

---

---

# An upper bound of unreliability of non-branching programs in an arbitrary complete finite basis for one-type constant faults at the outputs of computational operators

S. M. Grabovskaya<sup>1,\*</sup> and M. A. Alekhina<sup>2,\*\*</sup>

(Submitted by F. M. Ablayev)

<sup>1</sup>*Penza State University, Krasnaya st. 40, Penza, 440026 Russia*

<sup>2</sup>*Penza State Technological University, Baidukova passway/ Gagarin st. 1a/11, Penza, 440039 Russia*

Received July 15, 2017

**Abstract**—The implementation of Boolean functions by non-branching programs with a conditional stop operator is considered in an arbitrary complete finite basis. We assume that conditional stop operators of the program are absolutely reliable while all computational operators are prone to the output one-type constant faults of either type 0 or type 1. An upper bound for unreliability of the programs was found in different bases. It was proved that in some cases any Boolean function can be implemented by the non-branching program of arbitrarily high preassigned reliability.

**2010 Mathematical Subject Classification:** 68R99, 68M15, 94C12

**Keywords and phrases:** *a circuit, a non-branching program with conditional stop operators, output one-type constant faults, unreliability, a Boolean function*

## 1. INTRODUCTION

We consider the implementation of Boolean functions by non-branching programs with a conditional stop operator [1, 2] in an arbitrary complete finite basis. The computational operators of the program are assumed to be prone to output constant faults of either only type 0 or only type 1, and conditional stop operators are absolutely reliable. The program with a conditional stop operator is characterized by the presence of the conditional stop command giving the possibility of early stoppage of the program work under certain conditions (namely, under receipt unit at the stop operator input).

The notion of a non-branching program is closely related to the concept of a circuit of functional elements (FE). Diagrams of FE are models of electronic circuits, and non-branching programs (both with a conditional stop and without it) model the operation of computing devices. Despite these differences the results on reliability and complexity obtained for circuits of FE are transferred to non-branching programs without stop operators and vice versa. Generally speaking, circuits of FE can be considered a special case of non-branching programs when there are no stop operators in them.

For the first time the problem of synthesis of reliable circuits of unreliable elements was considered J. von Neumann [3]. He supposed that circuit elements are prone to inverse faults at the outputs. To raise reliability he used multiple duplication of original circuits which of course led to increase in complexity.

Therefore complexity was attended in further researches, among which we notice results of S. I. Ortyukov [4], D. Uhlig [5], S. V. Yablonsky [6]. The circuits proposed by S. I. Ortyukov and D. Uhlig were asymptotically optimal on complexity and functioned with a certain level of reliability.

Methods of constructing asymptotically optimal reliable circuits of unreliable elements were first considered by M. A. Alekhina [7]. Then researches were continued by V. V. Chugunova [8], A. V. Vasin [9] and O. Y. Barsukova [10].

---

\* E-mail: swetazin@mail.ru

\*\* E-mail: alekhina.marina19@yandex.ru

A. V. Chashkin [2] researched non-branching programs with a conditional stop operator. He found asymptotic estimates for the average calculation time of non-branching programs on the assumption that all program operators are absolutely reliable.

Prior to the author's work the problem of constructing reliable (and asymptotically optimal on reliability) non-branching programs with a conditional stop operator was not considered. The problem of constructing reliable non-branching programs was solved in [11]. For the first time this problem was researched for inverse faults at the outputs of computational operators. Let us formulate necessary definitions, denotations and known results about reliability of circuits of FE and non-branching programs [12].

## 2. DEFINITIONS AND DENOTATIONS

Let  $X = \{x_1, \dots, x_n\}$  be a set of independent Boolean variables, and let  $\mathbf{x} = (x_1, \dots, x_n)$  be a set of independent variables,  $n \in \mathbf{N}$ . Introduce a set of variables  $Y = \{y_1, \dots, y_l\}$ ; we call it the set of inner variables,  $l \in \mathbf{N}$ . Denote the output variable by  $z$ .

Furthermore, let  $a \in Y \cup \{z\}$  and  $b_1, \dots, b_d \in X \cup Y \cup \{z\}$  ( $d \in \{1, \dots, n\}$ ). Denote  $h$  a Boolean function from the basis  $B$  depending on at most  $d$  variables. We understand a computing command as the expression  $p: a = h(b_1, \dots, b_d)$ . We treat the variable  $a$  as the output of the computing command and we do variables  $b_1, \dots, b_d$  as inputs of this command.

Let now  $a \in X \cup Y \cup \{z\}$ . We understand a stop command as the expression  $p: \text{Stop}(a)$ . We treat the variable  $a$  as the input of the stop command  $p$ .

**Definition 1.** A sequence  $Pr = p_1 \dots p_j \dots p_L$  consisting of computing commands and stop ones is called a non-branching program with a conditional stop operator if with any  $j \in \{1, \dots, L\}$  each input of the command  $p_j$  is either an independent variable or the output of some computing command.

The non-branching program works at discrete time moments  $t = 0, 1, 2, \dots$ , it does not change values of independent variables, but changes values of inner variables  $y_i$  ( $i \in \{1, \dots, l\}$ ) and that of the output one  $z$ .

The result of the program  $Pr$  operating on the set  $\mathbf{x}$  is denoted by  $Pr(\mathbf{x})$ . The value  $Pr(\mathbf{x})$  is equal the value of the output variable  $z$  at the time of program stop. The program  $Pr$  calculates a Boolean function  $f(\mathbf{x})$  ( $\mathbf{x} = (x_1, \dots, x_n)$ ) if with no faults in the program for any  $\mathbf{x} \in \{0, 1\}^n$  there is the equality  $Pr(\mathbf{x}) = f(\mathbf{x})$ .

We assume that all computational operators of the basis  $B$  independently of each other with probability  $\varepsilon$  ( $\varepsilon \in (0, 1/2)$ ) are prone to one-type constant faults at the outputs. Constant faults of type 0 (1) at the outputs are characterized by the fact that in good condition a computational operator implements a function  $\varphi$ , ascribed to him, and in failure — a function 0 (1). Conditional stop operators are supposed to be absolutely reliable.

