{H}^s_{\omega_2}$, and $\{h_i : 1 \leq i \leq s\} \subseteq \mathcal{H}^t_{\omega_1}$.

The proof of Theorem 4.1 we present in the next section.

Notice. Noitice that the condition of Theorem 4.1 demands only the relation $t < k$. Number s of variables of function F does not depends on t and k and might be arbitrarily large. Each variable $x_j \in \{x_1,\ldots,x_k\}$ of the function $f_{\omega,g}(x)$ "can be used" by different functions $\{h_i : 1 \leq i \leq s\}$.

Theorem 4.2. Let $t < k$. For $p > 1$ let ω_p be a modulus of continuity defined in (2). Then function $f_{\omega_p,g}(x)$ over k variables is in the class $\mathcal{H}^k_{\omega_p}$ and cannot be represented by a superposition $Sp^k[\mathcal{H}^t_{\omega_p}, \mathcal{H}_\gamma]$ for arbitrary $s \geq 1$ and $\gamma \in (0,1]$. That is,

$$f_{\omega_p,g}(x) \in \mathcal{H}^k_{\omega_p} \backslash Sp^k[\mathcal{H}^t_{\omega_p}, \mathcal{H}_\gamma].$$

Proof:
Theorem is a particular case of Theorem 4.1. Modulus of continuity ω_p satisfies condition (7) of Theorem 4.1. According to the notation ω_2 in Theorem 4.1 class $\mathcal{H}^s_{\omega_2}$ is the Hölder class ($\mathcal{H}^s_{\omega_2} = \mathcal{H}^s_\gamma$). □

5. Proof of Theorem 4.1

From Property 2 we have that $Sp^k[\mathcal{H}^t_{\omega_1}, \mathcal{H}^s_{\omega_2}] \subseteq \mathcal{H}^k_\omega$. The next statement proves that our function $f_{\omega,g}$ is in $\mathcal{H}^k_\omega$.

Property 7. Let $\omega(\delta)$ be an increasing in δ function such that $\frac{\omega(\delta)}{\delta}$ is monotonously non-increasing in δ. Then $f_{\omega,g} \in \mathcal{H}^k_\omega$.

Proof:
For the simplicity we denote f our function $f_{\omega,g}$ in this proof. From Property 1 we have that function $\omega(\delta)$ is a modulus of continuity. Denote $\omega_f(\delta)$ a modulus of continuity of f. To prove inclusion $f \in \mathcal{H}^k_\omega$ we will show that

$$\omega_f(\delta) \leq 2k\omega(\delta). \tag{8}$$

For our function $f(x_1, x_2, \ldots, x_k)$, for $i \in \{1, 2, \ldots, k\}$ and $z_i = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k)$ denote $f_{z_i}(x_i) = f(x)$ the subfunction of f. Function

$$\omega_i(\delta) = \sup_{z_i \in I^{k-1}} \sup_{|x_i - x'_i| \leq \delta} |f_{z_i}(x_i) - f_{z_i}(x'_i)| \tag{9}$$

is called a partial modulus of continuity of function f [17]. It is known from function theory that for modulus of continuity and partial modulus of continuity the following is true (see [17], 3.4.31):

$$\max_{1 \leq i \leq k} \{\omega_i(\delta)\} \leq \omega_f(\delta) \leq \sum_{i=1}^{k} \omega_i(\delta).$$

From the above inequality we have that in order to prove (8) we have to show that for all $i \in \{1, \ldots, k\}$ it holds that

$$\omega_i(\delta) \leq 2\omega(\delta). \tag{10}$$

Let $i \in \{1, \ldots, k\}$, let $z_i \in I^{k-1}_m(b(w^{(i)}))$ where $w^{(i)} = (v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_k) \in \Sigma^m \times \cdots \times \Sigma^m$. From the definition of f we get that its subfunction f_{z_i} is defined on I_m. Let points $x_i, x'_i \in I_m$ and words $v_i, v'_i \in \Sigma^m$ are such that $x_i \in I_m(a(v_i))$ and $x'_i \in I_m(a(v'_i))$. By straightforward calculation we have that

