MSC 05C85 DOI: 10.14529/mmp160302

# AN INFERENCE ALGORITHM FOR MONOTONE BOOLEAN FUNCTIONS ASSOCIATED WITH UNDIRECTED GRAPHS

***D.N. Gainanov,*** Ural Federal University, Ekaterinburg, Russian Federation, damir.gainanov@gmail.com
***V.A. Rasskazova,*** Moscow Aviation Institute, Moscow, Russian Federation, 2265874@mail.ru

Boolean functions are a modelling tool useful in many applications; monotone Boolean functions make up an important class of these functions. For instance, monotone Boolean functions can be used for describing the structure of the feasible subsystems of an infeasible system of constraints, because feasibility is a monotone feature. In this paper we consider monotone Boolean functions (MBFs), associated with undirected graphs, whose upper zeros are defined as binary tuples for which the corresponding subgraph of the original undirected graphs is either the empty graph, or it has no edges.

For this class of MBFs, we present the settings of problems which are related to the search for upper zeros and maximal upper zeros of these functions. The notion of $k$-vertices and $(k,m)$-vertices in a graph is introduced. It is shown that for any $k$-vertices of the original graph there exists a maximal upper zero of an MBF associated with the graph, in which the component $x_i$ corresponding to this $k$-vertex takes the value 1.

Based on this statement, we construct an algorithm of searching for a maximal upper zero, for the class of MBFs under consideration, which allows one to find, under certain conditions, the solution to the problem of searching for a maximal upper zero, or to substantially reduce the dimension of the original problem.

The proposed algorithm was extended for the case of $(k,m)$-vertices. This extended algorithm allows one to fix a bound on the deviation of an upper zero of the MBF from the maximal upper zeros, in the sense of the number of units in these tuples. The algorithm has the complexity $O(n^2p)$, where $n$ is a number of vertices and $p$ is a number of edges of the original graph.

*Keywords: monotone Boolean function; upper zero of a monotone Boolean function; graph; algorithm of searching for maximal upper zeros of a monotone Boolean function.*

## Introduction

In a wide class of problems, infeasible systems of constraints occur naturally and become the research subject. A variety of such systems is treated in [1] by methods of combinatorial geometry and graph theory. The study of infeasible systems, whose constraints correspond to the vertices of undirected graphs, and the subsystems with two constraints are feasible if and only if the corresponding vertex pairs are edges of the graphs, is of special applied interest.

In this paper we associate with a graph a monotone Boolean function whose zeros correspond to the feasible subsystems of the initial infeasible system of constraints, in which any subsystem of infeasible system is feasible if and only if every pairs of constrains is also feasible.

The settings of Problems 1 and 2 in terms of inference of monotone Boolean functions and, more precisely, as the search for upper zeros and maximal upper zeros, make sense because such a setting allows one to use, for example, an algorithm of searching for upper zeros of monotone Boolean functions described in [1, 2]; see also [3–12], where the above–mentioned and similar algorithms from the family of *Find Border Algorithms* are discussed. In this context, the *border* means the union of the sets of all upper zeros and lower units of a monotone Boolean function. An extensive survey of the current state of the theory and practice of MBF inference is presented in [11, 13].

Problem 2 can also be solved by the algorithm proposed in [1]; among the upper zeros, we must find the maximal ones. In addition, an approximate algorithm, guided by the increasing collection of generated upper zeros, can be involved in research.

Let us turn to basic notions and problems.

## 1. Basic Notions and Problems

Let $[n] := \{1, \ldots, n\}$ denote the set of consecutive integers, and let $\mathbf{B}^n := \{0, 1\}^n$ denote the unit discrete $n$–dimensional cube. If $\boldsymbol{x} := (x_1, \ldots, x_n) \in \mathbf{B}^n$, then $\mathrm{supp}(\boldsymbol{x}) := \{i \in [n] : \ x_i = 1\}$.

For binary tuples $\boldsymbol{x}$ and $\boldsymbol{x}'$, of length $n$, the ordering $\boldsymbol{x} \leq \boldsymbol{x}'$ in $\mathbf{B}^n$ by definition holds if and only if $x_i \leq x'_i$, for all $i \in [n]$.

If $\boldsymbol{\mathcal{X}} \subseteq \mathbf{B}^n$ is a set of tuples, then $\max_{\subseteq} \boldsymbol{\mathcal{X}}$ denotes the subset of maximal elements of $\boldsymbol{\mathcal{X}}$ with respect to the partial order on $\mathbf{B}^n$, and $\max_{|\cdot|} \boldsymbol{\mathcal{X}}$ denotes the subset of all tuples from $\boldsymbol{\mathcal{X}}$ that have the maximal number of unit components.

A *Boolean function* $\mathfrak{f} : \ \mathbf{B}^n \to \mathbf{B}$ is called *monotone* if the implication

$$\boldsymbol{x}, \boldsymbol{x}' \in \mathbf{B}^n \ , \ \boldsymbol{x} \leq \boldsymbol{x}' \quad \Longrightarrow \quad \mathfrak{f}(\boldsymbol{x}) \leq \mathfrak{f}(\boldsymbol{x}')$$

holds. A tuple $\boldsymbol{x} \in \mathbf{B}^n$ is called a *zero* (respectively, a *unit*) of $\mathfrak{f}$ if $\mathfrak{f}(\boldsymbol{x}) = 0$ (respectively, $\mathfrak{f}(\boldsymbol{x}) = 1$).

A tuple $\boldsymbol{x} \in \mathbf{B}^n$ is called an *upper zero* of the monotone Boolean function $\mathfrak{f} : \mathbf{B}^n \to \mathbf{B}$ if $\mathfrak{f}(\boldsymbol{x}) = 0$, and $\mathfrak{f}(\boldsymbol{x}') = 1$ holds for all $\boldsymbol{x}' \in \mathbf{B}^n$ such that $\boldsymbol{x} < \boldsymbol{x}'$; dually, a tuple $\boldsymbol{x} \in \mathbf{B}^n$ is called a *lower unit* of the function $\mathfrak{f}$ if $\mathfrak{f}(\boldsymbol{x}) = 1$, and $\mathfrak{f}(\boldsymbol{x}') = 0$ holds for all $\boldsymbol{x}' \in \mathbf{B}^n$ such that $\boldsymbol{x}' < \boldsymbol{x}$. A tuple $\boldsymbol{x} \in \mathbf{B^n}$ is called a *maximal upper zero* of the MBF $\mathfrak{f}$ if $|\mathrm{supp}(\boldsymbol{x})| = \max\{\mathrm{supp}(\boldsymbol{x}') : \ \boldsymbol{x}' \in \max_{\subseteq} \mathfrak{f}^{-1}(0)\}$.

Let a simple undirected graph $\mathbf{G} := (V(\mathbf{G}), \mathcal{E}(\mathbf{G}))$ be given, with the vertex set $V(\mathbf{G}) := \{v_1, \ldots, v_n\}$ and the edge family $\mathcal{E}(\mathbf{G}) := \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_p\}$. If $U \subset V(\mathbf{G})$, then $\mathbf{G}\langle U\rangle$ denotes the induced subgraph of the graph $\mathbf{G}$, on the vertex set $U$. For a vertex $v \in V(\mathbf{G})$, $\mathcal{N}(v) \subset V(\mathbf{G})$ denotes the neighborhood of the vertex $v$ in the graph $\mathbf{G}$. For a subset of vertices $U \subseteq V(\mathbf{G})$, by $\binom{U}{2}$ denote the family of all unordered 2-subsets of the set $U$.

