International Journal of Applied Mathematics

Volume 34 No. 3 2021, 595-605
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
doi: http://dx.doi.org/10.12732/ijam.v34i3.13

SOLUTION OF SYSTEMS OF NONLINEAR BOOLEAN EQUATIONS

Aziz Ashurov

Tashkent University of Information Technologies
named after Muhammad al-Khwarizmi
Department of Algorithmization
and Mathematical Modelling
Str. Amir Temur, 108
Tashkent - 100200, UZBEKISTAN

Abstract: The paper proposes the methods and a software package for solving systems of Boolean equations, the propositions of which are formulas over an arbitrary basis, Zhegalkin polynomials consisting of nonlinear components and disjunctive normal forms.

The formation of Zhegalkin polynomials from formulas over an arbitrary basis, the finding of the maximum joint subsystems of logical equations and the search for intersections of the sets of solutions of several subsystems of the system of equations are considered.

AMS Subject Classification: 68R01, 03G05, 06E99, 90C09, 94D10
Key Words: recognition algorithms; object recognition; systems of logical equations; system of logical equations; discrete analysis; logical signs; Boolean algebraic equations; methods of logical algebra

1. Introduction

At present, a significant number of algorithms have been created for the synthesis of logical methods and experience has been accumulated to solve applied problems that arose in various fields of science and technology using these methods.

Received: March 18, 2021

One of the applications for the synthesis of methods is the problem of recognizing objects and phenomena, which can be reduced to solving systems of logical equations.

The paper proposes a SYSTEM complex program, developed based on the models of algorithms [1] and [2]. The SYSTEM program is intended for solving systems of nonlinear Boolean equations, the propositions of which are formulas over an arbitrary basis, Zhegalkin polynomials and disjunctive normal forms (d.n.f.). The program also solves the following tasks:

- transformation of Zhegalkin polynomials into disjunctive normal forms;
- formation of Zhegalkin polynomials from formulas over an arbitrary basis;
- finding the maximum joint subsystems of logical equations;
- search for intersections of solution sets of several subsystems of the system of equations;
- partition of the system of nonlinear equations.

Such problems arise in biology, medicine, military affairs, automation, control, experiment planning, etc. [1]

2. Mathematical statement

Let a system of logical equations be given

$$m = \begin{cases} f_1(x_1, x_2, \ldots, x_n) = \alpha_1, \\ f_2(x_1, x_2, \ldots, x_n) = \alpha_2, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots, \\ f_m(x_1, x_2, \ldots, x_n) = \alpha_m, \end{cases} \tag{1}$$

where each function (proposition) f_i, $i = 1, 2, \ldots, m$, can be implemented by formulas over M_1, M_2, M_3 and M_4 bases:

$$\begin{aligned} m_1 &= \left\{ \begin{array}{l} \rceil x, x_1 \wedge x_2, x_1 \vee x_2, x_1 \to x_2, x_1 \sim x_2 \\ x_1 + x_2, x_1/x_2, 1 \end{array} \right\}, \\ m_2 &= \{x_1 + x_2, x_1 \wedge x_2, 1\}, \\ m_3 &= \{x_1 \wedge x_2, x_1 \vee x_2, \rceil x\}, \\ m_4 &= \{x_1 + x_2, 1\}. \end{aligned}$$

The task is to find the class $\left\{\tilde{\beta}\right\}$ of binary sets $\tilde{\beta} = (\beta_1, \beta_2, \ldots, \beta_n)$ for which $f_i\left(\tilde{\beta}\right) = \alpha_i$, $i = 1, 2, \ldots, m$. To solve this problem, the SYSTEM uses the following blocks.

Block - 1.

Formation of Zhegalkin polynomials from formulas over an arbitrary basis. Consider proposition $n(x_1, x_2, \ldots, x_n)$ over the basis

$$m_1 = \left\{ \begin{array}{l} (x_1 + x_2) \bmod 2, x_1 \sim x_2, x_1 \to x_2, x_1 \wedge x_2 \\ x_1 \vee x_2, x_1/x_2, \neg x, 1 \end{array} \right\}.$$

Let

$$n_1 = A_1 + A_2 + \ldots + A_l \quad (l \geq 1)\,,$$

where A_i, $i = 1, 2, \ldots, l$ are formulas over

$$m_1 = \{x_1 \vee x_2, x_1 \wedge x_2, \rceil x, x_1 \to x_2, x_1/x_2, x_1 \sim x_2\}$$

basis.