We consider that the program with unreliable operators implements a Boolean function  $f(\mathbf{x})$  if with no faults in it for each input vector  $\mathbf{a}$  the value of the output variable  $z$  is equal  $f(\mathbf{a})$ .

**Remark 1.** A circuit of FE can be considered a special case of a non-branching program, namely, a non-branching program without stop operators.

**Definition 2.** Unreliability  $N_\varepsilon(Pr)$  of the program  $Pr$  is the maximum fault at the output of the program  $Pr$  for all possible input vectors.

For circuits of FE the following result is known.

**Theorem 1.** [13] In an arbitrary complete finite basis for faults of type 0 (1) at the FE outputs any Boolean function  $f$  can be implemented by the circuit  $S$  such that  $N_\varepsilon(S) \leq 3,11\varepsilon$  for all  $\varepsilon \in (0, 1/960]$ .

To raise reliability of the original circuits (programs) we use the function  $g(x_1, x_2, x_3, x_4)$  of the form  $(x_1^{\sigma_1} x_2^{\sigma_2} \vee x_3^{\sigma_3} x_4^{\sigma_4})^{\sigma_5}$  ( $\sigma_i \in \{0, 1\}$ ,  $i \in \{1, 2, 3, 4, 5\}$ ). Theorem 2 holds.

**Theorem 2.** [14] *Let  $B$  be a complete finite basis, and let there exists  $N$  such that any Boolean function  $f$  can be implemented by the non-branching program  $R_f$  with unreliability  $N_\varepsilon(R_f) \leq N$ . Let  $g(x_1, x_2, x_3, x_4)$  be a function of the form  $(x_1^{\sigma_1} x_2^{\sigma_2} \vee x_3^{\sigma_3} x_4^{\sigma_4})^{\sigma_5}$  ( $\sigma_i \in \{0, 1\}$ ,  $i \in \{1, 2, 3, 4, 5\}$ ),  $Pr_g$  – a program implementing  $g(x_1, x_2, x_3, x_4)$ , and  $N_\varepsilon(Pr_g)$  – unreliability of the program  $Pr_g$ . Then in this basis any Boolean function  $f$  can be implemented by such program  $Pr_f$  that there is the inequality*

$$N_\varepsilon(Pr_f) \leq \max\{\nu^1, \nu^0\} + 4N \cdot N_\varepsilon(Pr_g) + 4N^2, \quad (1)$$

where  $\nu^1$  and  $\nu^0$  are fault probabilities of the program  $Pr_g$  on the vectors  $(\bar{\sigma}_1, \bar{\sigma}_2, \bar{\sigma}_3, \bar{\sigma}_4)$  and  $(\sigma_1, \sigma_2, \sigma_3, \sigma_4)$  respectively.

The sets  $(\bar{\sigma}_1, \bar{\sigma}_2, \bar{\sigma}_3, \bar{\sigma}_4)$  and  $(\sigma_1, \sigma_2, \sigma_3, \sigma_4)$  are called the characteristic sets of the function  $g(x_1, x_2, x_3, x_4)$ .

**Lemma 1.** [15] *The variables identification in any non-linear and non-especial function of three or more variables can result in either a non-linear function of two variables or an especial function.*

In other words, by means of substitution (identification and/or renaming of variables [16]) it's possible to obtain either some non-linear function of two variables  $\phi(x_1, x_2) = x_1 x_2 \oplus \alpha_1 x_1 \oplus \alpha_2 x_2 \oplus \alpha_0$  or some especial function  $\varphi(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3 \oplus \beta_1 x_1 \oplus \beta_2 x_2 \oplus \beta_3 x_3 \oplus \beta_0$  where  $\alpha_0, \alpha_1, \alpha_2, \beta_0, \beta_1, \beta_2, \beta_3 \in \{0, 1\}$ .

Since in any complete finite basis there is a non-linear function, we further assume without loss of generality that the considered complete basis  $B$  contains either a non-linear function of two variables or an especial function.

For a non-linear function of two variables up to variable renaming there are 2 cases:

1.  $\phi(x_1, x_2) = x_1^{\alpha_1} \vee x_2^{\alpha_2}$  (generalized disjunction);
2.  $\phi(x_1, x_2) = x_1^{\alpha_1} \& x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ) (generalized conjunction).

For an especial function up to variable renaming there are 8 cases:

1.  $\varphi_1(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3 \oplus x_1 \oplus x_2 \oplus x_3 \oplus 1$ ;
2.  $\varphi_2(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3 \oplus x_1 \oplus 1$ ;
3.  $\varphi_3(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3 \oplus x_1 \oplus x_2 \oplus x_3$ ;
4.  $\varphi_4(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3 \oplus x_1$ ;
5.  $\varphi_5(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3 \oplus x_1 \oplus x_2 \oplus 1$ ;
6.  $\varphi_6(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3 \oplus x_1 \oplus x_2$ ;
7.  $\varphi_7(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3 \oplus 1$ ;
8.  $\varphi_8(x_1, x_2, x_3) = x_1 x_2 \oplus x_1 x_3 \oplus x_2 x_3$ .

## 3. MAIN RESULTS

If the computational operators of a non-branching program are prone to constant faults of type 0 at the outputs, then Theorem 3 is valid.

**Theorem 3.** 1) If the complete finite basis  $B$  contains one of the functions  $\varphi_1(x_1, x_2x_3)$ ,  $\varphi_4(x_1, x_2x_3)$ ,  $\varphi_5(x_1, x_2x_3)$ ,  $\varphi_6(x_1, x_2x_3)$ ,  $\varphi_7(x_1, x_2x_3)$ , or the function of the form  $x_1^{\alpha_1} \& x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), then any Boolean function  $f$  for  $\forall t \in \mathbf{N}$  can be implemented by the non-branching program  $Pr_f$  such that  $N_\varepsilon(Pr_f) \leq \varepsilon^t$  for all  $\varepsilon \in (0, 1/960]$ .

