UDC 519.177

N. A. Yakimova, *CandSc (Engineering), Assoc. Prof.*
Odesa I. I. Mechnikov National University
Department of Algebra, Geometry and Differential Equations
2 Zmiienka Vsevoloda St, Odesa, 65082, Ukraine
e-mail: nataliya_yakimova@onu.edu.ua
ORCID iD: https://orcid.org/0009-0005-8577-2775
N. V. Sharai, *CandSc (Physics & Mathematics), Assoc. Prof.*
Odesa I. I. Mechnikov National University
Department of Algebra, Geometry and Differential Equations
2 Zmiienka Vsevoloda St, Odesa, 65082, Ukraine
e-mail: sharay@onu.edu.ua
ORCID iD: https://orcid.org/0009-0003-4007-651X

ALGEBRAIC IMPLEMENTATION OF BINARY OPERATIONS ON GRAPHS

Graph theory is widely used from a practical point of view. Graphs play an important role in scientific research (for example, electrical diagrams), and also surround us in everyday life (for example, roads and paths maps). For everyday use, the geometric implementation of graphs is certainly the most convenient. But for computer processing of information, this is not rational. In these cases, an algebraic, namely matrix representation of graphs is used. Therefore, studies devoted to this topic are gaining more and more importance. This article considers the possibility of algebraic performing operations on adjacency matrices that represent graphs. These methods have their own characteristics and limitations.

MSC: 03G05, 03G25, 03F52, 06E25, 15B34.
Keywords: directed and undirected graph, adjacency matrix, operations on graphs, elementary logical operations, Boolean matrix, multivalued logic.
DOI: https://doi.org/10.18524/2519-206X.2025.1(45).352814

Introduction

When studying a system of objects connected by some arbitrary types of relationships, both directed and undirected graphs can be used. Each such system is an ordered collection of elements with which certain changes can occur. Each such specific system can be represented graphically as a graph or

Received 11.06.2025

in digital format as an adjacency matrix or incidence matrix of such a graph. In any case, changes in system elements or the results of the interaction of different such systems are reflected by operations on the vertices or edges of the corresponding graphs. The geometric implementation of such operations has already been well studied and described [1]. But computer processing of information involves its digital representation in matrix form. Algebraic matrix apparatus is also widely represented in mathematical research [2]. This article aims to establish a correspondence between known operations on matrices and operations on elements of an arbitrary graph or operations that reflect the interaction of multiple graphs. In this way, a transition from a geometric to algebraic method of not only representing, but also processing various information can be made.

Preliminary Results

If the matrices are Boolean, then with them it is possible to perform both ordinary algebraic operations on matrices and two-valued logic operations described in [3; 8]. If the matrices are not Boolean, then in order to perform the logical operations of disjunction and conjunction with them, it is necessary to use the apparatus of multi-valued logic. In this case, the operations of disjunction and conjunction of matrix elements are performed according to the following rules [4]:

$$x \vee y = \max\{x, y\} \tag{1}$$

$$x \wedge y = \min\{x, y\} \tag{2}$$

But both logical and arithmetic operations on matrices require certain conditions regarding their dimensionality. Different systems represented by geometric implementations of graphs do not always have the same number of objects (nodes). Therefore, the graphs corresponding to them will have a different number of vertices. This implies a different dimension of their adjacency matrices. Neither logical nor arithmetic operations can be performed on such matrices. This obstacle can be avoided by reducing both matrices to the same dimension by introducing additional identically named all-zero rows and columns into them. According to the characterization of graphs by their adjacency matrices, such pairs will correspond to isolated vertices [5; 9]. The new extended adjacency matrices of both graphs participating in the operation will have the same dimension. As is known, an isomorphism is a relation between graphs that preserves the incidence relation up to the numbering of vertices.