First step.

We bring the proposition A_i to Zhegalkin polynomial A_1^n by applying the transformations:

$$\begin{array}{c} \overline{A} = A + 1, \\ A \vee B = A \wedge B + A + B, \\ A \to B = A \wedge B + A + 1, \\ A \sim B = A + B + 1, \\ A/B = A \wedge B + 1, \end{array} \tag{2}$$

where A and B are the elementary conjunctions (e.c.) and simplifications of the form:

$$\begin{array}{l} A + A = 0, \;\; A + O = A, \\ A \vee A = A, \;\; A \wedge O = O, \\ A \wedge 1 = A, \;\; 1 + 1 = O. \end{array} \tag{3}$$

Let $i - 1$ steps of induction be fulfilled and Zhegalkin polynomial be constructed.

***i* - step.**

We reduce the proposition A_i to Zhegalkin polynomial A_i^n by applying transformations (2) and simplifications (3).

Form the expression n_i:

$$n_i' = A_i^n + n_{i-1}'.$$

The result is Zhegalkin polynomial n', equivalent to the formula n.

Block - 2.

Transformation of Zhegalkin polynomials in the reduced d.n.f. A proposition $n(x_1, x_2, \dots, x_n)$ over m_2 basis is given:

$$n = A_1 + A_2 + \ldots + A_t, \tag{4}$$

where A_i are the elementary conjunctions.

The task is to transform a proposition n over m_2 basis into d.n.f. Let t be the number of elementary terms in Zhegalkin polynomial n.

Lemma 1. *The number of elementary conjunctions in d.n.f. obtained from Zhegalkin polynomial* (4) *is* 2^{t-1} *and has an expansion; if* $t = 2k+1$, *then*

$$\begin{gathered}\sum_{i=1}^{t} A_i = \bigwedge_{i=1}^{t} A_i \bigvee_{j=1}^{t-1} \bigvee_{i=j}^{t} \left(B_{ji}\overline{A}_j A_i\right) \vee \ldots \vee \\ \vee \bigvee_{j=1}^{t-k_1+1} \bigvee_{l=j+1}^{t-k_1+2} \cdots \bigvee_{i=\tau+1}^{t} \left(B_{jl} \ldots \overline{A}_j \overline{A}_l \ldots \overline{A}_i\right),\end{gathered} \tag{5}$$

at $t = 2k$,

$$\begin{gathered}\sum_{i=1}^{t} A_i = \bigvee_{i=1}^{t} \left(B_i\overline{A}_i\right) \vee \bigvee^{t=2} \bigvee^{t-1} \bigvee^{t} \left(B_{ijl}\overline{A}_i\overline{A}_j\overline{A}_l\right) \vee \ldots \\ \ldots \bigvee_{i=1}^{t-k_1+1} \bigvee_{i=1}^{t-k_1+2} \cdots \bigvee_{Q=l+1}^{t} \left(B_{ij\ldots Q}\overline{A}_i\overline{A}_j \ldots \overline{A}_Q\right),\end{gathered} \tag{6}$$

where k_1 *is the number of elementary conjunctions with negation.*

$B_{ji\ldots l}$ is the product of elementary conjunctions A_k, $k = 1, 2, \ldots, n$ without $A_j, A_i, \ldots, A_l$. Using the assertion of the lemma, we proceed to formula (5) or (6) over $m_3 = \{x_1 \vee x_2, x_1 \wedge x_2, \rceil x\}$ basis. Then applying the identities:

$$\begin{gathered}\overline{A_1 \wedge A_2 \wedge \ldots \wedge A_l} = \overline{A}_1 \vee \overline{A}_2 \vee \ldots \vee \overline{A}_l, \\ \overline{A_1 \wedge A_2 \wedge \ldots \wedge A_l} = \overline{A}_1 \wedge \overline{A}_2 \wedge \ldots \wedge \overline{A}_l, \\ A \wedge (A_1 \vee A_2 \vee \ldots \vee A_l) = A \wedge A_1 \vee A \wedge A_2 \vee \ldots \vee A \wedge A_l,\end{gathered}$$

and

$$\begin{gathered}\overline{A} \wedge A = 0, \;\; A \wedge 1 = A, \;\; O \vee A = A, \\ A \wedge O = O, \;\; A \wedge A = A, \;\; A \vee A = A,\end{gathered}$$

pass to d.n.f.

