SOME IDEAS ABOUT CONNECTED GRAPHS ISOMORPHISM

Larisa Marchenko[1], Viktoria Podgornaya[2]

[1] Faculty of Mathematics and Programming Technologies, Francisk Scorina Gomel State University, Gomel, Belarus

[2] V.A.Belyi Metal-polymer Research Institute of National Academy of Sciences of Belarus, Gomel, Belarus

Abstract: In the paper we investigate the existence of graphs isomorphism and the search for invariants of connected graphs. A new graph invariant is formulated. It can be used to detect isomorphism of connected graphs. The vector space of all simple cycles of the graph and their edge-disjoint unions (cycle space) and the vector space of all cutting sets of the graph and their edge-disjoint unions (cut space) are constructed in the article for finding a new graph invariant. The authors investigate the method of constructing these vector spaces: cycle space and cut space. A new estimate of the dimensions of these vector spaces of the graph is given. The obtained invariant is demonstrated on a concrete example. A counterexample is constructed to confirm the fact that the proposed invariant can be used as a necessary but not sufficient condition for graphs isomorphism. A heuristic algorithm is proposed for constructing a one-to-one correspondence between sets of vertices of isomorphic graphs.

Keywords: graph, cycle, cutting set, vector space, invariant, isomorphism algorithm

1. Introduction

Isomorphism of graphs is the equivalence relation on the set of all graphs of the same order. Detection of graphs isomorphism is required in various fields of theoretical and applied knowledge [5]. A number of scientific studies of recent decades [1–4] are devoted to the problem of identifying isomorphic graphs as a class of equivalent objects, but the question still remains unresolved.

Currently, methods for detecting isomorphism for some types of graphs are known [6, 7]. For almost every particular algorithmic problem of detecting isomorphism of a special kind graphs, it was possible either to construct a polynomial algorithm or to prove its belonging to the class of NP-complete problems [5]. However,

---

Advances in Computer Science Research, vol. 14, pp. 105-123, 2018. DOI: 10.24427/acsr-2018-vol14-0007

detecting isomorphism of arbitrary graphs is a more complex task, which is still not finally resolved. One approach to identifying isomorphism of arbitrary graphs is the application of invariants.

In isomorphic graphs, by definition, all characteristics and properties are the same. A characteristic of a graph is called an invariant if it does not change with an isomorphic transformation of the graph. In many papers, an invariant is called complete if its coincidence for different graphs guarantees the existence of their isomorphism. You can require that the invariant does not change when you renumber the vertices of the graph. However, the latter requirement, in our opinion, is not essential, since it does not correspond to the notion of automorphism of the graph. Note that the graph automorphism is a special case of graphs isomorphism in a broad sense. On the other hand, one of the forms of graph representation is adjacency matrix, which allows to describe fully a given graph in a compact numerical form and depends on the numbering of vertices in it. It is obvious that the complete invariant of the graph remains unchanged under any numbering of the vertices and edges of the graph.

The main easily computable graph invariants are: the number of vertices and edges of the graph, the vector of local degrees of vertices, the number of connected components. Also, the number of vertices of the largest complete subgraph (density), the largest number of pairwise non-adjacent vertices of the graph (non-density), the chromatic number and the chromatic index of the graph, the Hadwigers number and others were used as invariants by a number of authors [8]. All the characteristics listed here are calculated from the original graph, but their values do not allow to restore the graph structure. Therefore, the requirement of equality of these characteristics is a necessary condition for graphs isomorphism, since counterexamples are known. That is, in the case of identical invariants, graphs may not be isomorphic [11]. For some classes of graphs, only a collection of several numerical characteristics helps to identify isomorphism. There are also a number of papers devoted to the detection of graph isomorphism using eigenvalues and the vectors of the adjacency matrix, the graph spectrum [11].

In the problem of detecting isomorphism, we have two main subtasks. First, the problem of identifying the isomorphism of two graphs without specifying the most bijective mapping between the sets of their vertices. Secondly, establishing a one-to-one correspondence between the sets of vertices of graphs that preserve the adjacency of the corresponding vertices and all other characteristic properties, i.e. construction of a map that is an automorphism on the set of vertices of the graph.

When detecting graphs isomorphism, the most important thing is to search for criteria. As noted above, the equality of known invariants gives only the necessary signs of isomorphism. The only currently known sufficient condition for graphs isomorphism is the equality of adjacency matrices converted to the same type by permutations of rows and corresponding columns. This transformation in the general case, as we know, requires of order n! of steps'.