Theorem 1. *Graphs are isomorphic if and only if their adjacency matrices can be obtained from each other by simultaneous permutations of the same-named rows and columns (i.e., simultaneously with the permutation of the i-th and j-th rows of the matrix, the permutation of the i- th and j-th columns of the matrix also occurs).*

Proof. Renumber the vertices of arbitrary graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)(|V_1| = |V_2| = p)$ by integers from 1 to p. If $A'(G_1) = A''(G_2)$ then the statement of the theorem is true. In the opposite case, the graphs G_1 and G_2 differ only in the numbering of the vertices. This means that there exists a permutation S on the set of vertices V that preserves adjacency, i.e. if $e^1_{ij}(v^1_i, v^1_j) \in E_1$, then $e^2_{s(i,j)}(s(v^1_i), s(v^1_j)) \in E_2$. Then we have $a''_{s(i)s(j)} = a'_{ij}$. The theorem is proved.

This theorem implies that, using an isomorphism transformation for each graph in the extended adjacency matrix, new isolated vertices will be assigned row-column pair numbers that reflect vertices that are absent in one graph but present in the other.

Main Results

Let's consider the basic operations on graphs.

Theorem 2. *The adjacency matrix of the result of the graph union operation corresponds to the disjunction of the adjacency matrices of the graphs being joined.*

Proof. By definition, the graph H is the union of the graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, if $H = (V_1 \cup V_2, E_1 \cup E_2)$. If the graphs being joined have the same number of vertices, then the set of vertices of the graph H coincides with the sets of vertices of the graphs G_1 and G_2, i.e. $V_1 \cup V_2 = V_1 = V_2$. In this case, their adjacency matrices $A(H)$, $A(G_1)$ and $A(G_2)$ will have the same dimension, so any operations can be performed with them without additional preliminary transformations. If the number of vertices in the graphs under study is different or has a different semantic load and, as a result, different numbering, then after introducing additional zero row-column pairs, we obtain matrices of the same dimension. By permutations, each of these matrices, according to Theorem 1, can be reduced to matrices of graphs isomorphic to the original ones, where all vertices have the same numbering, which corresponds to their meaningful loading. The elements of the adjacency matrix correspond to the number of edges connecting the corresponding vertices. Therefore, if in at least one of the graphs the vertices are adjacent, i.e. connected by a certain number of edges, then in the adjacency matrix of the union of these graphs the specified vertices will be connected by the same number of edges. This corresponds to the definition of the disjunction operation in its multivalued sense according to formula (1). Thus, to calculate the adjacency matrix of the result of the union of two arbitrary graphs, it is necessary to perform the disjunction operation of the adjacency matrices of these graphs. The theorem is proved.

We will illustrate the result of this theorem with an example. Consider the two directed graphs shown in Fig. 1.

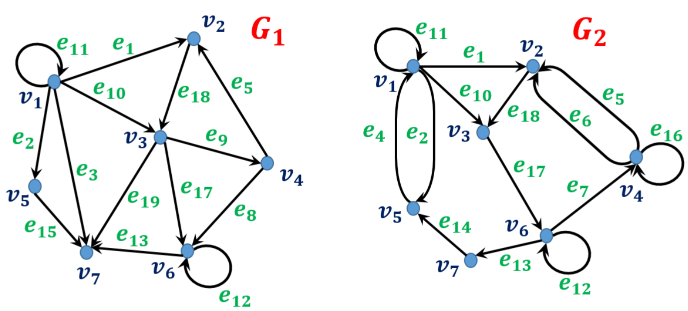

Figure 1: Directed graphs.