Denote by $\#(\cdot)$ the number of sets in a family, and by $|\cdot|$ the cardinality of a set.

Consider the monotone Boolean function $\mathfrak{f}_{\mathbf{G}} : \ \mathbf{B}^n \to \mathbf{B}$ whose set of *units* $\mathfrak{f}_{\mathbf{G}}^{-1}(1)$ is defined as following:

$$\mathfrak{f}_{\mathbf{G}}(\boldsymbol{x}) := 1 \quad \Longleftrightarrow \quad \#\left(\mathcal{E}(\mathbf{G}) \cap \tbinom{\{v_i \in V(\mathbf{G}): \ i \in \mathrm{supp}(\boldsymbol{x})\}}{2}\right) \geq 1 \ ; \tag{1}$$

in other words, we suppose $\mathfrak{f}_{\mathbf{G}}(\boldsymbol{x}) := 1$ if and only if the induced subgraph $\mathbf{G}\langle\{v_i \in V(\mathbf{G}) : \ i \in \mathrm{supp}(\boldsymbol{x})\}\rangle$ has at least one edge.

Another monotone Boolean function $\mathfrak{g}_{\mathbf{G}} : \mathbf{B}^n \to \mathbf{B}$, which is naturally associated with the graph $\mathbf{G}$, is defined by the set of its *zeros* $\mathfrak{g}_{\mathbf{G}}^{-1}(0)$ as following:

$$\mathfrak{g}_{\mathbf{G}}(\boldsymbol{x}) := 0 \quad \Longleftrightarrow \quad \text{subgraph } \mathbf{G}\langle\{v_i \in V(\mathbf{G}) : \ i \in \operatorname{supp}(\boldsymbol{x})\}\rangle \text{ is complete ;} \tag{2}$$

we relate to the complete graphs, the empty graph $\mathbf{G}\langle\emptyset\rangle$ and the isolated vertices $\mathbf{G}\langle\{v_i\}\rangle$, $v_i \in V(\mathbf{G})$.

The graph-theoretic construction that establishes the connection between MBFs from (1) and (2) is the complement of the graph. The *complement* $\overline{\mathbf{G}}$ *of the graph* $\mathbf{G}$ by definition has the vertex set $V(\mathbf{G})$ and the edge family $\binom{V(\mathbf{G})}{2} - \mathcal{E}(\mathbf{G})$. Definitions (1) and (2) imply the following useful identities:

$$\mathfrak{f}_{\mathbf{G}} = \mathfrak{g}_{\overline{\mathbf{G}}} \ , \quad \mathfrak{f}_{\overline{\mathbf{G}}} = \mathfrak{g}_{\mathbf{G}} \ .$$

**Problem 1.** *For the function* $\mathfrak{f}_{\mathbf{G}}$ *defined in* (1), *to find the set*

$$\max_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$$

*of its upper zeros.*

**Problem 2.** *For the function* $\mathfrak{f}_{\mathbf{G}}$, *to find the set*

$$\max_{|\cdot|} \max_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$$

*of its maximal upper zeros.*

## 2. An Algorithm for Finding a Maximal Upper Zero of a Monotone Boolean Function Associated with an Undirected Graph

Consider Problem 2, for graphs from a certain class in more detail.

**Proposition 1.** *Let* $v_i \in V(\mathbf{G})$ *be a vertex of a graph* $\mathbf{G} := (V(\mathbf{G}), \mathcal{E}(\mathbf{G}))$, *such that for its neighborhood* $\mathcal{N}(v_i)$ *the induced subgraph* $\mathbf{G}\langle\mathcal{N}(v_i)\rangle$ *of the graph* $\mathbf{G}$ *is complete. Then there exists a maximal upper zero* $\boldsymbol{x}' \in \max_{|\cdot|} \max_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$ *of the function* $\mathfrak{f}_{\mathbf{G}}$ *such that* $x'_i = 1$.

*Proof.* Consider an arbitrary maximal upper zero $\boldsymbol{x} \in \max_{|\cdot|} \max_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$ of the function $\mathfrak{f}_{\mathbf{G}}$, and associate with this zero the index set $I := \{s \in [n] : \ v_s \in \mathcal{N}(v_i)\}$. It is easy to see that among the elements of the set $I \dot\cup \{i\}$ there is at least one index $j$ such that $x_j = 1$, because otherwise we could find a tuple $\boldsymbol{x}' \in \mathbf{B}^n$ such that $x'_i = 1$ and $x'_s = x_s$ for all indices $s \in [n] - \{i\}$. Thus, because of $\mathfrak{f}_{\mathbf{G}}(\boldsymbol{x}) = 0$, and by the assumption that $x_s = 0$ for all $s \in I$, the definition of the function $\mathfrak{f}_{\mathbf{G}}$ implies that $\mathfrak{f}_{\mathbf{G}}(\boldsymbol{x}') = 0$. This contradicts the maximality of the upper zero $\boldsymbol{x}$, because we obtain the strict inclusion $\operatorname{supp}(\boldsymbol{x}') \supsetneqq \operatorname{supp}(\boldsymbol{x})$ and $\mathfrak{f}_{\mathbf{G}}(\boldsymbol{x}') = \mathfrak{f}_{\mathbf{G}}(\boldsymbol{x}) = 0$. Now let us consider the two possible cases. If $x_i = 1$, then we are done. If $x_i = 0$ and $x_s = 1$ for some index $s \in I$, then for the tuple $\boldsymbol{x}$ one can find the tuple $\boldsymbol{x}' \in \mathbf{B}^n$ (by the rule: $x'_j := x_j$ for all $j \in [n] - \{i, s\}$, $x'_i := 1$, and $x'_s := 0$), which is an upper zero of the function $\mathfrak{f}_{\mathbf{G}}$, in view of the completeness of the induced subgraph $\mathbf{G}\langle\mathcal{N}(v_i)\rangle$, and $|\operatorname{supp}(\boldsymbol{x}')| = |\operatorname{supp}(\boldsymbol{x})|$; we thus obtained a maximal upper zero $\boldsymbol{x}'$ of the function $\mathfrak{f}_{\mathbf{G}}$ such that $x'_i = 1$, as it was to be proved. □

**Definition 1.** *For an integer $k \in [n-1]$, a vertex $v \in V(\mathbf{G})$ of the graph $\mathbf{G} := (V(\mathbf{G}), \mathcal{E}(\mathbf{G}))$ is called a $k$-*vertex*, if $|\mathcal{N}(v)| = k$ and the induced subgraph $\mathbf{G}\langle\mathcal{N}(v)\rangle$ of the graph $\mathbf{G}$ is complete.*

**Definition 2.** *For integers $k, m \in [n-1]$, a vertex $v \in V(\mathbf{G})$ of the graph $\mathbf{G} := (V(\mathbf{G}), \mathcal{E}(\mathbf{G}))$ is called a $(k,m)$-*vertex*, if $k = |\mathcal{N}(v)|$ and $m = \binom{k}{2} - \#\big(\mathcal{E}(\mathbf{G}) \cap \binom{\mathcal{N}(v)}{2}\big)$.*

A $(k,m)$-vertex $v \in V(\mathbf{G})$ of the graph $\mathbf{G} := (V(\mathbf{G}), \mathcal{E}(\mathbf{G}))$ is its $k$-vertex when $m = 0$.