2) If the complete finite basis  $B$  contains one of the functions  $\varphi_2(x_1, x_2x_3)$ ,  $\varphi_3(x_1, x_2x_3)$ ,  $\varphi_8(x_1, x_2x_3)$ , or the function of the form  $x_1^{\alpha_1} \vee x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), then any Boolean function  $f$  can be implemented by the non-branching program  $Pr_f$  with unreliability no more  $\varepsilon + 9\varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .

If the computational operators of a non-branching program are prone to constant faults of type 1 at the outputs, then Theorem 4 is valid.

**Theorem 4.** 1) If a complete finite basis  $B$  contains one of the functions  $\varphi_1(x_1, x_2x_3)$ ,  $\varphi_4(x_1, x_2x_3)$ ,  $\varphi_5(x_1, x_2x_3)$ ,  $\varphi_6(x_1, x_2x_3)$ ,  $\varphi_7(x_1, x_2x_3)$ , or the function of the form  $x_1^{\alpha_1} \& x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), then any Boolean function  $f$  can be implemented by the non-branching program  $Pr_f$  with unreliability no greater  $\varepsilon + 9\varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .

2) If a complete finite basis  $B$  contains one of the functions  $\varphi_2(x_1, x_2x_3)$ ,  $\varphi_3(x_1, x_2x_3)$ ,  $\varphi_8(x_1, x_2x_3)$ , or the function of the form  $x_1^{\alpha_1} \vee x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), then any Boolean function  $f$  for  $\forall t \in \mathbf{N}$  can be implemented by the non-branching program  $Pr_f$  such that  $N_\varepsilon(Pr_f) \leq \varepsilon^t$  for all  $\varepsilon \in (0, 1/960]$ .

We preface Lemmas 2 – 9 to the proves of Theorems 3 and 4.

**Lemma 2.** In a complete finite basis, containing the function  $\varphi_1(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus x_2 \oplus x_3 \oplus 1$ , the function  $g(x_1, x_2, x_3, x_4) = x_1x_2 \vee \bar{x}_3x_4$  can be implemented by the non-branching program  $Pr_g$  such that  $N_\varepsilon(Pr_g) \leq \varepsilon$ , and besides  $\max\{\nu^1, \nu^0\} = \varepsilon$  for constant faults of type 0 and  $\max\{\nu^1, \nu^0\} = 0$  for constant faults of type 1 at the outputs of computational operators.

*Proof.* Let the complete finite basis  $B$  contains the function  $\varphi_1(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus x_2 \oplus x_3 \oplus 1 = x_1x_2x_3 \vee \bar{x}_1\bar{x}_2\bar{x}_3$ . In this basis we construct the non-branching program  $Pr_g$  implementing the function  $g(x_1, x_2, x_3, x_4) = x_1x_2 \vee \bar{x}_3x_4$ :

$$z = \varphi_1(x_1, x_2, x_3)$$

$$\text{stop}(x_3)$$

$$z = x_4$$

$$\text{stop}(x_4)$$

$$z = x_2$$

$$\text{stop}(x_1)$$

$$z = x_1$$

Obviously, unreliability of the program  $Pr_g$  is at most  $\varepsilon$  because the program  $Pr_g$  contains just one unreliable computational operator.

For the program  $Pr_g$  we estimate probabilities of faults on the characteristic sets in two cases: 1) for constant faults of type 0, and 2) for constant faults of type 1 at the outputs of computational operators.

1)  $\nu^1 = P_1(0, 0, 1, 0) = 0, \nu^0 = P_0(1, 1, 0, 1) = 0$ . Thus  $\max\{\nu^1, \nu^0\} = 0$ .

2)  $\nu^1 = P_1(0, 0, 1, 0) = \varepsilon, \nu^0 = P_0(1, 1, 0, 1) = 0$ . Thus  $\max\{\nu^1, \nu^0\} = \varepsilon$ .

□

**Lemma 3.** *In a complete finite basis, containing the function  $\varphi_2(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus 1$ , the function  $g(x_1, x_2, x_3, x_4) = (x_1 \vee x_2)(x_3 \vee \bar{x}_4)$  can be implemented by the non-branching program  $Pr_g$  such that  $N_\varepsilon(Pr_g) \leq \varepsilon$ , and besides  $\max\{\nu^1, \nu^0\} = \varepsilon$  for constant faults of type 0 and  $\max\{\nu^1, \nu^0\} = 0$  for constant faults of type 1 at the outputs of computational operators.*

*Proof.* Let the complete finite basis  $B$  contains the function  $\varphi_2(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus 1 = (\bar{x}_1 \vee x_2 \vee x_3)(x_1 \vee \bar{x}_2 \vee \bar{x}_3)$ . In this basis we construct the non-branching program  $Pr_g$  implementing the function  $g(x_1, x_2, x_3, x_4) = (x_1 \vee x_2)(x_3 \vee \bar{x}_4)$ :

$$z = \varphi_2(x_1, x_2, x_3)$$

$$\text{stop}(x_3)$$

$$z = x_3$$

$$\text{stop}(x_4)$$

$$z = x_1$$

$$\text{stop}(x_1)$$

$$z = x_2$$

Obviously, unreliability of the program  $Pr_g$  is at most  $\varepsilon$  because the program  $Pr_g$  contains just one unreliable computational operator.

For the program  $Pr_g$  we estimate probabilities of faults on the characteristic sets in two cases: 1) for constant faults of type 0, and 2) for constant faults of type 1 at the outputs of computational operators.

1)  $\nu^1 = P_1(0, 0, 0, 1) = 0, \nu^0 = P_0(1, 1, 1, 0) = \varepsilon$ . Thus  $\max\{\nu^1, \nu^0\} = \varepsilon$ .

2)  $\nu^1 = P_1(0, 0, 0, 1) = 0, \nu^0 = P_0(1, 1, 1, 0) = 0$ . Thus  $\max\{\nu^1, \nu^0\} = 0$ .