We note that to establish a bijective map, the Lux approach [9] is considered to be the most promising, which reduces the number of permutations using block structures. Based on this approach, Leslie Babay proposes a heuristic method for constructing graphs isomorphism [10].

Thus, the search for a sufficient condition for graphs isomorphism, which can be computed in polynomial time or at least almost polynomial, is relevant. Similarly, with respect to establishing a bijective map of vertex sets of isomorphic graphs. In this article new necessary condition for detection of graphs isomorphism are constructed.

2. Preliminary information

We present the necessary definitions and properties.

Definition. An abstract graph (graph) $G = (V, E)$ is a pair, which consist of vertices set $V = \{v\}$ and edges set $E = \{e = (u, v) | u, v \in V\}$. The graph G has an order n if $|V| = n$.

A pair of vertices can be connected by two or more edges, such edges are called multiples. An edge can begin and end at the same vertex, that is, $e = (v, v) \in E$. In this case, the edge is called a loop. A graph is called *simple* if it does not contain multiple edges and loops.

Definition. Graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are called isomorphic if there is a one-to-one correspondence $h : V_1 \to V_2$, preserving the adjacency of the corresponding vertices:

$$e_1 = (u, v) \in E_1 \Leftrightarrow e_2 = (h(u), h(v)) \in E_2.$$

Isomorphic graphs are naturally identified. And they can be represented in the same way. If the graphs G and H are isomorphic, then write $G \cong H$ (so $H \cong G$).

An isomorphic map of a graph onto itself is called an automorphism.

Definition. Let f be a function that associates with each graph G an element $f(G)$ from some set of M. The function f is called an invariant if its values on isomorphic graphs coincide:

$$G \cong H \Rightarrow f(G) = f(H)$$

for any graphs G and H.

The invariant f is called complete if, for any graphs G and H the equality of $f(G) = f(H)$ implies an isomorphism of graphs G and H.

As "conditional" full invariants, many authors consider the adjacency, incidence and Kirchhoffs matrices [12], since after transformation to the same form, they allow us to construct the required one-to-one correspondence between sets of graph vertices. The following theorem is known.

Theorem 1 [13]**.** Graphs are isomorphic if and only if their adjacency matrices (incidents, Kirchhoff) are permutationally similar, that is, they can be obtained one from the other by permutations of the rows and the corresponding columns.

Moreover, a uniform computational algorithm has not yet been formulated for an arbitrary graph without enumerating the elements of the set of vertices that has complexity of order less than $n!$.

For the considered vectors and matrices, addition and multiplication operations act in the Galois field $GF(2)$ modulo 2, unless otherwise specified.

The route in the graph $G = (V,E)$ is the finite sequence of its edges of the form (v_0, v_1), (v_1, v_2),..., (v_{k-1}, v_k). The number of edges in the route (with repetitions) is called *the length of the route*.

At the same time we have:

1) if $v_0 = v_k$, then the route is called *closed*, otherwise *open*;

2) if all edges of the route are different, then the route is called *a chain* and is denoted by $[v_0; v_k]$, if all vertices are different — *a simple chain*;

3) a closed chain $(v_k = v_0)$ is called *a cycle*, a closed simple chain is *a simple cycl*e.

The graph $G' = (V', E')$ is called the subgraph of the graph $G = (V,E)$, if $V' \subseteq V$ and $E' \subseteq E$. If $V' = V$, then the graph G' is called *the spanning subgraph* of the graph G.

The spanning tree of the graph $G = (V,E)$, $|V| = n$, is called a spanning subgraph without cycles.

The proof of Theorem 2.2 [12] implies that if a graph is connected, then it has at least one spanning tree. The converse is also true (Theorem 2.3, [12]).

Vertices u and v are called *reachable* if there is a route from u to v. A graph in which any two vertices are reachable is called *connected*. Any vertex v is connected with itself by a trivial route.

A connected subgraph G_1 of a graph G is called *a connected component* of a graph G if G_1 is maximal in the sense that no other connected subgraph G_2 of the graph G contains the subgraph G_1. The number of connected components of a graph is denoted by $k(G)$. For a connected graph, we have $k(G) = 1$.

The cutting set or cutset S of a connected graph G is the minimal set of edges whose removal makes the graph G disconnected.

Let G be a connected graph and $V = \{V_1, V_2\}$ be a partition of the set of its vertices: $V = V_1 \cup V_2$ and $V_1 \cap V_2 = \emptyset$. The set of edges of the graph G, one end of which belongs to V_1, and the other to V_2, is called a cut. The operation to delete the edges of the cut divides the graph into two connected components and makes it disconnected.