On the basis of Proposition 1 one can propose an efficient recursive algorithm for solving Problem 2, which finishes its work either by the construction of a maximal upper zero of the function $\mathfrak{f}_{\mathbf{G}}$, or by the reduction of Problem 2 for the function $\mathfrak{f}_{\mathbf{G}}$ to the new Problem 2 for a function $\mathfrak{f}_{\mathbf{G}'}$, where $\mathbf{G}' \subset \mathbf{G}$, that is, by the decrease of the dimension of the problem to be solved.

Given a vertex $v \in V_0 \subseteq V(\mathbf{G})$, denote by $\mathcal{N}(v, V_0) \subset V_0$ the neighborhood of the vertex $v$ in the induced subgraph $\mathbf{G}\langle V_0\rangle$.

**Algorithm 1.** Algorithm $A(\mathbf{G}, V_0)$ for finding a maximal upper zero $\boldsymbol{x} := (x_1, \ldots, x_n) \in \mathbf{B}^n$ of the function $\mathfrak{f}_{\mathbf{G}}$

`Input data:` $\mathbf{G}, V_0$

`Output data:` $V_0, \boldsymbol{x}$

1: $x_i = 0,\ i \in [n],\ v_i \in V_0$

2: `for each` $v_i \in V_0$ `do`

3: `if` $v_i$ is a $|\mathcal{N}(v_i, V_0)|$-vertex in the subgraph $\mathbf{G}\langle V_0\rangle$ `then`

4: $x_i \leftarrow 1$

$V_0 \leftarrow V_0 - (\{v_i\} \,\dot\cup\, \mathcal{N}(v_i, V_0))$

$A(\mathbf{G}, V_0)$

`end of condition`

`end of loop`

If at the end of the work of *Algorithm* 1 we get $V_0 = \emptyset$, then, according to Proposition 1, the resulting tuple $\boldsymbol{x} \in \mathbf{B}^n$ is a maximal upper zero of the function $\mathfrak{f}_{\mathbf{G}}$.

However, if at the end of the work of *Algorithm* 1 we have $V_0 \neq \emptyset$, then for all vertices of the graph $\mathbf{G}\langle V - V_0\rangle$ we determined the values of some components $x_i$ such that there exists a maximal upper zero $\boldsymbol{x}'$ of the function $\mathfrak{f}_{\mathbf{G}}$ with precisely the same values for these components, that is, $x'_i = x_i$; and yet we achieve the decrease of the dimension of the problem from $|V|$ to $|V_0|$.

**Lemma 1.** *Let two graphs $\mathbf{G}_1 := (V, \mathcal{E}(\mathbf{G}_1))$ and $\mathbf{G}_2 := (V, \mathcal{E}(\mathbf{G}_2))$ be given, with the same vertex set $V$, and*

$$\mathcal{E}(\mathbf{G}_1) \subseteq \mathcal{E}(\mathbf{G}_2)\ .$$

*Then*

$$\max_{|\cdot|} \max_{\subseteq} \mathfrak{f}^{-1}_{\mathbf{G}_2}(0) \subseteq \max_{\subseteq} \mathfrak{f}^{-1}_{\mathbf{G}_2}(0) \subseteq \mathfrak{f}^{-1}_{\mathbf{G}_2}(0) \subseteq \mathfrak{f}^{-1}_{\mathbf{G}_1}(0)\ .$$

*Proof.* It is clear that $\max_{|\cdot|} \max_{\subseteq} \mathfrak{f}^{-1}_{\mathbf{G}_2}(0) \subseteq \max_{\subseteq} \mathfrak{f}^{-1}_{\mathbf{G}_2}(0) \subseteq \mathfrak{f}^{-1}_{\mathbf{G}_2}(0)$.

Consider an arbitrary tuple $\boldsymbol{x} \in \mathbf{B}^n$ such that $\boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}_2}^{-1}(0)$. By the definition of the set of zeros $\mathfrak{f}_{\mathbf{G}_2}^{-1}(0)$ of the MBF $\mathfrak{f}_{\mathbf{G}_2}$, we have:

$$\#\left(\mathcal{E}(\mathbf{G}_2) \cap \binom{\{v_i\colon\, i \in \operatorname{supp}(\boldsymbol{x})\}}{2}\right) = 0 \; .$$

By the hypothesis of the lemma, we have $\mathcal{E}(\mathbf{G}_1) \subseteq \mathcal{E}(\mathbf{G}_2)$ and $V(\mathbf{G}_1) = V(\mathbf{G}_2)$; as a consequence,

$$\#\left(\mathcal{E}(\mathbf{G}_1) \cap \binom{\{v_i\colon\, i \in \operatorname{supp}(\boldsymbol{x})\}}{2}\right) = 0 \; , \quad \forall \boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}_2}^{-1}(0) \; ,$$

and

$$\boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}_1}^{-1}(0) \; . \tag{3}$$

Then for any tuples $\boldsymbol{x} \in \mathbf{B}^n$ such that $\boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}_2}^{-1}(0)$, inclusion (3) holds, that is,

$$\mathfrak{f}_{\mathbf{G}_2}^{-1}(0) \subseteq \mathfrak{f}_{\mathbf{G}_1}^{-1}(0) \; ,$$

as it was to be proved.

□

It should be mentioned that

$$\max_{\subseteq} \mathfrak{f}_{\mathbf{G}_2}^{-1}(0) \not\subseteq \max_{\subseteq} \mathfrak{f}_{\mathbf{G}_1}^{-1}(0) \; . \tag{4}$$

Indeed, consider the graphs

$$\begin{aligned} \mathbf{G}_1 &:= (V(\mathbf{G}_1),\ \mathcal{E}(\mathbf{G}_1)) = (V, \emptyset) \; , \\ \mathbf{G}_2 &:= (V(\mathbf{G}_2),\ \mathcal{E}(\mathbf{G}_2)) = \left(V, \tbinom{V}{2}\right) \; , \end{aligned}$$

for which we have $V(\mathbf{G}_1) = V(\mathbf{G}_2)$ and $\mathcal{E}(\mathbf{G}_1) \subseteq \mathcal{E}(\mathbf{G}_2)$. The graph $\mathbf{G}_1$ has no edges, therefore, the set of upper zeros of the function $\mathfrak{f}_{\mathbf{G}_1}$ consists of the unique tuple

$$\boldsymbol{x} := (1, 1, \ldots, 1) \; .$$

The graph $\mathbf{G}_2$ is complete; thus, the set of upper zeros of the function $\mathfrak{f}_{\mathbf{G}_2}$ has the form:

$$\begin{aligned} \boldsymbol{x}^1 &:= (1, 0, \ldots, 0) \; , \\ \boldsymbol{x}^2 &:= (0, 1, \ldots, 0) \; , \\ &\vdots \\ \boldsymbol{x}^n &:= (0, 0, \ldots, 1) \; . \end{aligned}$$

Any tuple $\boldsymbol{x} \in \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}_2}^{-1}(0)$ is a zero of the function $\mathfrak{f}_{\mathbf{G}_1}$, that is,

$$\max_{\subseteq} \mathfrak{f}_{\mathbf{G}_2}^{-1}(0) \subseteq \mathfrak{f}_{\mathbf{G}_1}^{-1}(0) \; , \quad \max_{\subseteq} \mathfrak{f}_{\mathbf{G}_2}^{-1}(0) \not\subseteq \max_{\subseteq} \mathfrak{f}_{\mathbf{G}_1}^{-1}(0) \; ,$$

as Lemma 1 asserts; this justifies (4).