□

**Lemma 4.** *In a complete finite basis, containing the function  $\varphi_3(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus x_2 \oplus x_3$ , the function  $g(x_1, x_2, x_3, x_4) = (\bar{x}_1 \vee \bar{x}_2)(x_3 \vee x_4)$  can be implemented by the non-branching program  $Pr_g$  such that  $N_\varepsilon(Pr_g) \leq \varepsilon$ , and besides  $\max\{\nu^1, \nu^0\} = \varepsilon$  for constant faults of type 0 and  $\max\{\nu^1, \nu^0\} = 0$  for constant faults of type 1 at the outputs of computational operators.*

*Proof.* Let the complete finite basis  $B$  contains the function  $\varphi_3(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus x_2 \oplus x_3 = (x_1 \vee x_2 \vee x_3)(\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3)$ . In this basis we construct the non-branching program  $Pr_g$  implementing the function  $g(x_1, x_2, x_3, x_4) = (\bar{x}_1 \vee \bar{x}_2)(x_3 \vee x_4)$ :

$$z = \varphi_3(x_1, x_2, x_3)$$

$$\text{stop}(x_3)$$

$$z = \varphi_3(x_1, x_2, x_4)$$

$$\text{stop}(x_4)$$

$$z = x_4$$

It is easy to verify that unreliability of the program  $Pr_g$  is no greater  $\varepsilon$  despite  $Pr_g$  contains two unreliable computational operators. Indeed, with this method of constructing a non-branching program, when independent variables enter the inputs of the stop operators, the fault in the output of the program on an arbitrary input set of variables can occur due to only the fault of one of the computational operators while the work of the other computational operator does not affect the result of the program.

For the program  $Pr_g$  we estimate probabilities of faults on the characteristic sets in two cases: 1) for constant faults of type 0, and 2) for constant faults of type 1 at the outputs of computational operators.

$$1) \nu^1 = P_1(1, 1, 0, 0) = 0, \nu^0 = P_0(0, 0, 1, 1) = \varepsilon. \text{ Thus } \max\{\nu^1, \nu^0\} = \varepsilon.$$

$$2) \nu^1 = P_1(1, 1, 0, 0) = 0, \nu^0 = P_0(0, 0, 1, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = 0.$$

□

**Lemma 5.** *In a complete finite basis, containing the function  $\varphi_4(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1$ , the function  $g(x_1, x_2, x_3, x_4) = \bar{x}_1x_2 \vee \bar{x}_3x_4$  can be implemented by the non-branching program  $Pr_g$  such that  $N_\varepsilon(Pr_g) \leq \varepsilon$ , and besides  $\max\{\nu^1, \nu^0\} = 0$  for constant faults of type 0 and  $\max\{\nu^1, \nu^0\} = \varepsilon$  for constant faults of type 1 at the outputs of computational operators.*

*Proof.* Let the complete finite basis  $B$  contains the function  $\varphi_4(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 = \bar{x}_1x_2x_3 \vee x_1\bar{x}_2\bar{x}_3$ . In this basis we construct the non-branching program  $Pr_g$  implementing the function  $g(x_1, x_2, x_3, x_4) = \bar{x}_1x_2 \vee \bar{x}_3x_4$ :

$$z = \varphi_4(x_1, x_2, x_3)$$

$$\text{stop}(x_3)$$

$$z = x_4$$

$$\text{stop}(x_4)$$

$$\text{stop}(x_1)$$

$$z = x_2$$

Obviously, unreliability of the program  $Pr_g$  is at most  $\varepsilon$  because the program  $Pr_g$  contains just one unreliable computational operator.

For the program  $Pr_g$  we estimate probabilities of faults on the characteristic sets in two cases: 1) for constant faults of type 0, and 2) for constant faults of type 1 at the outputs of computational operators.

$$1) \nu^1 = P_1(1, 0, 1, 0) = 0, \nu^0 = P_0(0, 1, 0, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = 0.$$

$$2) \nu^1 = P_1(1, 0, 1, 0) = \varepsilon, \nu^0 = P_0(0, 1, 0, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = \varepsilon.$$

□

**Lemma 6.** *In a complete finite basis, containing the function  $\varphi_5(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus x_2 \oplus 1$ , the function  $g(x_1, x_2, x_3, x_4) = \bar{x}_1x_2 \vee \bar{x}_3x_4$  can be implemented by the non-branching program  $Pr_g$  such that  $N_\varepsilon(Pr_g) \leq \varepsilon$ , and besides  $\max\{\nu^1, \nu^0\} = 0$  for constant faults of type 0 and  $\max\{\nu^1, \nu^0\} = \varepsilon$  for constant faults of type 1 at the outputs of computational operators.*

*Proof.* Let the complete finite basis  $B$  contains the function  $\varphi_5(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus x_2 \oplus 1 = \bar{x}_1\bar{x}_2 \vee \bar{x}_1x_3 \vee \bar{x}_2x_3$ . In this basis we construct the non-branching program  $Pr_g$  implementing the function  $g(x_1, x_2, x_3, x_4) = \bar{x}_1x_2 \vee \bar{x}_3x_4$ :

$$\begin{aligned} z &= \varphi_5(x_1, x_2, x_3) \\ stop(x_3) \\ z &= x_4 \\ stop(x_4) \\ stop(x_1) \\ z &= x_2 \end{aligned}$$

Obviously, unreliability of the program  $Pr_g$  is no greater  $\varepsilon$  because the program  $Pr_g$  contains just one unreliable computational operator.