$$\begin{aligned} |f_{z_i}(x_i) - f_{z_i}(x'_i)| = \ & \omega(\Delta_m) \prod_{j=1, j\neq i}^{k} \Psi^1_{m,a(v_j)}(x_j) |(2g_m(v) - 1)\Psi^1_{m,a(v_i)}(x_i) - \\ & (2g_m(v') - 1)\Psi^1_{m,a(v'_i)}(x'_i)|, \end{aligned} \tag{11}$$

here $v = (v_1, \ldots, v_{i-1}, v_i, v_{i+1}, \ldots, v_k)$, and $v' = (v_1, \ldots, v_{i-1}, v'_i, v_{i+1}, \ldots, v_k)$. Let $\delta = |x_i - x'_i|$.

If $g_m(v) = g_m(v') = \gamma$ then, using (11) and Property 4, we have that

$$\begin{aligned} |f_{z_i}(x_i) - f_{z_i}(x'_i)| = \ & \omega(\Delta_m)|(2\gamma - 1)| \prod_{j=1, j\neq i}^{k} \Psi^1_{m,a(v_j)}(x_j) |\Psi^1_{m,a(v_i)}(x_i) - \Psi^1_{m,a(v'_i)}(x'_i)| \\ \leq \ & \omega(\Delta_m)|\Psi^1_{m,a(v_i)}(x_i) - \Psi^1_{m,a(v'_i)}(x'_i)| \leq \omega(\Delta_m)\omega_{\Psi^1_m}(\delta). \end{aligned}$$

If $g_m(v) \neq g_m(v')$ then, using (11) and Property 4, we have that

$$|f_{z_i}(x_i) - f_{z_i}(x_i')| \leq 2\omega(\Delta_m)\omega_{\Psi_m^1}(\delta).$$

From relation (9), last two inequalities and Property 4 we have that on segment I_m the partial modulus of continuity $\omega_i(\delta)$ of the f satisfy

$$\omega_i(\delta) = \begin{cases} 2\omega(\Delta_m)\frac{\delta}{\Delta_m}, & \text{if } \ 0 < \delta \leq \Delta_m \\ 2\omega(\Delta_m), & \text{if } \ \Delta_m \leq \delta \leq 2\Delta_m. \end{cases} \tag{12}$$

Now we estimate behavior of $\omega_i(\delta)$ on I_m. We consider two cases:

1. Case $\delta \in [\Delta_m, 2\Delta_m]$. From (12) we have that $\omega_i(\delta)$ is constant on segment $[\Delta_m, 2\Delta_m]$:

$$\omega_i(\delta) = 2\omega(\Delta_m).$$

Using the property that function $\omega(\delta)$ increase when δ increase we have that $\omega(\Delta_m) \leq \omega(\delta)$. Combining two last relations get that

$$\omega_i(\delta) \leq 2\omega(\delta)$$

2. Case $0 < \delta < \Delta_m$. From (12) we have that

$$\omega_i(\delta) = 2\omega(\Delta_m)\frac{\delta}{\Delta_m}.$$

Using the property that $\frac{\omega(\delta)}{\delta}$ does not increase when δ increase we have that

$$\frac{\omega(\delta)}{\delta} \geq \frac{\omega(\Delta_m)}{\Delta_m}.$$

Combining two last relations we get that

$$\omega_i(\delta) \leq 2\omega(\delta).$$

Finally combining two cases we have that

$$\omega_i(\delta) \leq 2\omega(\delta). \tag{13}$$

The last relation proves the statement of the property. □

The proof of the second part of the statement ($f_{\omega_p,g}(x) \notin Sp^k[\mathcal{H}^t_{\omega_p}, \mathcal{H}_1]$) of Theorem 4.1 use communication complexity arguments and is based on computing communication complexity of discrete approximations of the function $f_{\omega,g}$.