For both of these graphs, we can construct adjacency matrices in tabular form

$A(G_1) =$

Initial \ Final	v_1	v_2	v_3	v_4	v_5	v_6	v_7
v_1	1	1	1	0	1	0	1
v_2	0	0	1	0	0	0	0
v_3	0	0	0	1	0	1	1
v_4	0	1	0	0	0	1	0
v_5	0	0	0	0	0	0	1
v_6	0	0	0	0	0	1	1
v_7	0	0	0	0	0	0	0

$A(G_2) =$

Initial \ Final	v_1	v_2	v_3	v_4	v_5	v_6	v_7
v_1	1	1	1	0	1	0	0
v_2	0	0	1	0	0	0	0
v_3	0	0	0	0	0	1	0
v_4	0	2	0	1	0	0	0
v_5	1	0	0	0	0	0	0
v_6	0	0	0	1	0	1	1
v_7	0	0	0	0	1	0	0

or in the usual algebraic form

$$A(G_1) = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad A(G_2) = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

These matrices are constructed assuming that the row number corresponds to the starting vertex and the column number corresponds to the final vertex of each edge. But these graphs can be defined by these matrices from the very beginning. Performing operations on such graphs does not require reproduction of their geometric implementation [6]. The graph G_1 does not contain multiple edges, so its adjacency matrix is Boolean. Graph G_2 contains strictly parallel edges $e_5(v_4, v_2)$ and $e_6(v_4, v_2)$. Therefore, its adjacency matrix contains the element $a_{42} = 2$, i.e. it is not Boolean [3; 8]. But if we take into account that the disjunction operation for multivalued logic is performed according to rule (1), then the disjunction for these matrices takes the form [6; 7]:

$$A(G_1) \vee A(G_2) = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \vee \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 2 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} = A_{\cup}$$

If the union operation of the specified graphs is performed geometrically, then we will get the graph shown in Fig. 2.

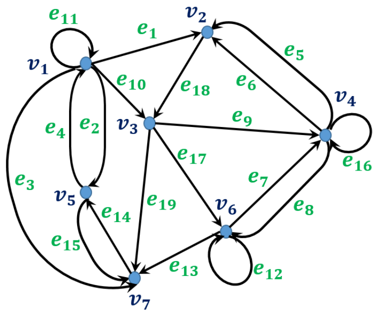

Figure 2: Graph $G_1 \cup G_2$.

It is easy to see that the adjacency matrix A_U corresponds to this graph.

Theorem 3. *The adjacency matrix of the result of the graph intersection operation corresponds to the conjunction of the adjacency matrices of the intersecting graphs.*

Proof. By definition, the graph F is the intersection of the graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, if $F = (V_1 \cap V_2, E_1 \cap E_2)$. If the graphs being joined have the same number of vertices, then the set of vertices of the graph F coincides with the sets of vertices of the graphs G_1 and G_2, i.e. $V_1 \cap V_2 = V_1 = V_2$. In this case, their adjacency matrices $A(F)$, $A(G_1)$ and $A(G_2)$ will have the same dimension, so any operations can be performed with them without additional preliminary transformations. If the sets of vertices in the graphs under study do not coincide, then, as in the case of unification graphs, after introducing additional zero row-column pairs, we obtain matrices of the same dimension, which are transformed into matrices of graphs isomorphic to the original graphs. The elements of the adjacency matrix correspond to the number of edges connecting the corresponding vertices. Therefore, if in at least one of the graphs any two vertices are connected by a smaller number of edges than in the second graph, then in the adjacency matrix of the intersection of these graphs the specified vertices will be connected by the same smaller number of edges. This corresponds to the definition of the conjunction operation in its multivalued sense according to formula (2). Thus, to calculate the adjacency matrix of the result of the intersection of two arbitrary graphs, it is necessary to perform the conjunction operation of the adjacency matrices of these graphs. The theorem is proved.

Let us illustrate this theorem using the example of the graphs shown in Fig. 1. As already noted, for multivalued logic the conjunction operation is performed according to rule (2), so the conjunction for the matrices $A(G_1)$ and $A(G_2)$ takes the form:

$$A(G_1) \wedge A(G_2) = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \wedge \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = A_\cap$$

If the intersection of the specified graphs is performed geometrically, we will get the graph shown in Fig. 3.