For the program  $Pr_g$  we estimate probabilities of faults on the characteristic sets in two cases: 1) for constant faults of type 0, and 2) for constant faults of type 1 at the outputs of computational operators.

$$1) \nu^1 = P_1(1, 0, 1, 0) = 0, \nu^0 = P_0(0, 1, 0, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = 0.$$

$$2) \nu^1 = P_1(1, 0, 1, 0) = \varepsilon, \nu^0 = P_0(0, 1, 0, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = \varepsilon.$$

□

**Lemma 7.** *In a complete finite basis, containing the function  $\varphi_6(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus x_2$ , the function  $g(x_1, x_2, x_3, x_4) = x_1x_2 \vee \bar{x}_3x_4$  can be implemented by the non-branching program  $Pr_g$  such that  $N_\varepsilon(Pr_g) \leq \varepsilon$ , and besides  $\max\{\nu^1, \nu^0\} = 0$  for constant faults of type 0 and  $\max\{\nu^1, \nu^0\} = \varepsilon$  for constant faults of type 1 at the outputs of computational operators.*

*Proof.* Let the complete finite basis  $B$  contains the function  $\varphi_6(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1 \oplus x_2 = x_1x_2 \vee x_1\bar{x}_3 \vee x_2\bar{x}_3$ . In this basis we construct the non-branching program  $Pr_g$  implementing the function  $g(x_1, x_2, x_3, x_4) = x_1x_2 \vee \bar{x}_3x_4$ :

$$\begin{aligned} z &= \varphi_6(x_1, x_2, x_3) \\ stop(x_3) \\ z &= x_4 \\ stop(x_4) \\ z &= x_2 \\ stop(x_1) \\ z &= x_1 \end{aligned}$$

Obviously, unreliability of the program  $Pr_g$  is no greater  $\varepsilon$  because the program  $Pr_g$  contains just one unreliable computational operator.

For the program  $Pr_g$  we estimate probabilities of faults on the characteristic sets in two cases: 1) for constant faults of type 0, and 2) for constant faults of type 1 at the outputs of computational operators.

$$1) \nu^1 = P_1(0, 0, 1, 0) = 0, \nu^0 = P_0(1, 1, 0, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = 0.$$

$$2) \nu^1 = P_1(0, 0, 1, 0) = \varepsilon, \nu^0 = P_0(1, 1, 0, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = \varepsilon.$$

□

**Lemma 8.** *In a complete finite basis, containing the function  $\varphi_7(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus 1$ , the function  $g(x_1, x_2, x_3, x_4) = \bar{x}_1\bar{x}_2 \vee \bar{x}_3x_4$  can be implemented by the non-branching program  $Pr_g$  such that  $N_\varepsilon(Pr_g) \leq \varepsilon$ , and besides  $\max\{\nu^1, \nu^0\} = 0$  for constant faults of type 0 and  $\max\{\nu^1, \nu^0\} = \varepsilon$  for constant faults of type 1 at the outputs of computational operators.*

*Proof.* Let the complete finite basis  $B$  contains the function  $\varphi_7(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus 1 = \bar{x}_1\bar{x}_2 \vee \bar{x}_1\bar{x}_3 \vee \bar{x}_2\bar{x}_3$ . In this basis we construct the non-branching program  $Pr_g$  implementing the function  $g(x_1, x_2, x_3, x_4) = \bar{x}_1\bar{x}_2 \vee \bar{x}_3x_4$ :

$$z = \varphi_7(x_1, x_2, x_3)$$

$$\text{stop}(x_3)$$

$$z = x_4$$

$$\text{stop}(x_4)$$

$$\text{stop}(x_1)$$

$$\text{stop}(x_2)$$

$$z = \varphi_7(x_1, x_2, x_3)$$

Unreliability of the program  $Pr_g$  is at most  $\varepsilon$  since independent variables enter the inputs of the stop operators. So on an arbitrary input set of variables, the fault in the output of the program can occur due to only the fault of one of the computational operators while the work of the other computational operator does not affect the result of the program.

For the program  $Pr_g$  we estimate probabilities of faults on the characteristic sets in two cases: 1) for constant faults of type 0, and 2) for constant faults of type 1 at the outputs of computational operators.

$$1) \nu^1 = P_1(1, 1, 1, 0) = 0, \nu^0 = P_0(0, 0, 0, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = 0.$$

$$2) \nu^1 = P_1(1, 1, 1, 0) = \varepsilon, \nu^0 = P_0(0, 0, 0, 1) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = \varepsilon.$$

□

**Lemma 9.** *In a complete finite basis, containing the function  $\varphi_8(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3$ , the function  $g(x_1, x_2, x_3, x_4) = (x_1 \vee x_2)(x_3 \vee \bar{x}_4)$  can be implemented by the non-branching program  $Pr_g$  such that  $N_\varepsilon(Pr_g) \leq \varepsilon$ , and besides  $\max\{\nu^1, \nu^0\} = \varepsilon$  for constant faults of type 0 and  $\max\{\nu^1, \nu^0\} = 0$  for constant faults of type 1 at the outputs of computational operators.*

*Proof.* Let the complete finite basis  $B$  contains the function  $\varphi_8(x_1, x_2, x_3) = x_1x_2 \oplus x_1x_3 \oplus x_2x_3 = x_1x_2 \vee x_1x_3 \vee x_2x_3$ . In this basis we construct the non-branching program  $Pr_g$  implementing the function  $g(x_1, x_2, x_3, x_4) = (x_1 \vee x_2)(x_3 \vee \bar{x}_4)$ :

$$z = \varphi_8(x_1, x_2, x_3)$$

$$\text{stop}(x_3)$$

$$z = x_3$$

$$\text{stop}(x_4)$$

$$z = x_1$$

$$\text{stop}(x_1)$$

$$z = x_2$$



Obviously, unreliability of the program  $Pr_g$  is no greater  $\varepsilon$  because the program  $Pr_g$  contains just one unreliable computational operator.

For the program  $Pr_g$  we estimate probabilities of faults on the characteristic sets in two cases: 1) for constant faults of type 0, and 2) for constant faults of type 1 at the outputs of computational operators.

$$1) \nu^1 = P_1(0, 0, 0, 1) = 0, \nu^0 = P_0(1, 1, 1, 0) = \varepsilon. \text{ Thus } \max\{\nu^1, \nu^0\} = \varepsilon.$$

$$2) \nu^1 = P_1(0, 0, 0, 1) = 0, \nu^0 = P_0(1, 1, 1, 0) = 0. \text{ Thus } \max\{\nu^1, \nu^0\} = 0.$$

□

Let us prove Theorem 3.