Let us define the quantity $\max_0 \mathfrak{f}_{\mathbf{G}} := |\operatorname{supp}(\boldsymbol{x})|$, where $\boldsymbol{x} \in \max\limits_{|\cdot|} \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$, that is the number of unit components in a maximal upper zero of the function $\mathfrak{f}_{\mathbf{G}}$.

**Corollary 1.** *Let* $\mathbf{G}_1 := (V, \mathcal{E}_1)$ *and* $\mathbf{G}_2 := (V, \mathcal{E}_2)$ *be graphs such that* $\mathcal{E}_1 \subseteq \mathcal{E}_2$. *Then*

$$\max\nolimits_0 \mathfrak{f}_{\mathbf{G}_1} \geq \max\nolimits_0 \mathfrak{f}_{\mathbf{G}_2} \ .$$

*Proof.* Let $\boldsymbol{x} \in \max\limits_{|\cdot|} \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}_2}^{-1}(0)$. According to Lemma 1, we have $\boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}_1}^{-1}(0)$.

By the definition of the maximal upper zeros of the function, for any tuple $\boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}_1}^{-1}(0)$ there exists a tuple $\boldsymbol{x}' \in \max\limits_{|\cdot|} \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}_1}^{-1}(0)$ such that $\boldsymbol{x}' \geq \boldsymbol{x}$. Then

$$\max\nolimits_0 \mathfrak{f}_{\mathbf{G}_1} = |\mathrm{supp}(\boldsymbol{x}')| \geq |\mathrm{supp}(\boldsymbol{x})| = \max\nolimits_0 \mathfrak{f}_{\mathbf{G}_2} \ ,$$

as it was to be proved. □

**Proposition 2.** *Let* $\mathbf{G} := (V(\mathbf{G}), \mathcal{E}(\mathbf{G}))$ *be a graph for which vertices* $v_i$ *and* $v_j$ *are not adjacent. Then*

$$\max\nolimits_0 \mathfrak{f}_{\mathbf{G}} \geq \max\nolimits_0 \mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}} \geq \max\nolimits_0 \mathfrak{f}_{\mathbf{G}} - 1 \ . \tag{5}$$

*Proof.* The inequality $\max_0 \mathfrak{f}_{\mathbf{G}} \geq \max_0 \mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}}$ follows from Corollary 1.

Let us prove the inequality $\max_0 \mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}} \geq \max_0 \mathfrak{f}_{\mathbf{G}} - 1$. Let $\boldsymbol{x} := (x_1, \ldots, x_n)$ be a maximal upper zero of the function $\mathfrak{f}_{\mathbf{G}}$.

**Case** 1. Suppose that $x_i = 0$ and $x_j = 0$. Then $\boldsymbol{x}$ is clearly a zero of the function $\mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}}$, and it is a maximal upper zero, because otherwise we would obtain, by definition, that there exists a maximal upper zero $\boldsymbol{x}'$ of the function $\mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}}$ such that $\boldsymbol{x}' > \boldsymbol{x}$ and $|\mathrm{supp}(\boldsymbol{x}')| > |\mathrm{supp}(\boldsymbol{x})|$. According to Lemma 1, we obtain that $\boldsymbol{x}'$ is a zero of the function $\mathfrak{f}_{\mathbf{G}}$, but this contradicts the maximality of $\boldsymbol{x}$.

Thus, in this case, we have:

$$\max\nolimits_0 \mathfrak{f}_{\mathbf{G}} = \max\nolimits_0 \mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}} \geq \max\nolimits_0 \mathfrak{f}_{\mathbf{G}} - 1 \ .$$

**Case** 2. Suppose that $x_i = 1$ and $x_j = 0$. If the edge $(v_i, v_j)$ is added, then the tuple $\boldsymbol{x}$ is again a zero of the function $\mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}}$ and, as it was shown above, it is also a maximal upper zero of the function $\mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}}$.

**Case** 3. Suppose that $x_i = 1$ and $x_j = 1$. If the edge $(v_i, v_j)$ is added, then we obtain that $\boldsymbol{x}$ is not a zero of the function $\mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}}$. In this case, we can find a tuple $\boldsymbol{x}'$ for which $x'_s = x_s$ for all $s \in [n] - \{i\}$, and $x'_i = 0$. The tuple $\boldsymbol{x}'$ will be a zero of the function $\mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}}$. Moreover, by construction,

$$|\mathrm{supp}(\boldsymbol{x}')| = |\mathrm{supp}(\boldsymbol{x})| - 1 \ .$$

By the definition of the maximal upper zeros of the function, we have:

$$\max\nolimits_0 \mathfrak{f}_{\mathbf{G} \cup \{(v_i, v_j)\}} \geq |\mathrm{supp}(\boldsymbol{x}')| = |\mathrm{supp}(\boldsymbol{x})| - 1 = \max\nolimits_0 \mathfrak{f}_{\mathbf{G}} - 1 \ ,$$

as it was to be proved. □

**Corollary 2.** *For a graph* $\mathbf{G} := (V(\mathbf{G}), \mathcal{E}(\mathbf{G}))$, *let* $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_t\} \subset \binom{V(\mathbf{G})}{2} - \mathcal{E}(\mathbf{G})$ *be a subfamily of* $t$ *vertex pairs that are not edges of the graph* $\mathbf{G}$.

*Then*

$$\max\nolimits_0 \mathfrak{f}_{\mathbf{G} \cup \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_t\}} \geq \max\nolimits_0 \mathfrak{f}_{\mathbf{G}} - t \ .$$

*Proof.* It suffices to apply Proposition 2, $t$ times, to the graph $\mathbf{G}$. □

On the basis of Proposition 2, one can modify *Algorithm* 1 in such a way that the work of the algorithm will continue until the set of remaining vertices $V_0$ becomes empty and, besides, a zero $\boldsymbol{x}$ of the function $\mathfrak{f}_{\mathbf{G}}$ will be found, for which, at the same time, we will calculate the estimate $(\max_0 \mathfrak{f}_{\mathbf{G}} - |\mathrm{supp}(\boldsymbol{x})|)$ of the deviation of the number of unit components in the resulting tuple $\boldsymbol{x}$ from the number of unit components in a maximal upper zero of the function $\mathfrak{f}_{\mathbf{G}}$.