Block - 3.

Solution of a joint system given in the form of a d.n.f. The algorithm was given in [1].

Block - 4.

Intersection of sets of solutions to subsystems of equations. Let $L_1, L_2, \ldots, L_t$ be the solution sets of subsystems $m'_1, m'_2, \ldots, m'_t$ of systems m, respectively [2, 3]. The task is to find the set L: $L = \bigcap_{i=1}^{t} L_i$

Construct the algorithm in stages:

- the sets $L_1, L_2, \ldots, L_t$ are arranged in ascending order of cardinality. Let $|L_{i_1}| \leq \ldots \leq |L_{i_t}|$, where $|M|$ is the cardinality of set M;
- compute the sets $M_L = \bigcap_{j=1}^{t} L_{ij}$ by induction in t, $t = 2, \ldots, T$.

The first step of induction.

Find the intersection $M_2 = L_{i_1} \bigcap L_{i_2}$. Let M_{t-1} be constructed.

***i* - th step of induction.**

Construct $M_t = M_{t-1} \bigcap L_{i_t}$.

Thus, we obtain the intersection of the solution sets for subsystem $m'_1, m'_2, \ldots, m'_t$ of system m.

Block - 5.

Solution of systems of nonlinear logical equations. The algorithm was given in [4].

Block - 6.

Partitioning the system. Consider system (1) in m_2 basis. The task is to split the system m into two groups [5]:

- subsystem of linear equations;

- subsystem of nonlinear equations.

The algorithm consists of one stage. We form subsystem (2) by separating equations containing at least one e.c., the rank of which is two. The set of such equations gives subsystem (2). The set of remaining equations forms subsystem (1).

Block - 7.

Finding the maximum joint subsystems of systems of logical equations [6]. Let

$$\begin{cases} f_1 = u'_1 \vee u'_2 \vee \ldots \vee u'_{k_1} = 1, \\ f_2 = u''_1 \vee u''_2 \vee \ldots \vee u''_{k_2} = 1, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots, \\ f_m = u_1^m \vee u_2^m \vee \ldots \vee u_{k_m}^m = 1 \end{cases}$$

be the system of Boolean equations, where u_i^j is the elementary conjunction $i = 1, 2, \ldots, k_i$, $j = 1, 2, \ldots, m$. Consider the matrix $\|a_{ij}\|_{m\times m}$

$$a_{ij} = \begin{cases} 1, & \text{if } f_i \cdot f_j \neq 0, \\ 0, & \text{if } f_i \cdot f_j = 0. \end{cases}$$

It is clear that the matrix $\|a_{ij}\|$ is symmetric with respect to the diagonal and a_{ii}, $i = 1, 2, \ldots, m$. To calculate the element a_{ij} in $\|a_{ij}\|$, multiply d.n.f. f_i and f_j. If e.c. $u \in f_i$ and $u_1 \in f_j$ exist and $u \cdot u_1 \neq 0$, then $a_{ij} = 1$, if not $a_{ij} = 0$. It is known that if the subsystem

$$\begin{cases} f_{i_1} = 1, \\ f_{i_2} = 1, \\ \ldots\ldots, \\ f_{i_k} = 1, \end{cases} \tag{7}$$

is joint then the matrix $\|a_{ij}\|_{k\times k}$ of the matrix $\|a_{ij}\|_{m\times m}$ is the unity matrix; in the case when the sub matrix $\|a_{i_t j_t}\|_{k\times k}$ contains at least one zero element, the subsystem (7) is inconsistent. An algorithm for identifying the maximum unit sub matrix is constructed inductively [7].

The first step.

Find one of the rows in $\|a_{ij}\|_{m\times m}$, containing the largest number of zeros. We do not consider the rows and columns whose numbers correspond to zero elements of this row. Let the following matrix be constructed, before the execution of the i-th step of the induction, $\left\|b_{ij}^{i-1}\right\|_{(m-i+1)\times(m-i+1)}$.

i-th step.