Definition 1. Let $f(x) \in C([0,1]^k)$. Let $\alpha(n) = \min\{f(x) : x \in I_n^k = [\frac{1}{n}, \frac{2}{n}]^k\}$, and $\beta(n) = \max\{f(x) : x \in I_n^k = [\frac{1}{n}, \frac{2}{n}]^k\}$. We call a discrete function

$$df : \underbrace{\Sigma^n \times \cdots \times \Sigma^n}_{k} \to [\alpha(n), \beta(n)]$$

an $\varepsilon(n)$-approximation of $f(x)$, if for arbitrary $v = (v_1, \ldots, v_k) \in \Sigma^n \times \cdots \times \Sigma^n$ it holds that

$$|f(b(v)) - df(v)| \leq \varepsilon(n).$$

Recall that $b(v)$ is the center of the cube $I_n^k(b(v))$ and according to Property 3 we have that $I_n^k = \bigcup_{v \in \Sigma^{kn}} I_n^k(b(v))$.

We will use the standard one-way communication computation for computing the Boolean function $g_n \in g$. That is, two processors P_u and P_w obtain inputs in accordance with the partition $pat(n,k)$of input v. The first part $u = (u_1, \ldots, u_k)$ of the input sequence v is known to P_u and the second part $w = (w_1, \ldots, w_k)$ of v is known to P_w.

The communication computation of a Boolean function g_n is performed in accordance with a one-way protocol Φ as follows. P_u sends message m (binary word) to P_w. Processor P_w computes and outputs the value $g_n(u, w)$. The communication complexity C_Φ of the communication protocol Φ for the partition $pat(n,k)$ of an input $v = (v_1, \ldots, v_k)$ is the length $|m|$ of the message m. The communication complexity $C_{g_n}(pat(n,k))$ of a Boolean function g_n is defined as follows

$$C_{g_n}(pat(n,k)) = \min\{C_\Phi : \Phi \quad \text{computes} \quad g_n\}.$$

Lemma 1. For the Boolean function $g_n \in g$ it holds that

$$C_{g_n}(pat(n,k)) \geq k(n-1) - \log kn.$$

Proof:
With the function $g_n(u, w)$ we associate a $2^{kl(n,k)} \times 2^{kd(n,k)}$ communication matrix CM_{g_n} whose (u, w) entry is $g_n(u, w)$. Using the fact that $C_{g_n}(pat(n,k)) = \lceil \log nrow(CM_{g_n}) \rceil$, where $nrow(CM_{g_n})$ is the number of distinct rows of the communication matrix CM_{g_n} (see [19]) and the fact that for the g_n it holds that $nrow(CM_{g_n}) = 2^{kl(n,k)} \geq 2^{k(n-1)}/kn$ we obtain the statement of Lemma 1. □

Similarly we define one-way communication computation for a discrete function $df(v)$ for the partition $pat(n,k)$of input v, $v = (u, w)$.

$$C_{df}(pat(n,k)) = \min\{C_\phi : \phi(pat(n,k)) \text{ computes } df(v)\}.$$

We define a communication complexity $C_f(pat(n,k), \varepsilon(n))$ of an $\varepsilon(n)$-approximation of the function f as follows:

$$C_f(pat(n,k), \varepsilon(n)) = \min\{C_{df}(pat(n,k)) : df(v) \text{ is an } \varepsilon(n)\text{-approximation of } f\}.$$

The next lemma states that the communication complexity of Boolean function g_n gives the lower bound for a relevant $\varepsilon(n)$-approximation df of function f in cube I_n^k.

Lemma 2. For arbitrary $\varepsilon(n)$ such that $\varepsilon(n) < \omega(\Delta_n)$ it holds that $C_{f_{\omega,g}}(pat(n,k), \varepsilon(n)) = \Omega(n)$.

Proof:
To prove the statement of the lemma we show that

$$C_{g_n}(pat(n,k)) \leq C_f(pat(n,k), \varepsilon(n))$$

and apply the statement of Lemma 1.