**Algorithm 2.** Algorithm $A_m(\mathbf{G}, V_0)$

`Input data:` $\mathbf{G}$, $V_0$, $m \in [n]$
`Output data:` $V_0$, Ind, $\boldsymbol{x}$
Ind $= 0$
`for each` $v_i \in V_0$ `do`
  `if` $v_i$ is a $(|\mathcal{N}(v_i, V_0)|, m)$-vertex in the subgraph $\mathbf{G}\langle V_0\rangle$ `then`
    $x_i \leftarrow 1$
    $V_0 \leftarrow V_0 - (\{v_i\} \dot\cup \mathcal{N}(v_i, V_0))$
    Ind $\leftarrow 1$
    `break`
  `end of condition`
`end of loop`

*Algorithm* 2 sequentially checks, for the given value of $m$ and for each vertex of the initial set $V_0$, whether it is a $(|\mathcal{N}(v_i, V_0)|, m)$-vertex. If there are no such vertices, then no operations are performed, and the resulting set $V_0$ at the end of the work of the algorithm coincides with the input set $V_0$, the flag Ind $= 0$, a binary tuple $\boldsymbol{x}$ is not determined. In the case when such a vertex $v_i$ is found, the output set $V_0$ will be obtained from the input set $V_0$ by means of the "removal" of the vertex $v_i$ and its neighborhood, Ind $= 1$, and the corresponding component $x_i$ of the tuple $\boldsymbol{x}$ takes the value of 1.

**Algorithm 3.** Algorithm $B(\mathbf{G}, V_0)$

`Input data:` $\mathbf{G}$, $V_0$
`Output data:` $\boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}}^{-1}(0)$
`while` $V_0 \neq \emptyset$
  $m = 0$
  Ind $= 1$
  `while` (Ind $= 1$) & $V_0 \neq \emptyset$ `do`
    $A_m(\mathbf{G}, V_0)$
    Ind $\leftarrow$ Ind$(A_m(\mathbf{G}, V_0))$
  `end of loop`

  `while` (Ind $= 0$) & $V_0 \neq \emptyset$ `do`
    $m \leftarrow m + 1$
    $A_m(\mathbf{G}, V_0)$
    Ind $\leftarrow$ Ind$(A_m(\mathbf{G}, V_0))$
  `end of loop`
`end of loop`

During operation *Algorithm* 3, as the result of repeated calls of *Algorithm* 2, the tuple $\boldsymbol{x}$ is formed, which is a zero of the function $\mathfrak{f}_{\mathbf{G}}$.

**Proposition 3.** *Let $v_i$ be a $(k,m)$-vertex in a graph $\mathbf{G} := (V(\mathbf{G}), \mathcal{E}(\mathbf{G}))$. Then there exists a tuple $\boldsymbol{x}' \in \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$ such that $x'_i = 1$ and*

$$|\mathrm{supp}(\boldsymbol{x}')| \geq \max\nolimits_0 \mathfrak{f}_{\mathbf{G}} - m\ .$$

*Proof.* Suppose, according to the definition of the $(k,m)$–vertices, that for $v_i \in V(\mathbf{G})$ we have

$$\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_m\} := \tbinom{\mathcal{N}(v_i)}{2} - \left(\mathcal{E}(\mathbf{G}) \cap \tbinom{\mathcal{N}(v_i)}{2}\right)\ .$$

Then the vertex $v_i$ is a $k$-vertex in the graph $\mathbf{G}_1$, which is obtained from the graph $\mathbf{G}$ by the addition of $m$ edges $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_m\}$ into the neighborhood of the vertex $v_i$ of the graph $\mathbf{G}$ to turn the induced subgraph $\mathbf{G}_1\langle\mathcal{N}(v_i)\rangle$ into a complete graph.

According to Proposition 1, there exists a tuple $\boldsymbol{x}$ such that $x_i = 1$ and $\boldsymbol{x} \in \max\limits_{|\cdot|} \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}_1}^{-1}(0)$.

According to Corollary 2, for the graph $\mathbf{G}_1$ we have:

$$|\mathrm{supp}(\boldsymbol{x})| = \max\nolimits_0 \mathfrak{f}_{\mathbf{G}_1} \geq \max\nolimits_0 \mathfrak{f}_{\mathbf{G}} - m\ .$$

It follows from Lemma 1 that $\boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}}^{-1}(0)$. By the definition of the upper zeros, there exists a tuple $\boldsymbol{x}' \in \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$ such that $\boldsymbol{x}' \geq \boldsymbol{x}$ and, as a consequence,

$$|\mathrm{supp}(\boldsymbol{x}')| \geq |\mathrm{supp}(\boldsymbol{x})| \geq \max\nolimits_0 \mathfrak{f}_{\mathbf{G}} - m\ ,$$

as it was to be proved.

□

In every next loop of *Algorithm* 1, the search is terminated when some $k$-vertex is found. Such an approach minimizes the number of operations in the working loop of the algorithm, but it does not necessarily lead to the best solution in the case when $V_0 \neq \emptyset$.

Let us present an *Algorithm* 4, in each next working loop of which the parameters $k$ and $m$ are calculated for every vertex from the current set $V_0$.

---

**Algorithm 4.**

`Input data:` $\mathbf{G}$, $V_0$, $m = 0$

`Output data:` $\boldsymbol{x} \in \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$, and $m$ which is the estimate of deviation from $\max_0 \mathfrak{f}_{\mathbf{G}}$

`while` $V_0 \neq \emptyset$

for all vertices $v_i \in V_0 \neq \emptyset$, to calculate the parameters $k_i$ and $m_i$ such that $v_i$ is a $(k_i, m_i)$-vertex in the graph $\mathbf{G}\langle V_0\rangle$; in the set $V_0$, to extract the subset $V'_0 \subseteq V_0$ of vertices with the minimal values of the parameter $m_i$. Among the extracted vertices in the set $V'_0$, to find a vertex $v_{i_0} \in V'_0$ with the maximal value of the parameter $k_{i_0}$

$x_{i_0} \leftarrow 1$

$m \leftarrow m + m_{i_0}$

$V_0 \leftarrow V_0 - (\{v_{i_0}\} \mathbin{\dot\cup} \mathcal{N}(v_{i_0}, V_0))$

`end of loop`

---

*Algorithm* 4 finds a tuple $\boldsymbol{x} \in \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$, for which the precision estimate $\max_0 \mathfrak{f}_{\mathbf{G}} - |\mathrm{supp}(\boldsymbol{x})| \le m$ of the solution is true.

Let us estimate the complexity of *Algorithm* 4.

For each vertex $v_i$ from the current set $V_0$, it is necessary to find the number of vertices in the neighborhood $\mathcal{N}(v_i, V_0)$ and the number of new edges that should be added into the neighborhood $\mathcal{N}(v_i, V_0)$ for turning the induced subgraph $\mathbf{G}\langle\mathcal{N}(v_i, V_0)\rangle$ into a complete graph. We remove the vertices $v_i \dot\cup \mathcal{N}(v_i, V_0)$ and the edges $\boldsymbol{e}_i \in \mathbf{G}\langle\{v_i\}\dot\cup\mathcal{N}(v_i, V_0)\rangle$ until the current set of vertices $V_0$ becomes empty. Given the input data $V(\mathbf{G}) = \{v_1, \ldots, v_n\}$ and $\mathcal{E}(\mathbf{G}) = \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_p\}$, we obtain the following estimate. The common number of iterations undertaken during the work of *Algorithm* 4 is less than or equal to $n$; every iteration demands no more than $O(np)$ actions for the computation of the parameters $k$ and $m$; and no more than $O(p)$ actions are needed for the removal of a vertex and its neighborhood from the current graph. Thus, *Algorithm* 4 has the complexity of $O(n \cdot np + np) = O(n^2 p)$.