In the matrix, delete one of the rows that contains the maximum number of zeros and the columns that correspond to zero elements of this row. The algorithm terminates at the next -th step if $\left\|b_{ij}^{i-1}\right\|_{(m-p)\times(m-p)}$ is the unity matrix. Let

$$\begin{cases} f_{i_1} = 1, \\ \dots\dots, \\ f_{i_l} = 1. \end{cases} \quad (8)$$

The subsystem of equations corresponds to the maximum unit submatrix of the matrix $\|a_{ij}\|_{m\times m}$. Then, to find the maximum joint subsystem, all equations of system (8) are multiplied sequentially. The inconsistent equation is excluded from the system. Note that in the process of solving 1, various logical operations (gluing, absorption, generalized gluing, multiplication) are performed between elementary conjunction (e.c.) of the form:

$$k = x_{i_1}^{\sigma_1} \wedge \dots \wedge x_{i_k}^{\sigma_k}, \quad 1 \le k \le n$$
$$\sigma_i \in \{0,1\}, \quad i = 1, 2, \dots, n.$$

Consider the method for representing e.c. by decimal codes (triples) and establishing the rules of Boolean algebra over these codes. The e.c. is assumed in correspondence to the sets $\tilde{\alpha} = (\alpha_1, \alpha_2, \dots, \tilde{\alpha}_n)$, $\tilde{\beta} = (\beta_1, \beta_2, \dots, \beta_n)$ and $\tilde{\gamma} = (\gamma_1, \gamma_2, \dots, \gamma_n)$:

$$\alpha_j = \begin{cases} 1, & \text{if } \ j \in \{i_1, i_2, \dots, i_k\} \ \text{ and } \ \sigma_j = 1, \\ 0, & \text{if not,} \end{cases}$$
$$\beta_j = \begin{cases} 1, & \text{if } \ j \in \{i_1, i_2, \dots, i_k\} \ \text{ and } \ \sigma_j = 0, \\ 0, & \text{if not,} \end{cases}$$
$$\gamma_j = \begin{cases} 1, & \text{if } \ j \notin \{i_1, i_2, \dots, i_k\} \\ 0, & \text{if not,} \end{cases}$$
$$j = 1, 2, \dots, n.$$

Let

$$a = \sum_{i=1}^{n} \gamma_i 2^{n-i}, \ \ b = \sum_{i=1}^{n} \alpha_i 2^{n-i}, \ \ c = \sum_{i=1}^{n} \beta_i 2^{n-i}.$$

Let us call the triple (a, b, c) the e.c. k. Introduce an integer positive function $\mathfrak{N}(x, y)$ of decimal positive arguments x, y that performs bit wise multiplication of binary representations of the arguments. Let us assume that (a_i, b_i, c_i) are the e.c. codes, respectively. The study in [8] showed that:

- E.c. k_i, k_j are glued together if $a_i = a_j$, $|b_i - b_j| = 2^P$ and $\left|N_{b_i} - N_{b_j}\right| = 1$, where P is a positive integer $(0 \le P \le N-1)$ and N_b is the norm of the set $(b_1, b_2, \ldots, b_n)$, $b = \sum\limits_{i=1}^{n} \beta_i 2^{n-i}$.

- E.c. k_i absorbs e.c. k_j, if $\mathfrak{N}(b_i, b_j) = b_i$ and $\mathfrak{N}(c_i, c_j) = c_i$.

- For E.c. k_i and k_j the operation of generalized gluing is applicable if $\mathfrak{N}(b_i, c_j) = 2^P$, $\mathfrak{N}(b_j, c_i) = 0$, $\mathfrak{N}(b_i, c_j) = 0$, $\mathfrak{N}(b_j, c_i) = 2^P$, where P is a positive integer $(0 \le P \le n-1)$.

- E.c. k_i and are not orthogonal, that is, $k_i \cdot k_j \neq 0$ if and only if $\mathfrak{N}(b_i, c_j) = \mathfrak{N}(b_j, c_i) = 0$.

The algorithms for solving systems (1) use the representations of e.c. by decimal codes and the rules of Boolean algebra are performed on these codes in accordance with assertions [9].