*Proof.* Let  $B$  be an arbitrary complete finite basis; the conditional stop operators of some non-branching program are absolutely reliable, and the computational operators independently of each other with probability  $\varepsilon$  ( $\varepsilon \in (0, 1/960]$ ) go to the failure conditions of type 0 at the outputs. Let  $f$  be any Boolean function.

For the basis  $B$  by Lemma 1 at least one of the following conditions holds:

- the basis  $B$  contains a non-linear function of two variables, i.e. either
  - 1) generalized disjunction  $x_1^{\alpha_1} \vee x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), or
  - 2) generalized conjunction  $x_1^{\alpha_1} \& x_2^{\alpha_2}$ ;
- the basis  $B$  contains a special function, i.e. either
  - 3) one of the functions  $\varphi_i(x_1, x_2, x_3)$ ,  $i \in \{1, 4, 5, 6, 7\}$ , or
  - 4) one of the functions  $\varphi_i(x_1, x_2, x_3)$ ,  $i \in \{2, 3, 8\}$ .

Let's consider each of these four cases.

1) Let the basis  $B$  contains some function of the form  $x_1^{\alpha_1} \vee x_2^{\alpha_2}$ .

It is shown in [17] that in the basis  $B$  any Boolean function  $f$  can be implemented by the non-branching program  $Pr_f$  with unreliability  $N_\varepsilon(Pr_f) \leq \varepsilon + 9\varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ . Theorem 3 is true in this part.

In cases 2 and 3 first of all we implement the function  $f$  by the non-branching program  $Pr_f$  such that  $N_\varepsilon(Pr_f) \leq \varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ , and then the function  $f$  for  $\forall t \in \mathbf{N}$  is implemented by the program with unreliability at most  $\varepsilon^t$  for all  $\varepsilon \in (0, 1/960]$ . Note that for  $t = 1$  and  $t = 2$  this theorem is true since  $\varepsilon^2 \leq \varepsilon$ . Therefore, it remains to consider the case  $t \geq 3$ .

2) Let the basis  $B$  contains some function of the form  $x_1^{\alpha_1} \& x_2^{\alpha_2}$ .

Previously [18] it was proved that in the basis  $B$  any Boolean function  $f$  can be implemented by the non-branching program  $Pr_f$  such that  $N_\varepsilon(Pr_f) \leq \varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .

3) Let the basis  $B$  contains at least one of the functions  $\varphi_1(x_1, x_2, x_3)$ ,  $\varphi_4(x_1, x_2, x_3)$ ,  $\varphi_5(x_1, x_2, x_3)$ ,  $\varphi_6(x_1, x_2, x_3)$ , or  $\varphi_7(x_1, x_2, x_3)$ .

Then the function  $g(x_1, x_2, x_3, x_4)$ , equal to either the function  $x_1x_2 \vee \bar{x}_3x_4$  (Lemmas 2 and 7), either  $\bar{x}_1x_2 \vee \bar{x}_3x_4$  (Lemmas 5 and 6), or  $\bar{x}_1\bar{x}_2 \vee \bar{x}_3x_4$  (Lemma 8), can be implemented by the program  $Pr_g$  with unreliability  $N_\varepsilon(Pr_g) \leq \varepsilon$  and  $\max\{\nu^1, \nu^0\} = 0$ .

By Theorem 1 the function  $f$  can be implemented by the circuit  $S$  such that  $N_\varepsilon(S) \leq 3,11\varepsilon$  for all  $\varepsilon \in (0, 1/960]$ .

We apply Theorem 2 using the program  $Pr_g$  and taking the circuit  $S$  as the program  $R_f$  (see Remark 1). We construct the program  $Pr_f$  assuming  $N = 3,11\varepsilon$  and taking into account the conditions  $\max\{\nu^1, \nu^0\} = 0$  and  $N_\varepsilon(Pr_g) \leq \varepsilon$ . By Theorem 2 the non-branching program  $Pr_f$  implements the function  $f$  with unreliability  $N_\varepsilon(Pr_f) \leq 52\varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .

Let's perform one more iteration step. We now take for  $R_f$  already constructed program  $Pr_f$  unreliability of which for all  $\varepsilon \in (0, 1/960]$  satisfies the inequality  $N_\varepsilon(R_f) \leq 52\varepsilon^2 \leq 0,0542\varepsilon$ . Let  $N = 0,0542\varepsilon$ . We again use Theorem 2 considering  $\max\{\nu^1, \nu^0\} = 0$  and  $N_\varepsilon(Pr_g) \leq \varepsilon$ . Then the function  $f$  can be implemented by the non-branching program  $Pr_f^{(2)}$  such that  $N_\varepsilon(Pr_f^{(2)}) \leq 0,23\varepsilon^2 \leq \varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .

Thus, in cases 2 and 3 an arbitrary function  $f$  can be implemented by such non-branching program (denote it by  $Pr_f^*$ ) that  $N_\varepsilon(Pr_f^*) \leq \varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .

In each of cases 2 and 3 we again use Theorem 2 taking now for  $R_f$  the non-branching program  $Pr_f^*$  and assuming  $N = \varepsilon^2$ . Then the function  $f$  can be implemented by such program  $Pr_f^{*(1)}$  that the inequality  $N_\varepsilon(Pr_f^{*(1)}) \leq 4N(N_\varepsilon(Pr_g) + N) \leq 4\varepsilon^2(\varepsilon + \varepsilon^2) = 4\varepsilon^3(1 + \varepsilon) \leq 4\varepsilon^3 \cdot 1,1 \leq 5\varepsilon^3$  (as  $\varepsilon \in (0, 1/960]$ ) is valid.