Suppose the contrary. That is, suppose that for some $\varepsilon(n) < \omega(\Delta_n)$ it holds that

$$C_{g_n}(pat(n,k)) > C_f(pat(n,k), \varepsilon(n)).$$

This means that there exists an $\varepsilon(n)$-approximation df of the function $f(x_1, \ldots, x_k)$ such that for $2^{kl(n,k)} \times 2^{kd(n,k)}$ communication matrices CM_{g_n} and CM_{df} of functions g_n and df it holds that

$$nrow(CM_{g_n}) > nrow(CM_{df}).$$

From the last inequality it follows that there exist two inputs u and u' such that two rows $row_{g_n}(u)$ and $row_{g_n}(u')$ are different but two rows $row_{df}(u)$ and $row_{df}(u')$ are equal. This means that there exists an input sequence w for which it holds that

$$\begin{aligned} g_n(u,w) &\neq g_n(u',w), \\ df(u,w) &= df(u',w). \end{aligned} \tag{14}$$

Let $g_n(u,w) = 1$ and $g_n(u',w) = 0$. Let us denote $v = (u,w),\ v' = (u',w)$. From the definition (6) of the $f_{\omega,g}$ we have that in the centers $b(v)$ and $b(v')$ of cubes $I_n^k(b(v))$ and $I_n^k(b(v'))$ it holds that

$$\begin{aligned} f_{\omega,g}(b(v)) &= (2g_n(v) - 1)\omega(\Delta_n)\Psi^k_{n,b(v)}(b(v)), \\ f_{\omega,g}(b(v')) &= (2g_n(v') - 1)\omega(\Delta_n)\Psi^k_{n,b(v)}(b(v')). \end{aligned}$$

From the definition of the function Ψ we have that in the centers $b(v)$ it holds $\Psi^k_{n,b(v)}(b(v)) = 1$ (Property 5). From the above we get:

$$f_{\omega,g}(b(v)) = \omega(\Delta_n), \tag{15}$$
$$f_{\omega,g}(b(v')) = -\omega(\Delta_n). \tag{16}$$

From Definition 1 it holds that

$$|f_{\omega,g}(b(v)) - df(v)| \leq \varepsilon(n) < \omega(\Delta_n), \tag{17}$$
$$|f_{\omega,g}(b(v')) - df(v')| \leq \varepsilon(n) < \omega(\Delta_n). \tag{18}$$

Now from (14), from (15), (16), (17), and (18) we get that

$$2\omega(\Delta_n) = |f_{\omega,g}(b(v)) - f_{\omega,g}(b(v'))| \leq$$

$$\leq |f_{\omega,g}(b(v)) - df(v)| + |f_{\omega,g}(b(v')) - df(v')| < 2\omega(\Delta_n).$$

The contradiction proves the statement of Lemma 2. □

Let $dh_i : \underbrace{\Sigma^n \times \cdots \times \Sigma^n}_{t} \to \mathbf{R}$, $1 \le i \le t$, be discrete functions and let $DF : \underbrace{\Sigma^n \times \cdots \times \Sigma^n}_{k} \to \mathbf{R}$ (here $\mathbf{R}$ denote the set of real numbers) be the following discrete function:

$$DF = F(dh_1(v_1^1, \ldots, v_t^1), \ldots, dh_s(v_1^s, \ldots, v_t^s)),$$

where function $F(y_1, \ldots, y_s)$ is an arbitrary continuous function.