3. General structure of the program

The SYSTEM program consists of blocks $ABDV1, \ldots, ABDV7$ and a control program:

- $ABDV1$ - serves to transform a formula over an arbitrary basis into Zhegalkin polynomial;

- $ABDV2$ - performs the operation of Zhegalkin polynomial transformation into a d.n.f.;

- $ABDV3$ - solves joint systems of Boolean equations given in the form of d.n.f.;

- $ABDV4$ - finds general solutions for several subsystems;

- $ABDV5$ - solves a system of linear Boolean equations;

- $ABDV6$ - serves to partition a system of nonlinear Boolean equations into subsystems;

- $ABDV7$ - designed to find the maximum joint subsystems, the systems of logical equations.

The control program carries out data input and its primary processing, after which there is access to one of the blocks. The user determines the choice and sequence of the blocks by specifying the name-integer parameters PR and SOB. If a parameter identifying block is specified incorrectly, then a diagnostic error message is printed and the program terminates.

PR is a feature within a program that indicates the type of logical systems. The identifier formally takes values 1 and 4. At $PR = 1$, the program solves the system of linear logical equations and only the block $ABDV5$ participates in it. At $PR = 2$, the system is solved, the statements of which are given in the form of Zhegalkin polynomials; and the blocks $ABDV1, \ldots, ABDV7$ participate in it. At $PR = 3$, the system of Boolean equations (propositions of d.n.f.) is solved using the blocks $ABDV3$, $ABDV7$. And finally, at $PR = 4$, the system of Boolean equations is solved (the propositions of which are given in the form of formulas over an arbitrary basis) using blocks $ABDV1, \ldots, ABDV7$. Otherwise, the program terminates its run and displays the message "initial data is incorrect".

Let us denote the input feature by SOB, the identifier of which takes formal values 1 and 2 and indicates the compatibility of the system of logical equations.

If at $SOB = 2$, the $ABDV7$ blocks is not executed within the program, it is assumed that the system is compatible. At $SOB = 1$, the program is executed through the $ABDV7$ modulus. A simplified block diagram of the program is shown in the figure 1. The identifier $L11$ takes a value in the $ABDV6$ blocks and is used as a conditional parameter.

The input data are:

- an attribute of the object type (logical equations) PR;
- an attribute of SOB systems incompatibility;
- a number of variables N;
- a number of equations M;
- the systems of Boolean equations (an object) FK in analytical form.

After the end of the program, the following information is displayed:

- the numbers of incompatible equations of the system;
- the solutions of maximal joint subsystems in the form of: binary sets, reduced d.n.f. of the array of decimal representations e.c. of d.n.f.