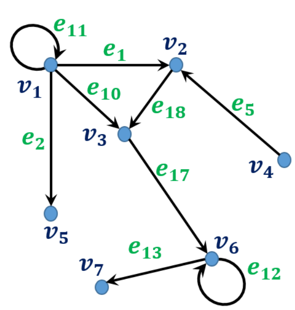

Figure 3: Graph $G_1 \cap G_2$.

It is easy to see that the adjacency matrix $A_\cap$ corresponds to this graph [6].

Theorem 4. *The adjacency matrix of the result of performing the ring sum operation of graphs corresponds to the arithmetic subtraction of the adjacency* *matrices of the union and intersection of the graphs that participate in the specified operation.*

Proof. By definition, a graph R is a ring sum of graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ if it does not contain isolated vertices and consists of edges belonging to either graph G_1 or graph G_2, but not to both simultaneously, i.e. [7]

$$E_{\oplus} = \{(E_{G_1} \cup E_{G_2}) \backslash (E_{G_1} \cap E_{G_2})\} \tag{3}$$

The set of edges $E_{G_1} \cup E_{G_2}$ is the result of the graphs union operation, and the set $E_{G_1} \cap E_{G_2}$ is the result of their intersection. This means that the matrix

$$A(R) = A_{\cup} - A_{\cap} = (A(G_1) \vee A(G_2)) - (A(G_1) \wedge A(G_2)) \tag{4}$$

will contain complete information about all edge-connected vertices of the graph R, i.e. will be its adjacency matrix in the extended sense. The theorem is proved.

By the definition of a ring sum of graphs, the graph obtained as a result of this operation cannot contain isolated vertices. Therefore, if they appear, they must be deleted from the resulting graph. Deleting a vertex from the graph entails deleting all edges incident to it, that is, deleting all connections of this object or node with other objects or nodes [7; 10]. This means that when deleting the vertex v_i from the adjacency matrix, it is necessary to delete the i-th row and i-th column. In this regard, the algorithm for performing the operation of deleting the vertex v_i from the graph in the matrix representation is similar to the algorithm for constructing the minor M_{ii} for the adjacency matrix of this graph. Let, for example, it is necessary to delete the vertex v_2 from the graph G_2. Let's construct for the matrix $A(G_2)$ the minor M_{22} [1]:

$$M_{22}(A(G_2)) = \begin{vmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{vmatrix}$$

$$M_{22}(A(G_2)) = \begin{vmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{vmatrix}$$

Therefore, the following adjacency matrix will correspond to the new graph $G_2' = G_2 \backslash \{v_2\}$

$$A(G_2') = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

The algorithm for deleting rows and columns from the matrix is already computerized. In this case, the software implementation will provide two shifts: for rows and columns [6; 7]. A sign of an isolated vertex in the adjacency matrix is the presence of a row and column of the same name that are all zero [5]. According to the described algorithm, this zero row-column pair should be removed from the extended adjacency matrix of the ring sum of graphs if such a pair appeared as a result of the procedure described in Theorem 4. The final matrix obtained as a result of these actions will be the adjacency matrix of the ring sum of the graphs under study.

Let us illustrate the result of Theorem 4 using the example of the graphs shown in Fig. 1. For these graphs, the matrices A_U and $A_\cap$ have already been obtained. Therefore,

$$A_U - A_\cap = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 2 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} - \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} = A_{\oplus}$$

This matrix does not contain all zero row-column pairs. This means that the ring sum operation did not result in any isolated vertices that would be subsequently deleted from the resulting graph. Therefore, the obtained adjacency matrix does not require further processing by constructing its corresponding minor. Thus, the resulting matrix is the final adjacency matrix of the ring sum of the considered graphs. If we perform the ring sum operation of graphs G_1 and G_2 geometrically, we obtain the graph shown in Fig. 4.