Now we take as  $N$  the quantity  $5\varepsilon^3$  and again use Theorem 2 (we'll make one more step of the iteration). Then the function  $f$  can be implemented by such program  $Pr_f^{*(2)}$  that the inequality  $N_\varepsilon(Pr_f^{*(2)}) \leq 5\varepsilon \cdot 5\varepsilon^3 = (5\varepsilon)^2\varepsilon^2$  holds.

It is easy to see that after  $m$  iterations the function  $f$  can be implemented by the program  $Pr_f^{*(m)}$  such that  $N_\varepsilon(Pr_f^{*(m)}) \leq (5\varepsilon)^m\varepsilon^2$ .

Let  $t$  be any natural number,  $t \geq 3$  (we remind that for  $t = 1$  and  $t = 2$  this theorem is true). We choose a natural number  $m$  for which the inequality  $m \geq (t - 2) \log_{5\varepsilon} \varepsilon$  is valid. Then  $(5\varepsilon)^m \leq \varepsilon^{t-2}$ , i.e.  $(5\varepsilon)^m\varepsilon^2 \leq \varepsilon^t$ . Hence,  $N_\varepsilon(Pr_f^{*(m)}) \leq \varepsilon^t$ .

Note that since  $m$  is the number of iterations it is reasonable to choose the smallest natural number  $m$  for which the inequality  $m \geq (t - 2) \log_{5\varepsilon} \varepsilon$  holds.

4) Let the basis  $B$  contains one of the functions  $\varphi_2(x_1, x_2, x_3)$ ,  $\varphi_3(x_1, x_2, x_3)$ , or  $\varphi_8(x_1, x_2, x_3)$ .

Then the function  $g(x_1, x_2, x_3, x_4)$ , equal to either the function  $(x_1 \vee x_2)(x_2 \vee \bar{x}_4)$  (Lemmas 3 and 9) or  $(\bar{x}_1 \vee \bar{x}_2)(x_2 \vee x_4)$  (Lemma 4), can be implemented by the program  $Pr_g$  with unreliability  $N_\varepsilon(Pr_g) \leq \varepsilon$  and  $\max\{\nu^1, \nu^0\} = \varepsilon$ .

By Theorem 1 the function  $f$  can be implemented by such circuit  $S$  that  $N_\varepsilon(S) \leq 3,11\varepsilon$  for all  $\varepsilon \in (0, 1/960]$ .

We apply Theorem 2 using the program  $Pr_g$  and taking the program  $S$  as the program  $R_f$  (see Remark ??). We construct the program  $Pr_f$  assuming  $N = 3,11\varepsilon$  and taking into account the conditions  $\max\{\nu^1, \nu^0\} = \varepsilon$  and  $N_\varepsilon(Pr_g) \leq \varepsilon$ . By Theorem 2 the non-branching program  $Pr_f$  implements the function  $f$  with unreliability  $N_\varepsilon(Pr_f) \leq \varepsilon + 52\varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .

Let's perform one more iteration step. Now we take for  $R_f$  already constructed program  $Pr_f$  unreliability of which for all  $\varepsilon \in (0, 1/960]$  satisfies the inequality  $N_\varepsilon(R_f) \leq \varepsilon + 52\varepsilon^2 \leq 1,0542\varepsilon$ . Let  $N = 1,0542\varepsilon$ . Use Theorem 2 taking  $\max\{\nu^1, \nu^0\} = \varepsilon$  and  $N_\varepsilon(Pr_g) \leq \varepsilon$ . Then the function  $f$  can be implemented by a non-branching program  $Pr_f$  such that  $N_\varepsilon(Pr_f) \leq \varepsilon + 9\varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .  $\square$

**Corollary 1.** *For faults of type 0 at the outputs of computational operators the following holds:*

1) *if the complete finite basis contains one of the functions  $\varphi_1(x_1, x_2, x_3)$ ,  $\varphi_4(x_1, x_2, x_3)$ ,  $\varphi_5(x_1, x_2, x_3)$ ,  $\varphi_6(x_1, x_2, x_3)$ ,  $\varphi_7(x_1, x_2, x_3)$ , or the function of the form  $x_1^{\alpha_1} \& x_2^{\alpha_2} (\alpha_1, \alpha_2 \in \{0, 1\})$ , then any Boolean function can be implemented by the non-branching program of arbitrarily high preassigned reliability for all  $\varepsilon \in (0, 1/960]$  (as  $\lim_{t \rightarrow \infty} \varepsilon^t = 0$ );*

2) *if the complete finite basis contains one of the functions  $\varphi_2(x_1, x_2, x_3)$ ,  $\varphi_3(x_1, x_2, x_3)$ ,  $\varphi_8(x_1, x_2, x_3)$ , or the function of the form  $x_1^{\alpha_1} \vee x_2^{\alpha_2} (\alpha_1, \alpha_2 \in \{0, 1\})$ , then any Boolean function can be implemented by the non-branching program with unreliability asymptotically no greater than  $\varepsilon$  for  $\varepsilon \rightarrow 0$ .*

Theorem 4 is proved similarly to Theorem 3 using the same Lemmas 2-9 but in the case of type 1 faults at the outputs of computational operators.

**Corollary 2.** *For faults of type 1 at the outputs of computational operators the following holds:*

1) *if the complete finite basis contains one of the functions  $\varphi_1(x_1, x_2, x_3)$ ,  $\varphi_4(x_1, x_2, x_3)$ ,  $\varphi_5(x_1, x_2, x_3)$ ,  $\varphi_6(x_1, x_2, x_3)$ ,  $\varphi_7(x_1, x_2, x_3)$ , or the function of the form  $x_1^{\alpha_1} \& x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), then any Boolean function can be implemented by the non-branching program with unreliability asymptotically no greater than  $\varepsilon$  for  $\varepsilon \rightarrow 0$ ;*