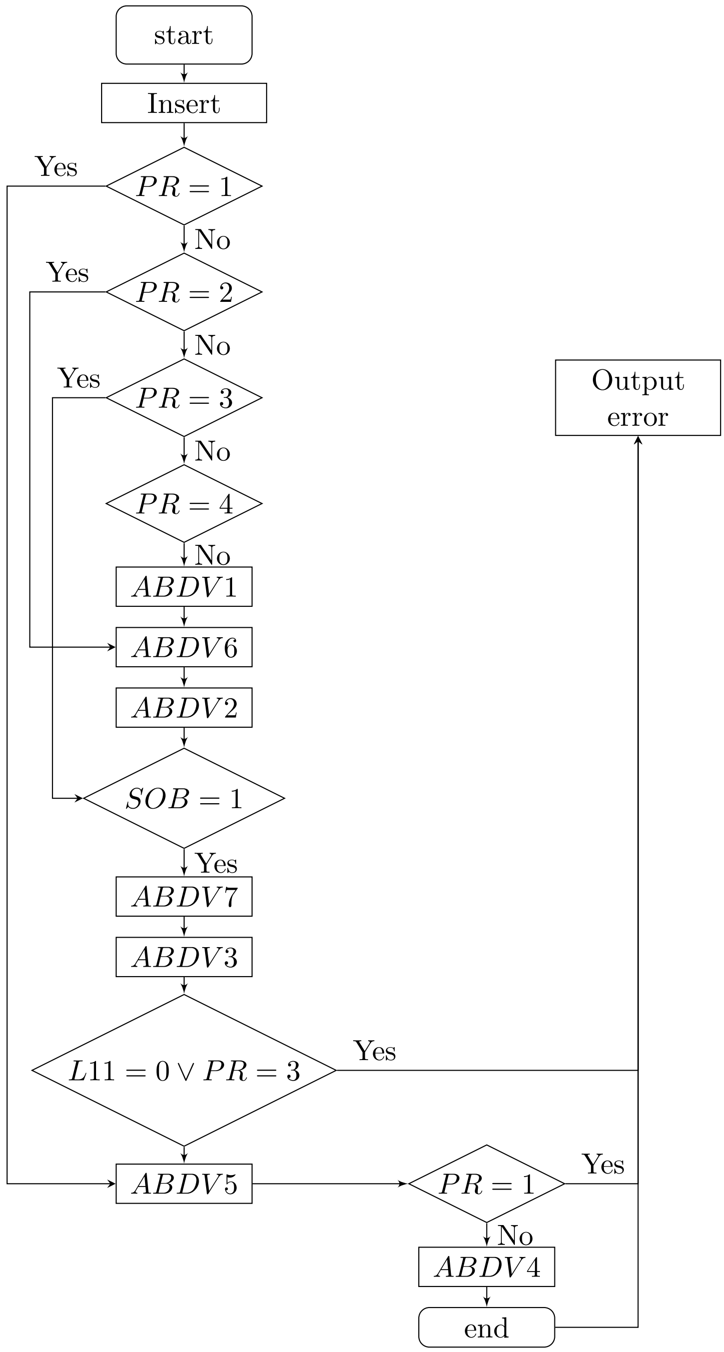

Figure 1: Program block diagram

Conclusion

The SYSTEM program was developed in this paper, designed to solve the systems of nonlinear Boolean equations, the propositions of which are formulas over an arbitrary basis, Zhegalkin polynomials and disjunctive normal forms. The program solves the following tasks:

- transformation of Zhegalkin polynomials into d.n.f.;
- formation of Zhegalkin polynomials from formulas over an arbitrary basis;
- finding the maximum joint subsystems of logical equations;
- search for intersections of solution sets of several subsystems of the system of equations;
- partition of the system of nonlinear equations.

References

[1] A.V. Kabulov, A.A. Bayzhumanov, Some issues of analytical transformations in minimization of logical functions, *Questions in Computational and Applied Mathematics, Tashkent, RISO AN UzSSR*, **65** (1981).

[2] A.V. Kabulov, A.A. Bayzhumanov, Representation of logical operations in the decimal system of calculus, *Algorithms, Tashkent, RISO AN UzSSR*, **45** (1981).

[3] A.V. Kabulov, A.A. Bayzhumanov, A new approach to solving systems of Boolean equations, *Proceedings of the X Conference of Young Scientists in the NPO Cybernetics of the Academy of Sciences of the Uzbek SSR*, **76** (1982).

[4] A.V. Kabulov, I.H. Normatov, About problems of decoding and searching for the maximum upper zero of discrete monotone functions, *Journal of Physics: Conference Series*, **1260** (2019), # 102006.

[5] A.V. Kabulov, E. Urunbaev, A.O. Ashurov, Logical method for constructing the optimal corrector of fuzzy heuristic algorithms, *2019 International Conference on Information Science and Communications Technologies (ICISCT)*, **1210** (2019), 1–4.

[6] A.V. Kabulov, I.H. Normatov, A.O. Ashurov, Computational methods of minimization of multiple functions, *Journal of Physics: Conference Series*, **1260** (2019), # 102007.

[7] A.V. Kabulov, I.H. Normatov, A. Karimov, Algorithmization control of complex systems based on functioning tables, *Journal of Physics: Conference Series*, **1441** (2020), # 012141.

[8] A.V. Kabulov, I.H. Normatov, E. Urunbaev, A.O. Ashurov, About the problem of minimal tests searching, *Advances in Mathematics: Scientific Journal*, **9** (2020), 10419–10430.

[9] A.V. Kabulov, I.H. Normatov, E. Urunbaev, A.O. Ashurov, Synthesis methods of optimal discrete corrective functions, *Advances in Mathematics: Scientific Journal*, **9** (2020), 6467–6482.