Lemma 3. For the discrete function DF it holds that

$$C_{DF}(pat(n,k)) \le \sum_{i=1}^{s} C_{dh_i}(pat(n,k))$$

Proof:
The one-way communication protocol $\phi^*(pat(n,k))$ for function DF use processors P_u^* and P_w^*. Given an input $v = (u, w)$ (input u goes to P_u^* and w goes to P_w^*) protocol $\phi^*(pat(n,k))$ simulate in parallel one-way protocols $\phi_1(pat(n,k))$, $\phi_2(pat(n,k))$,..., $\phi_s(pat(n,k))$ which compute dh_1, dh_2, ..., dh_s, respectively. On getting a message from P_u^* and the input w processor P_w^* computes outputs y_1,...,y_s of protocols $\phi_1(pat(n,k))$, $\phi_2(pat(n,k))$, ... $\phi_s(pat(n,k))$ and then computes and outputs the value $DF(v)$. □

The next lemma uses the condition (7) of Theorem 4.1 to prove that discrete approximations of functions from $Sp^k[\mathcal{H}_{\omega_1}^t, \mathcal{H}_{\omega_2}^s]$ have small one-way communication complexity.

Lemma 4. Let functions ω_1, ω_2 satisfy conditions of Theorem 4.1 and let $\omega(\delta) = \omega_2(\omega_1(\delta))$. Then for an arbitrary function f from $Sp^k[\mathcal{H}_{\omega_1}^t, \mathcal{H}_{\omega_2}^s]$ there exists an $\varepsilon'(n) < \omega(\Delta_n)$, such that

$$C_f(pat(n,k), \varepsilon'(n)) = o(n).$$

Proof:
The function $f \in Sp^k[\mathcal{H}_{\omega_1}^t, \mathcal{H}_{\omega_2}^s]$ is represented as a superposition of the form

$$F\left(h_1(x_1^1, \ldots, x_t^1), \ldots, h_s(x_1^s, \ldots, x_t^s)\right),$$

where $F \in \mathcal{H}_{\omega_2}^s$ and $\{h_i : 1 \le i \le s\} \subset \mathcal{H}_{\omega_1}^t$. Denote $H = \{h_i : 1 \le i \le s\}$. Let $\varepsilon(n) = \omega_1(\Delta_n) / \log \frac{1}{\omega_1(\Delta_n)}$. We consider the following $\varepsilon(n)$-approximation dh for functions $h \in H$. Let $\alpha(n) = \min\{h(x) : x \in I_n^t = [\frac{1}{n}, \frac{2}{n}]^t\}$, and $\beta(n) = \max\{h(x) : x \in I_n^t = [\frac{1}{n}, \frac{2}{n}]^t\}$. Let

$$\mathcal{R}_{\varepsilon(n)} = A \cup \{\beta(n)\},$$

where

$$A = \left\{\alpha_i : \alpha_i = \alpha(n) + \varepsilon(n)i,\ i \in \{0, 1, \ldots, \lfloor \tfrac{\beta(n)-\alpha(n)}{\varepsilon(n)} \rfloor\right\}.$$

We define dh to be

$$dh : \underbrace{\Sigma^n \times \cdots \times \Sigma^n}_{t} \to \mathcal{R}_{\varepsilon(n)}.$$

To prove the statement of the lemma we show (using the selected $\varepsilon(n)$) the following two points:

1. There exists $\varepsilon'(n) < \omega(\Delta_n)$ such that the discrete function
$$DF(v_1, \dots, v_k) = F(dh_1(v_1^1, \dots, v_t^1), \dots, dh_s(v_1^s, \dots, v_t^s))$$
is the $\varepsilon'(n)$–approximation of our function f and that
2. Communication complexity of DF is small, that is,
$$C_{DF}(pat(n,k)) = o(n). \tag{19}$$