2) *if the complete finite basis contains one of the functions  $\varphi_2(x_1, x_2, x_3)$ ,  $\varphi_3(x_1, x_2, x_3)$ ,  $\varphi_8(x_1, x_2, x_3)$ , or the function of the form  $x_1^{\alpha_1} \vee x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), then any Boolean function can be implemented by the non-branching program of arbitrarily high preassigned reliability for all  $\varepsilon \in (0, 1/960]$  (as  $\lim_{t \rightarrow \infty} \varepsilon^t = 0$ ).*

#### 4. CONCLUSIONS

1. If the complete finite basis contains one of the functions  $\varphi_1(x_1, x_2, x_3)$ ,  $\varphi_4(x_1, x_2, x_3)$ ,  $\varphi_5(x_1, x_2, x_3)$ ,  $\varphi_6(x_1, x_2, x_3)$ ,  $\varphi_7(x_1, x_2, x_3)$ , or the function of the form  $x_1^{\alpha_1} \& x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), then
  - for faults of type 0 at the outputs of computational operators any Boolean function can be implemented by the non-branching program with a conditional stop operator of arbitrarily high preassigned reliability for all  $\varepsilon \in (0, 1/960]$ ;
  - for faults of type 1 at the outputs of computational operators any Boolean function can be implemented by the non-branching program with unreliability at most  $\varepsilon + 9\varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ .
2. If the complete finite basis contains one of the functions  $\varphi_2(x_1, x_2, x_3)$ ,  $\varphi_3(x_1, x_2, x_3)$ ,  $\varphi_8(x_1, x_2, x_3)$ , or the function of the form  $x_1^{\alpha_1} \vee x_2^{\alpha_2}$  ( $\alpha_1, \alpha_2 \in \{0, 1\}$ ), then
  - for faults of type 0 at the outputs of computational operators any Boolean function can be implemented by the non-branching program with unreliability at most  $\varepsilon + 9\varepsilon^2$  for all  $\varepsilon \in (0, 1/960]$ ;
  - for faults of type 1 at the outputs of computational operators any Boolean function can be implemented by the non-branching program of arbitrarily high preassigned reliability for all  $\varepsilon \in (0, 1/960]$ .

**Acknowledgments.** Partially supported by Russian Foundation for Basic Research, Grant 17-01-00451-a.

#### REFERENCES

1. A. V. Chashkin, "On the average time for the computation of the values of Boolean functions," *Discrete Anal. Issled. Oper. Ser. 1* **4** (1), 60–78 (1997) [in Russian].
2. A. V. Chashkin, "On the mean time for computing Boolean operators," *Discrete Anal. Issled. Oper. Ser. 1* **5** (1), 88–103 (1998) [in Russian].
3. J. von Neuman, "Probabilistic logics and the synthesis of reliable organisms from unreliable components," *Automata studies (AM-34)*, Ed. by C. Shannon and Mc. Carthy J., 34–98 (Princeton University Press, 1956).
4. S. I. Ortyukov, "On redundancy of Boolean function realization with circuits of unreliable elements," in *Proc. of Discrete Mathematics and its Applications Seminar*, 1987, Moscow, USSR, 166–168 [in Russian].
5. D. Uhlig, "Reliable networks from unreliable gates with almost minimal complexity," *Fundamentals of Computation Theory, FCT 1987, Lecture Notes in Computer Science* **278**, 462–469 (Springer, Berlin, Heidelberg, 1987).
6. S. V. Yablonskiy, "Asymptotically the best method of synthesis of reliable circuits from unreliable elements," *Banach Center* **7**, 11–19 (1982).

7. M. A. Alekhina, Synthesis, reliability and complexity of circuits from unreliable functional elements, D.sci. (Phys. Math.) diss., Penz. State Univ., Penza (2004) [in Russian].
8. V. V. Chugunova, Synthesis of asymptotically optimal reliable circuits under inverse failures at the element outputs, Ph.d. (Phys. Math.) diss., Penz. State Univ., Penza (2007) [in Russian].
9. A. V. Vasin, Asymptotically optimal circuits on reliability in complete final basis of three-input gates, Ph.d. (Phys. Math.) diss., Penz. State Univ., Penza (2010) [in Russian].
10. O. Y. Barsukova, The synthesis of reliable circuits which realizes binary and ternary logics functions, Ph.d. (Phys. Math.) diss., Penz. State Univ., Penza (2014) [in Russian].
11. S. M. Grabovskaya, Asymptotically optimal reliable non-branching programs, Ph.D. (Phys. Math.) diss., Penz. State Univ., Penza (2012) [in Russian].
12. M. A. Alekhina and S. M. Grabovskaya, "Reliability of nonbranching programs in an arbitrary complete finite basis," *Russ. Math.* **56** (2), 10–18 (2012).
13. M. A. Alekhina, "On reliability of circuits over an arbitrary complete finite basis under single-type constant faults at outputs of elements," *Discrete Math. Appl.* **22** (4), 383–391 (2012).
14. S. M. Grabovskaya, "On methods of raise the reliability of non-branching programs with a conditional stop operator," in *Proceeding Int. Conf. Problems of automation and control in technical systems, dedicated to the 70th anniversary of the Victory in the Great Patriotic War*, 2015, Penza, Russia, 339–342 [in Russian].
15. N. P. Red'kin, "On complete checking tests for circuits of functional elements," in *Mathematical Problems of Cybernetics* **2** (Nauka, Moscow, 1989), 198–222 [in Russian].
16. S. V. Yablonskii, *Introduction into Discrete Mathematics* (Nauka, Moscow, 2001) [in Russian].
17. S. M. Grabovskaya, "On reliability of non-branching programs in bases containing generalized disjunction," in *Proceeding XIX Int/ Sci. and Meth. Conf University Education (MKUO-2015), dedicated to the 70th anniversary of the Victory in the Great Patriotic War*, 2015, Penza, Russia, 218–219 [in Russian].
18. S. M. Grabovskaya, "On reliability of non-branching programs in bases containing generalized conjunction," in *Proceeding Int. Conf. Problems of automation and control in technical systems, dedicated to the 70th anniversary of the Victory in the Great Patriotic War*, 2015, Penza, Russia, 342–345 [in Russian].