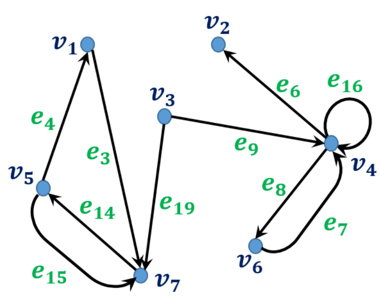

Figure 4: Graph $G_1 \bigoplus G_2$.

For this graph, it is also easy to see that the adjacency matrix $A_{\oplus}$ corresponds to it.

Theorem 5. *If two graphs do not contain multiple edges, or their number between corresponding vertices in these graphs differs by no more than one, then the adjacency matrix of the ring sum of these graphs can be obtained as a result of the sum modulo 2 operation of their adjacency matrices.*

Proof. As in the cases of graph union and intersection, first, if necessary, it is necessary to construct extended adjacency matrices for both graphs in order to achieve their same dimension. After that, using the isomorphism transformation, they should be reduced to matrices corresponding to the graphs under study. After that, the following situations are possible for the obtained adjacency matrices.

A). If the graphs do not contain multiple edges at all, then their adjacency matrices are Boolean. In this case, if two vertices are adjacent in both graphs (in the case of a directed graph, the vertices in both graphs are connected by the same directed edge), then in both matrices there will be ones at the corresponding place. But the given edge is not included in the result of the ring sum of these graphs. Therefore, in the adjacency matrix of the result, the corresponding element must be equal to zero. If two vertices are adjacent in only one graph, then such an edge will be an element of the ring sum of these graphs, that is, the corresponding element of the adjacency matrix of the result will be equal to one. If the vertices are not adjacent in any graph, then the result of the ring sum will also not be adjacent, i.e. the corresponding element of the result's adjacency matrix will be equal to zero. All this corresponds to the sum modulo 2 as an elementary Boolean operation.

B). In general, the sum modulo 2 is defined as the remainder of dividing the sum of the corresponding numbers by 2. If two vertices in both graphs are connected by the same number of q-multiple edges, then the sum of these edges will be equal to $2q$, which is an even number. The remainder of dividing such a number by 2 will always be zero, i.e. the corresponding matrix element will be equal to zero. This means that no multiple edge common to both graphs will be an element of the ring sum of graphs, which corresponds to the definition of this operation.

C). If any two vertices in both graphs are connected by multiple edges, and the number of these edges differs by more than one, then the number of multiple edges between the vertices when performing the ring sum of graphs must be equal to this number. This means that the corresponding element of the adjacency matrix of the result of this operation must be greater than one. But such a number cannot be the result of the sum modulo 2 operation. In this case, to calculate the adjacency matrix of the ring sum of graphs, we must use the formula (4) proposed by Theorem 4.

D). If the number of multiple edges connecting two vertices in both graphs differs by exactly one, then these numbers can be denoted as k and $k+1$. It follows from this that

$$(k + k + 1) \bmod 2 = (2k + 1) \bmod 2 = 1$$

So, in this situation, the result of the sum modulo 2 is indeed equal to the number of edges that are different among the multiple edges between the two vertices. Incidentally, situation A) can be considered a special case of this situation. The theorem is proven.

Let us illustrate the result of Theorem 5 with an example. Let us perform the sum modulo 2 operation with the given matrices $A(G_1)$ and $A(G_2)$, keeping in mind its general definition as the remainder of dividing the corresponding sum by 2 [6]. The indicated matrices correspond to situation D) described in Theorem 5.

$$A(G_1) \oplus A(G_2) = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \oplus \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} =$$

$$= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} = A_{\oplus}.$$

This matrix coincides with the matrix $A_{\oplus}$ for the ring sum of graphs G_1 and G_2, calculated by formula (4), that is, the adjacency matrix of the graph obtained as a result of the ring sum of two graphs can be calculated in the this way. The results of Theorem 4 and Theorem 5 clearly shows that for matrix execution of operations on graphs, the simultaneous use of both arithmetic and logical operations is permissible.