We start by showing the point 1. Let $v = (v_1, \dots, v_k) \in \Sigma^n \times \cdots \times \Sigma^n$. We will prove that for some $\varepsilon'(n) < \omega(\Delta_n)$ it holds that
$$|f(b(v)) - DF(v_1, \dots, v_k)| \le \varepsilon'(n). \tag{20}$$
Let $x = (x_1, \dots, x_k) = b(v) = (a(v_1), \dots, a(v_k))$. Due to the fact that for each $i \in \{1, 2, \dots, s\}$ the function dh_i is $\varepsilon(n)$-approximation of the continuous function h_i it holds that
$$|h_i(x_1^i, \dots, x_t^i) - dh_i(v_1^i, \dots, v_t^i)| \le \varepsilon(n).$$
Now from the fact that $F \in \mathcal{H}^s_{\omega_2}$ and the above we get that
$$|F\left(h_1(x_1^1, \dots, x_t^1), \dots, h_s(x_1^s, \dots, x_t^s)\right) -$$
$$F(dh_1(v_1^1, \dots, v_t^1), \dots, dh_s(v_1^s, \dots, v_t^s))| \le \omega_2(\varepsilon(n)),$$
where the function ω_2 is the modulus of continuity of F. Since $\mathcal{H}^s_{\omega_2}$ is a Hölder class ($\mathcal{H}^s_{\omega_2} = \mathcal{H}^s_\gamma$) we have that there exists a constant $M > 0$ such that $\omega_2(\delta) \le M(\delta^\gamma)$. From this we get that
$$\omega_2(\varepsilon(n)) \le M \left(\frac{\omega_1(\Delta_n)}{\log \frac{1}{\omega_1(\Delta_n)}} \right)^\gamma$$
Now we let $\varepsilon'(n) = M \left(\frac{\omega_1(\Delta_n)}{\log \frac{1}{\omega_1(\Delta_n)}} \right)^\gamma$. Since the function $\omega_1(\delta)$ decreases when δ decreases for n large enough we get that $\varepsilon'(n) < \omega(\Delta_n)$. The above proves (20) and the point 1.

To prove the point 2 (the relation (19)) we use Lemma 3. Thus to prove prove the relation (19) it is enough to show that $\varepsilon(n)$-approximation dh of the function $h \in H$ has small communication complexity, that is:
$$C_{dh}(pat(n,k)) = o(n) \tag{21}$$

First we show that
$$|\mathcal{R}_{\varepsilon(n)}| = 2^{o(n^{1-t/k})}. \tag{22}$$
To prove (22) it is enough to show that
$$|A| = 2^{o(n^{1-t/k})}.$$
Using the definition of the set A and from the fact that $\beta(n) - \alpha(n)$ is bounded by a constant we get that $|A| = 2^{O(\log 1/\varepsilon(n))}$. From the condition (7) of Theorem 4.1 (remind that $\Delta_n = \frac{1}{2(n+1)2^n}$) we get that
$$\log \frac{1}{\varepsilon(n)} = O\left(\log \frac{1}{\omega_1(\Delta_n)}\right) = o\left(\left(\log \frac{1}{\Delta_n}\right)^{1-t/k}\right) = o(n^{1-t/k}).$$

The last proves the relation (22).

With the function dh we associate a $2^{tl(n,k)} \times 2^{td(n,k)}$ communication matrix $CM_{dh}(n)$ whose (u, w) entry is $dh(u, w)$.

$$C_{dh}(pat(n,k), \varepsilon(n)) = \lceil \log nrow(CM_{dh}(n)) \rceil. \tag{23}$$

Clearly we have that

$$nrow(CM_{dh}(n)) \leq \min \left\{ 2^{tl(n,k)}, \left| \mathcal{R}_{\varepsilon(n)} \right|^{2^{td(n,k)}} \right\}$$

or

$$nrow(CM_{dh}(n)) \leq \left| \mathcal{R}_{\varepsilon(n)} \right|^{2^{td(n,k)}}. \tag{24}$$

From the definition of the partition $pat(n, k)$ we have that $d(n, k) = \lceil \frac{\log nk}{k} \rceil$. Using (24), (22) for the equality (23) we obtain inequality (21).

□

Finally combining statements of Lemma 4 and Lemma 2 we get the proof of Theorem 4.1.