## 3. Solving the Problem of Searching for a Maximal Upper Zero

For some applied problems that are reduced to Problem 2, either exact results were obtained, or the significant decrease of the dimension of Problem 2 was achieved.

**Example 1.** The graph $\mathbf{G} := (V := \{v_1, \ldots, v_{22}\}, \mathcal{E})$ is specified by the incidence lists $\mathcal{N}(v_i)$ of its vertices, $i \in [22]$, $V_0 = V$:

$\mathcal{N}(v_1) := \{v_2, v_3, v_4, v_6, v_8, v_9\}$ ,
$\mathcal{N}(v_2) := \{v_1, v_3, v_4, v_6, v_{12}\}$ ,
$\mathcal{N}(v_3) := \{v_1, v_2, v_4, v_7, v_{12}\}$ ,
$\mathcal{N}(v_4) := \{v_1, v_2, v_3, v_5, v_6, v_8, v_9, v_{10}, v_{12}\}$ ,
$\mathcal{N}(v_5) := \{v_4, v_6, v_7, v_9, v_{10}\}$ ,
$\mathcal{N}(v_6) := \{v_1, v_2, v_4, v_5, v_7, v_8, v_9, v_{12}\}$ ,
$\mathcal{N}(v_7) := \{v_3, v_5, v_6\}$ ,
$\mathcal{N}(v_8) := \{v_1, v_4, v_6, v_9\}$ ,
$\mathcal{N}(v_9) := \{v_1, v_4, v_5, v_6, v_8, v_{10}\}$ ,
$\mathcal{N}(v_{10}) := \{v_4, v_5, v_9, v_{11}, v_{18}, v_{20}\}$ ,
$\mathcal{N}(v_{11}) := \{v_{10}, v_{12}, v_{13}, v_{14}, v_{15}\}$ ,
$\mathcal{N}(v_{12}) := \{v_2, v_3, v_4, v_6, v_{11}, v_{17}\}$ ,
$\mathcal{N}(v_{13}) := \{v_{11}, v_{14}, v_{15}\}$ ,
$\mathcal{N}(v_{14}) := \{v_{11}, v_{13}, v_{15}\}$ ,
$\mathcal{N}(v_{15}) := \{v_{11}, v_{13}, v_{14}, v_{16}\}$ ,
$\mathcal{N}(v_{16}) := \{v_{15}, v_{17}\}$ ,
$\mathcal{N}(v_{17}) := \{v_{12}, v_{16}, v_{18}, v_{19}, v_{21}, v_{22}\}$ ,
$\mathcal{N}(v_{18}) := \{v_{10}, v_{17}, v_{19}, v_{21}, v_{22}\}$ ,
$\mathcal{N}(v_{19}) := \{v_{17}, v_{18}, v_{21}, v_{22}\}$ ,
$\mathcal{N}(v_{20}) := \{v_{10}, v_{21}, v_{22}\}$ ,
$\mathcal{N}(v_{21}) := \{v_{17}, v_{18}, v_{19}, v_{20}\}$ ,
$\mathcal{N}(v_{22}) := \{v_{17}, v_{18}, v_{19}, v_{20}\}$ .

Acting in accordance with *Algorithm* 1, for each vertex $v_i \in V_0$ we check whether it is a $k$-vertex in the graph $\mathbf{G}$.
$A(\mathbf{G}, V_0)$:

$v_{1\ (2,3,4,5,6,7)}$ is not a 6 $(5, 5, 9, 5, 8, 3)$-vertex.
$v_8$ is a 4-vertex $\Rightarrow$ $x_8 \leftarrow 1$; $V_0 \leftarrow V_0 - \{v_1, v_4, v_6, v_8, v_9\}$.
$v_2$ is a 2-vertex $\Rightarrow$ $x_2 \leftarrow 1$; $V_0 \leftarrow V_0 - \{v_2, v_3, v_{12}\}$.
$v_5$ is not a 2-vertex.
$v_7$ is a 1-vertex $\Rightarrow$ $x_7 \leftarrow 1$; $V_0 \leftarrow V_0 - \{v_5, v_7\}$.
$v_{10\ (11)}$ is not a 3 (4)-vertex.
$v_{13}$ is a 3-vertex $\Rightarrow$ $x_{13} \leftarrow 1$; $V_0 \leftarrow V_0 - \{v_{11}, v_{13}, v_{14}, v_{15}\}$.
$v_{10}$ is not a 2-vertex.
$v_{16}$ is a 1-vertex $\Rightarrow$ $x_{16} \leftarrow 1$; $V_0 \leftarrow V_0 - \{v_{16}, v_{17}\}$.
$v_{10\ (18,19,20,21,22)}$ is not a 2 $(4, 3, 3, 3, 3)$-vertex.

$\boldsymbol{x} = (0,1,0,0,0,0,1,1,0,0,0,0,1,0,0,1,0,0,0,0,0,0)$ is a zero of the function $\mathfrak{f}_{\mathbf{G}}$, $\boldsymbol{x} \in \mathfrak{f}_{\mathbf{G}}^{-1}(0)$; besides, a maximal upper zero $\boldsymbol{x}' \in \max\limits_{|\cdot|}\max\limits_{\subseteq}\mathfrak{f}_{\mathbf{G}}^{-1}(0)$ of the function $\mathfrak{f}_{\mathbf{G}}$ has the form:

$$\boldsymbol{x}' = (0,1,0,0,0,0,1,1,0,x_{10},0,0,1,0,0,1,0,x_{18},x_{19},x_{20},x_{21},x_{22})\ .$$

Thus, the dimension of the problem was decreased from $|V_0| = 22$ to $|V_0| = |\{v_{10}, v_{18}, v_{19}, v_{20}, v_{21}, v_{22}\}| = 6$.

For exhausting the vertex set $V_0$, we follow *Algorithm* 3, that is, among the vertices from the set $V_0$ we search for $(k,m)$-vertices (the case of $m = 0$ corresponds to the search for $k$-vertices, which was undertaken by *Algorithm* 1).

**Table 1**

The result of the work of *Algorithm* 3

| $m$ | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | | $\boldsymbol{x}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Ind | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | | |
| $v_1$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_2$ | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 1 |
| $v_3$ | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_4$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_5$ | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_6$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_7$ | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | | 1 |
| $v_8$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 1 |
| $v_9$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_{10}$ | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | | 1 |
| $v_{11}$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_{12}$ | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_{13}$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | | 1 |
| $v_{14}$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_{15}$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | | 0 |
| $v_{16}$ | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | | 1 |
| $v_{17}$ | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | | 0 |
| $v_{18}$ | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | | 0 |
| $v_{19}$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | | 0 |
| $v_{20}$ | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | | 0 |
| $v_{21}$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | | 1 |
| $v_{22}$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | | 1 |

**Example 2.** Acting in accordance with *Algorithm* 3, for each vertex $v_i \in V_0$ we check whether it is a $(k,m)$-vertex in the graph $\mathbf{G}$.