Conclusion

In the graphs considered as example, each edge is given a serial number. In practical application, these numbers may mean a certain content load. But with matrix display, this content can be lost. The matrix reflects the presence or absence of an edge, that is, the presence or absence of a connection between objects. Therefore, if in two graphs between two vertices the same edge has a different content load (for example, a road and a dirt road), then the adjacency matrix will only show the presence or absence of this connection without explaining its nature. But usually, in practical applications, information about the presence of a connection is sufficient, therefore, for binary operations on directed graphs [7], the use of elementary multivalued logic operations on adjacency matrices is an effective mathematical tool. The same algorithms have differences depending on whether directed or free graphs are involved in the considered operations. Depending on the types of graphs, there are also restrictions on the display of meaningful information by the matrices of these graphs. But in practical applications, these restrictions are usually insignificant. Therefore, for each operation on graphs and each type of graph, it is possible to propose a combination of algebraic operations (arithmetic and logical) that allow obtaining the matrix of a new graph, or a clear, easily programmable algorithm for transforming the matrices of the initial graphs [6; 11]. None of the considered operations on graphs is impossible in the matrix implementation. The proposed algorithms can significantly simplify the computer processing of graphs.

References

1. A.A. Zykov A.A. (2007). Lekcii z algebry [Lectures on algebra]. Odesa, Astroprint, 2007.

2. Gantmaker F. (2010). Teoriya matrytz [Matrix theory]. Kyiv: Nadrukovano v Ukraini, 560 p.

3. Yakimova N.A. (2019). Predykatni logichni matritzi [Predicative logical matrices]. Visnyk Odes 'kogo Natzional'nogo universitetu. Re-

searches in Mathematics and Mechanics. Vol.24, issue 2(34), pp.67–74. https://doi.org/10.18524/2519-206x.2019.2(34).190052

4. Shaporev S.D. (2005). Matematicheskaya logika [Mathematical logic]. St-Peterburg: BHV- Peterburg, 416 p.

5. Yakimova N.A. (2021). Dyskretna matematyka. Chastyna 1. Teoriya mnozhin. Teoriya grafiv [Discrete Math. Part 1. Set theory. Graf theory]. Odesa: ONU im. I. I. Mechnikova, 102 p.

6. Yakimova N.A., Klishin M.E. (2022). Matrichne podannya operaciy nad grafami [Matrix representation of operations on graphs]. Visnyk Odes 'kogo Natzional'nogo universitetu. Researches in mathematics and mechanics. Vol. 27, issue 1-2(39-40)), pp. 111-131. https://doi.org/10.18524/2519-206X.2022.1-2(39-40).294314

7. Yakimova N.A., Sharai N.V. (2024). Performing some operations on graphs using adjacency matrices. Proceedings of the 10th International Scientific and Practical Conference «Theory and Practice of Science: Key Aspects», Rome, Italy. P. 184-196. http://doi.org/10.51582/interconf.19-20.09.2024.019

8. Yakimova N.A. (2023). Operacii nad blochnymy predikatnymy matrytzyamy [Operations on block predicate matrices]. Visnyk Odes 'kogo Natzional'nogo universitetu. Researches in mathematics and mechanics. Vol.28, issue 1-2(41-42)), pp. 185-199. https://doi.org/10.18524/2519-206X.2023.1-2(41-42).305269

9. Yakimova N.A., Klishin M.E. (2024). Binary operations on graphs in matrix representation. Proceedings of the XXX International Scientific and Practical Conference «Youth, education and science through today's challenges», Porto, Portugal. P. 120-125. http://doi.org/10.46299/ISG.2024.1.30

10. Yakimova N.A. (2025). Possibility of matrix representation of basic unary operations on graphs. SWorldJournal. Vol. 30, part 2, 147153. http://doi.org/10.30888/2663-5712.2025-30-02-013

11. Kruglov V.E., Yakimova N.A. (2025). Predicate model of some operations on logic matrices. Proceedings of the 14^{th} International Scientific and Practical Conference «International Forum: Problems and scientific solutions», Melbourne, Australia. P. 249-254. http://doi.org/10.51582/interconf.2024.231

Якімова Н. А., Шарай Н .В.