6. Concluding remarks

We conclude with open problems. Whether using discrete approximation together with communication technique is possible to present hard functions $\mathcal{H}_1^k$ and from $\mathcal{F}_p^k$. That is to present an explicit functions:

- from $\mathcal{H}_1^k$ which cannot be presented by a superposition of functions from $\mathcal{H}_1^t$ if $t < k$;
- from $\mathcal{F}_p^k$ which cannot be represented by a superposition of functions from $\mathcal{F}_q^t$, if $\frac{k}{p} > \frac{t}{q}$?

Acknowledgment

We thank referees for important comments and their helpful feedback on an initial version of this paper.

References

[1] Ablayev, F., Ablayeva, S.: A Discrete Approximation and Communication Complexity Approach to the Superposition Problem, *Electronic Colloquium in Computational Complexity*, TR98-050, 1998.

[2] Ablayev, F., Ablayeva, S.: A Discrete Approximation and Communication Complexity Approach to the Superposition Problem, *Proc. Fundamentals of Computation Theory*, (R. Freivalds, Ed.), LNCS 2138, Springer-Verlag, Berlin, 2001.

[3] Arnold, V.: On functions of Three Variables, *Dokladi Akademii Nauk*, **114**, 1958, 679-681.

[4] Beiu, V., Zawadzki, A.: Why VLSI/NANO Library Design Should Use Kolmogorov's Superpositons, *Proc. Nano and Giga Challenges in Microelectronic NGCM*, 2004.

[5] Beiu, V., Zawadski, A., Andonie, R., Aunet, S.: Using Kolmogorov Inspired Gates for Low Power Nanoelectronics, *Proc. International Work-Conference on Artificial Neural Networks* (J. Cabestany, A. Prieto, F. Herna'ndez Eds.) LNCS 3512, Springer-Verlag, Berlin, 2005.

[6] Bläser, M., Vicari, E.: Distributed Algorithmic Mechanism Design and Algebraic Communication Complexity, *Proc. the 1st International Symposium on Algorithmic Game Theory* (B. Monien, U. Schroeder Eds.) LNCS 4997 Springer-Verlag, Berlin, 2008.

[7] Hansen, K., Lachish, O., Miltersen, P.: Hilbert's thirteenth problem and circuit complexity *The 20th International Symposium on Algorithms and Complexity* December 2009, (To appear in *Proceedings*)

[8] Grigoriev, D.: Probabilisitc Communication Complexity Over the Reals, *Computational Complexity Theoretical Computer Science*, **17**(536–548), 2008.

[9] Hilbert,D.: Mathematische Probleme, *Nachr. Akad. Wiss*, Gottingen, 253-297, 1900.

[10] Hromkovich, J.: Communication complexity and parallel computing, *Speinger-Verlag, Berlin*, 1997.

[11] Jukna, S.: Entropy of operators or why matrix multiplication is hard for small depth circuits, *Electronic Colloquium in Computational Complexity*, TR08-019, 2008.

[12] Kolmogorov, A.: On Representation of Continuous Functions of Several Variables by a superposition of Continuous Functions of one Variable and Sum Operation. Dokladi Akademii Nauk, **114**, (953-956), 1957.

[13] Kolmogorov, A.: Different approaches on the estimation of hardness of defining and computing of functions, *Proc. of the International Congress of Mathematicians*, Stockholm, (352-356), 1963.

[14] Kushilevitz, E., Nisan, N.: Communication complexity, *Cambridge University Press*, 1997.

[15] Lorentz, G.: Metric entropy and approximation *Bulletin of American Mathematical Society* **72**, (903-937), 1966.

[16] Marchenkov, S.: On One Method of Analysis of superpositions of Continuous Functions, *Problemi Kibernetici*, **37**, (5-17), 1980.

[17] Timan, A.: Theory of approximation of functions of a real variables, *Oxford Pergammon Press*, 1963.

[18] Vitushkin, A.: On Representation of Functions by Means of Superpositions and Related Topics, L'Enseignement mathematique, **23**, (255-320), 1977.

[19] Yao, A.: Some Complexity Questions Related to Distributive Computing, *Proc. of ACM Symposium on the Theory of Computing*, **11**, (209-213), 1979.