$V_0 \neq \emptyset$, $m = 0$:

$\mathrm{Ind} = 0 \Rightarrow m \leftarrow m + 1 = 1$, $A_1(\mathbf{G}, V_0)$:

$v_{10}$ is a $(2,1)$-vertex: $x_{10} \leftarrow 1$, $V_0 \leftarrow V_0 - \{v_{10}, v_{18}, v_{20}\}$.

$\mathrm{Ind} = 1 \Rightarrow m = 0$, $A_0(\mathbf{G}, V_0)$:

$v_{19}$ is not a 2-vertex;

$v_{21}$ is a 1-vertex: $x_{21} \leftarrow 1$, $V_0 \leftarrow V_0 - \{v_{19}, v_{21}\}$.
Ind $= 1 \Rightarrow m = 0$, $A_0(\mathbf{G}, V_0)$:
$v_{22}$ is a 0-vertex: $x_{22} \leftarrow 1$, $V_0 \leftarrow V_0 - \{v_{22}\}$.
$V_0 = \emptyset$.

$\boldsymbol{x}' = (0,1,0,0,0,0,1,1,0,1,0,0,1,0,0,1,0,0,0,0,1,1)$ is a zero of the function $\mathfrak{f}_{\mathbf{G}}$, and it is a maximal upper zero of the function $\mathfrak{f}_{\mathbf{G}\cup\{(v_{18},v_{20})\}}$; then, according to Proposition 2, the number of unit components in a maximal upper zero of the function $\mathfrak{f}_{\mathbf{G}}$ is restricted by the inequality:

$$\max{}_0\mathfrak{f}_{\mathbf{G}} \leq \max{}_0\mathfrak{f}_{\mathbf{G}\cup\{(v_{18},v_{20})\}} + 1 = |\mathrm{supp}(\boldsymbol{x}')| + 1 = 9\ .$$

**Table 2**

The work of *Algorithm* 4

| | $k/m$ | $k/m$ | $k/m$ | $k/m$ | $k/m$ | $k/m$ | $k/m$ | $k/m$ | $k/m$ | $\boldsymbol{x}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $v_1$ | 6/5 | | | | | | | | | 0 |
| $v_2$ | 5/2 | 2/0 | 2/0 | | | | | | | 1 |
| $v_3$ | 5/5 | 3/2 | 3/2 | | | | | | | 0 |
| $v_4$ | 9/19 | | | | | | | | | 0 |
| $v_5$ | 5/4 | 2/1 | 2/1 | 2/1 | | | | | | 0 |
| $v_6$ | 8/15 | | | | | | | | | 0 |
| $v_7$ | 3/2 | 2/1 | 2/1 | 1/0 | | | | | | 1 |
| $v_8$ | 4/0 | | | | | | | | | 1 |
| $v_9$ | 6/5 | | | | | | | | | 0 |
| $v_{10}$ | 6/12 | 4/6 | 3/3 | 3/3 | 2/1 | 2/1 | 1/0 | | | 1 |
| $v_{11}$ | 5/7 | 5/7 | | | | | | | | 0 |
| $v_{12}$ | 6/10 | 4/5 | 3/2 | | | | | | | 0 |
| $v_{13}$ | 3/0 | 3/0 | | | | | | | | 1 |
| $v_{14}$ | 3/0 | 3/0 | | | | | | | | 0 |
| $v_{15}$ | 4/3 | 4/3 | | | | | | | | 0 |
| $v_{16}$ | 2/1 | 2/1 | 1/0 | 1/0 | 1/0 | | | | | 1 |
| $v_{17}$ | 6/10 | 6/10 | 6/10 | 5/5 | 5/5 | | | | | 0 |
| $v_{18}$ | 5/5 | 5/5 | 5/5 | 5/5 | 5/5 | 4/4 | | | | 0 |
| $v_{19}$ | 4/1 | 4/1 | 4/1 | 4/1 | 4/1 | 3/1 | | | | 1 |
| $v_{20}$ | 3/3 | 3/3 | 3/3 | 3/3 | 3/3 | 3/3 | 1/0 | | | 0 |
| $v_{21}$ | 4/3 | 4/3 | 4/3 | 4/3 | 4/3 | 3/2 | | | | 0 |
| $v_{22}$ | 4/3 | 4/3 | 4/3 | 4/3 | 4/3 | 3/2 | | | | 0 |

It is convenient to describe the result of the work of *Algorithm* 3 in the form of Table 1. The columns of the table correspond to the current state of the set $V_0$. We sequentially remove $k$-vertices and their neighborhoods from the set $V_0$, associating to the corresponding components $x_i$ of the value 1 in the case when $v_i$ is a $k$-vertex, and of the value 0 otherwise.

Table 2 describes the work of *Algorithm* 4. Every column of the table represents an iteration of *Algorithm* 4; the nonzero elements of a column correspond to the set $V_0$, and in an $i$-th row's values of $k$ and $m$ are related to the vertex $v_i$ in the current subgraph $\mathbf{G}\langle V_0\rangle$.

For the resulting tuple $\boldsymbol{x} = (0,1,0,0,0,0,1,1,1,0,1,0,0,1,0,0,1,0,0,1,0,0,0)$ it holds that $\boldsymbol{x} \in \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$ and according to Corollary 2 from Proposition 2 we see that $|\mathrm{supp}(\boldsymbol{x})| = 7 \geq \max_0 \mathfrak{f}_{\mathbf{G}} - 1$, or equally $\max_0 \mathfrak{f}_{\mathbf{G}} \leq 8$.

Earlier, for the tuple $\boldsymbol{x}'$ obtained with the help of *Algorithm* 3, we also obtained that $\max_0 \mathfrak{f}_{\mathbf{G}} \leq 9$. Since $\boldsymbol{x'} \in \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$, $|\mathrm{supp}(\boldsymbol{x'})| = 8$ and $\max_0 \mathfrak{f}_{\mathbf{G}} \leq 8$, we see that $\boldsymbol{x'} \in \max\limits_{|\cdot|} \max\limits_{\subseteq} \mathfrak{f}_{\mathbf{G}}^{-1}(0)$ and $\max_0 \mathfrak{f}_{\mathbf{G}} = 8$.