Алгебраїчна інтерпретація бінарних операцій над графами

Резюме

Теорія графів має широке розповсюдження з практичної точки зору. Графи відіграють важливу роль в наукових дослідженнях (наприклад, електросхеми), а також оточують нас у повсякденному житті (наприклад, карти доріг та шляхів). Для побутового застосування, безумовно, найзручнішою є геометрична реалізація графів. Але для комп'ютерної обробки інформації це не є раціональним. В цих випадках використовується алгебраїчне, а саме матричне подання графів. Тому все більшого значення набувають дослідження, присвячені саме цій темі. В даній статті розглядається можливість алгебраїчного виконання операцій над матрицями суміжності, якими подано графи. Ці методи мають свої особливості та обмеження.

Ключові слова: орієнтований та неорієнтований граф, матриця суміжності, операції над графами, елементарні логічні операції, булева матриця, багатозначна логіка.

Список літератури

1. Зиков А. А. Лекции по алгебре. — Одесса: Астропринт, 2007.
2. Гантмахер Ф. Р. Теория матриц. — К.: Наукова думка, 2010. — 560 с.
3. Якімова Н. А. Предикатні логічні матриці // Вісник Одеського національного університету. Дослідження в математиці і механіці. 2019. Т. 24, № 2(34). С. 67–74. https://doi.org/10.18524/2519-206x.2019.2(34).190052
4. Шапорев С. Д. Математическая логика. — Санкт-Петербург: БХВ-Петербург, 2005. — 416 с.
5. Якімова Н. А. Дискретна математика. Частина 1. Теорія множин. Теорія графів. — Одеса: ОНУ ім. І. І. Мечникова, 2021. — 102 с.
6. Якімова Н. А., Клишин М. Є. Матричне подання операцій над графами // Вісник Одеського національного університету. Дослідження в Математиці і механіці. 2022. Т. 27, № 1–2(39-40). С. 111–131. https://doi.org/10.18524/2519-206X.2022.1-2(39-40).294314
7. Yakimova N. A., Sharai N. V. Performing some operations on graphs using adjacency matrices // Proceedings of the 10th International Scientific and Practical Conference "Theory and Practice of Science: Key Aspects", Rome, Italy, 2024. P. 184–196. http://doi.org/10.51582/interconf.19-20.09.2024.019
8. Якімова Н. А. Операції над блочними предикатними матрицями // Вісник Одеського національного університету. Дослідження в Математиці і механіці. 2023. Т. 28, № 1–2(41–42). С. 185–199. https://doi.org/10.18524/2519-206X.2023.1-2(41-42).305269

9. Yakimova N. A., Klishin M. E. Binary operations on graphs in matrix representation // Proceedings of the XXX International Scientific and Practical Conference "Youth, Education and Science through Today's Challenges", Porto, Portugal, 2024. P. 120–125. http://doi.org/10.46299/ISG.2024.1.30

10. Yakimova N. A. Possibility of matrix representation of basic unary operations on graphs // SWorldJournal. 2025. Vol. 30, Part 2. P. 147–153. http://doi.org/10.30888/2663-5712.2025-30-02-013

11. Kruglov V. E., Yakimova N. A. Predicate model of some operations on logic matrices // Proceedings of the 14th International Scientific and Practical Conference "International Forum: Problems and Scientific Solutions", Melbourne, Australia, 2025. P. 249–254. http://doi.org/10.51582/interconf.2024.231