## References


1. Gainanov D.N. *Kombinatornaya geometriya i grafy v analize nesovmestnykh sistem i raspoznavanii obrazov* [Combinatorial Geometry and Graphs in an Analysis of Infeasible Systems and Pattern Recognition]. Moscow, 2014. (in Russian)
2. Gainanov D.N. On One Criterion of the Optimality of an Algorithm for Evaluating Monotonic Boolean Functions. *Zhurnal vychislitel'noi matematiki i matematicheskoi fiziki* [USSR Computational Mathematics and Mathematical Physics], 1984, vol. 24, no. 4, pp. 176–181. (in Russian)
3. Korshunov A.D. Monotone Boolean functions. *Uspekhi matematicheskikh nauk* [Progress in Mathematical Sciences], 2003, vol. 58, no. 5 (535), pp. 89–162. (in Russian)
4. Sapozhenko A.A. *Problema Dedekinda i metod granichnykh funktsionalov* [Dedekind's Problem and the Method of Boundary Functionals]. Moscow, 2009. (in Russian)
5. Bioch J.C., Ibaraki T., Makino K. Minimum Self-Dualdecompositions of Positive Dual-Minor Boolean Functions. *Discrete Applied Mathematics*, 1999, vol. 96-97, pp. 307–326.
6. Boros E., Hammer P., Ibaraki T., Kawakami K. Polynomial Time Ecognition of 2-monotonic Positive Boolean Functions Given by an Oracle. *SIAM Journal on Computing*, 1997, no. 26, pp. 93–109.
7. Domingo C., Mishra N., Pitt L. Efficient Read-Restricted Monotone CNF/DNF Dualization by Learning with Membership Queries. *Machine Learning*, 1999, no. 37 (1), pp. 89–110.
8. Makino K., Ibaraki T. A Fast and Simple Algorithm for Identifying 2-Monotonic Positive Boolean Functions. *Journal of Algorithms*, 1998, no. 26 (2), pp. 291–305.
9. Makino K., Ibaraki T. The Maximum Latency and Identification of Positive Boolean Functions. *SIAM Journal on Computing*, 1997, no. 26, pp. 1363–1383.
10. Torvik V.I., Triantaphyllou E. Guided Inference of Nested Monotone Boolean Functions. *Information Sciences*, 2003, no. 151 (SUPPL), pp. 171–200.
11. Triantaphyllou E. *Data Mining and Knowleadge Discovery Via Logic-Based Methods. Theory, Algorithms and Applications.* N.Y., Springer, 2010.
12. Valiant L. A Theory of the Learnable. *Communications of the ACM*, 1984, no. 27 (11), pp. 1134–1142.
13. Torvik V.I., Triantaphyllou E. Inference of Monotone Boolean Functions. *Encyclopedia of Optimization.* N.Y., Springer, 2009, pp. 1591–1598.


*Received April 1, 2016*

**УДК 519.1** **DOI: 10.14529/mmp160302**

# АЛГОРИТМ РАСШИФРОВКИ МОНОТОННЫХ БУЛЕВЫХ ФУНКЦИЙ, ПОРОЖДАЕМЫХ НЕОРИЕНТИРОВАННЫМИ ГРАФАМИ

***Д.Н. Гайнанов, В.А. Рассказова***

Существует достаточно прикладных задач, в которых одним из инструментов моделирования служат булевы функции, среди которых важную роль играют монотонные булевы функции. Например, монотонные булевы функции являются удобным средством для описания структуры совместных подсистем несовместных систем условий, поскольку совместность является монотонным свойством.

В работе рассматриваются монотонные булевы функции, порождаемые неориентированными графами, в которых нули функции определяются как такие двоичные наборы, для которых соответствующий подграф исходного неориентированного графа пуст, или не содержит ребер. Для такого класса монотонных булевых функций даются постановки задач, связанных с выделением верхних нулей и максимальных верхних нулей функции. Вводятся понятия $k$-вершины и $(k, m)$-вершины в неориентированном графе. Показано, что для любой $k$-вершины исходного графа существует максимальный верхний нуль порожденной монотонной булевой функции, в котором компонента $x_i$, соответствующая этой $k$-вершине, принимает значение 1.

На основе этого утверждения построен алгоритм выделения максимального верхнего нуля для рассматриваемого класса монотонных булевых функций, который гарантирует, при определенных условиях, нахождение точного решения задачи поиска максимального верхнего нуля, либо приводит к снижению размерности исходной задачи. Предложенный алгоритм обобщается для случая использования $(k, m)$-вершин. Построенный алгоритм выделяет верхний нуль монотонной булевой функции и дает оценку его отклонения от максимального верхнего нуля по числу единиц в этих наборах. Алгоритм имеет сложность $O(n^2p)$, где $n$ – число вершин и $p$ – число ребер исходного графа.

*Ключевые слова: монотонная булева функция; верхний нуль монотонной булевой функции; неориентированный граф; алгоритм поиска максимальных верхних нулей монотонной булевой функции.*

*Работа проводилась при финансовой поддержке КЦП (Коллективный центр превосходства) «Квантум и видеоинформационные технологии» программа развития Уральского федерального университета им. первого президента России Б.Н. Ельцина.*

## Литература

1. Гайнанов, Д.Н. Комбинаторная геометрия и графы в анализе несовместных систем и распознавании образов / Д.Н. Гайнанов. – М.: Наука, 2014.
2. Гайнанов, Д.Н. Об одном критерии оптимальности алгоритма расшифровки монотонных булевых функций / Д.Н. Гайнанов // Журнал вычислительной математики и математической физики. – 1984. – Т. 24, № 8. – С. 1250–1257.
3. Коршунов, А.Д. Монотонные булевы функции / А.Д. Коршунов // Успехи математических наук. – 2003. – Т. 58, № 5 (535). – С. 89–162.

4. Сапоженко, А.А. Проблема Дедекинда и метод граничных функционалов / А.А. Сапоженко. – М.: Физматлит, 2009.
5. Bioch, J.C. Minimum Self-Dualdecompositions of Positive Dual-Minor Boolean Functions / J.C. Bioch, T. Ibaraki, K. Makino // Discrete Applied Mathematics. – 1999. – V. 96–97. – P. 307–326.
6. Boros, E. Polynomial Time Ecognition of 2-monotonic Positive Boolean Functions Given by an Oracle / E. Boros, P. Hammer, T. Ibaraki, K. Kawakami // SIAM Journal on Computing. – 1997. – № 26. – P. 93–109.
7. Domingo, C. Efficient Read-restricted Monotone CNF/DNF Dualization by Learning with Membership Queries / C. Domingo, N. Mishra, L. Pitt // Machine Learning. – 1999. – № 37 (1). – P. 89–110.
8. Makino, K. A Fast and Simple Algorithm for Identifying 2-Monotonic Positive Boolean Functions / K. Makino, T. Ibaraki // Journal of Algorithms. – 1998. – № 26 (2). – P. 291–305.
9. Makino, K. The Maximum Latency and Identification of Positive Boolean Functions / K. Makino, T. Ibaraki // SIAM Journal on Computing. – 1997. – № 26. – P. 1363–1383.
10. Torvik, V.I. Guided Inference of Nested Monotone Boolean Functions / V.I. Torvik, E. Triantaphyllou // Information Sciences. – 2003. – № 151 (SUPPL). – P. 171–200.
11. Triantaphyllou, E. Data Mining and Knowleadge Discovery Via Logic-Based Methods. Theory, Algorithms and Applications / E. Triantaphyllou. – N.Y.: Springer, 2010.
12. Valiant, L. A Theory of the Learnable / L. Valiant // Communications of the ACM. – 1984. – № 27 (11). – P. 1134–1142.
13. Torvik, V.I. Triantaphyllou E. Inference of Monotone Boolean Functions / Torvik, V.I. – Encyclopedia of Optimization. – N.Y.: Springer, 2009. – P. 1591–1598.

Дамир Насибуллович Гайнанов, кандидат технических наук, заведующий кафедрой «Аналитика больших данных и методы видеоанализа», Уральский федеральный университет им. первого Президента России Б.Н. Ельцина (г. Екатеринбург, Российская Федерация), damir.gainanov@gmail.com.

Варвара Андреевна Рассказова, аспирант, кафедра «Теория вероятностей», Московский авиационный институт (г. Москва, Российская Федерация), 2265874@mail.ru.

*Поступила в редакцию 1 апреля 2016 г.*