Definition[12]. Let T be an arbitrary spanning tree, $T^* = G \setminus T$ be the corresponding counting spanning tree of the graph $G = (V, E)$, $|V| = n$, $|E| = m$. And let e_i^* be the chord (edge) of counting spanning tree T^*. Since T is an acyclic graph, the graph $T \cup e_i^*$ contains exactly one cycle C_i. The cycle C_i consists of the chord e_i^* and those edges of the spanning tree T, which form a single simple chain between the terminal vertices of the chord e_i^*. The cycle C_i is called *a basic cycle* with respect to the chord e_i^* and the spanning tree T. The number of all basic cycles in an arbitrary graph is equal to the cyclomatic number of the graph $\nu(G) = m - n + k$, where k is the number of connected components.

The set of all basic cycles is called *the fundamental system of cycles* or *cycle basis* relative to the fixed spanning tree of T. The fundamental system of cycles is associated with a specific spanning tree. The number of spanning trees is equal to the algebraic complement of any element of the Kirchhoff's matrix. If we take another spanning tree, then it will correspond to a different set of cycles that form the *cycle basis*.

Removing the edge e_j from the spanning tree T breaks it into 2 components of connectivity T_1 and T_2. Let V_1 and V_2 be the sets of vertices of the components T_1 and T_2, respectively. And G_1 and G_2 be the subgraphs of the graph G, which are generated by the sets of the vertices V_1 and V_2. Obviously, T_1 is the spanning tree of the subgraph G_1, and T_2 is the spanning tree of the subgraph G_2. Consequently, the subgraphs G_1 and G_2 are connected. The separating cut V_1 and V_2 is the cutting set of the graph G. The cutting set S_j made up of edges connecting the vertices of the components T_1 and T_2 of the spanning tree is called *the base cut* of G.

The set of all basic cuts is called *the fundamental system of cuts* or *the fundamental cutsets* of the graph G with respect to the spanning tree T. The number of the basic cuts in an arbitrary graph is equal to *the rank* of the graph $\nu^*(G) = n - k$.

Properties of basic cycles and basic cuts:

1. The base cycle C_i with respect to the chord e_i^* of the T^* of the connected graph G includes exactly those edges of the spanning tree T, which correspond to the basic cuts which include this chord.

2. The base cut S_j with respect to the edge e_j of the spanning tree T of the connected graph G includes exactly those chords of the counting spanning trees T^*, which correspond to the basic cycles which are including this edge.

Let $(e_1, e_2, \dots, e_m)$ be a sequence of all edges of the graph $G = (V, E)$, $|V| = n$, $|E| = m$.

The base cycle C_i, $i = 1, \dots, \nu$, determines the vector $(c_{i1}, c_{i2}, \dots, c_{im})$, where $c_{ij} = 1$, if $e_j \in C_i$, and $c_{ij} = 0$, if $e_j \notin C_i$. The fundamental system of cycles corresponds to *the matrix of cycles* $C(G) = [c_{ij}]$, $i = 1, \dots, \nu$, $j = 1, \dots, m$. Since each basic cycle C_i contains exactly one chord, the matrix $C(G)$ can be transformed to canonical form by rearranging the columns

$$\hat{C}(G)_{\nu\times m} \sim \begin{pmatrix} 1 & 0 & \dots & 0 & a_{1\nu+1} & \dots & a_{1m-\nu} \\ 0 & 1 & \dots & 0 & a_{2\nu+1} & \dots & a_{2m-\nu} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & a_{\nu\nu+1} & \dots & a_{m-\nu} \end{pmatrix} = [\mathbf{E}_\nu \mid \mathbf{C}^*], \tag{1}$$

where $\mathbf{C}^* = \begin{pmatrix} a_{1\nu+1} & \dots & a_{1m-\nu} \\ a_{2\nu+1} & \dots & a_{2m-\nu} \\ \dots & \dots & \dots \\ a_{\nu\nu+1} & \dots & a_{\nu\, m-\nu} \end{pmatrix}$, $a_{ij} \in \{0, 1\}$, for $i = 1, \dots, \nu$, $j = 1, \dots, m - \nu$.

In contrast to [12], here we construct the cycle matrix for the graph, and not for the digraph.

In the matrix $C(G)_{\nu\times m}$, the columns of the unit submatrix $\mathbf{E}_\nu$ correspond to the edges of the counting spanning tree T^*, followed by the columns corresponding to the edges of the spanning tree T. Here we note that the cycle matrix does not define the entire graph up to isomorphism. For example, vertices of a graph of degree 1 will not be present in it.

Similarly to the basic cut S_i, for $i = 1, \dots, v^*$, there is a vector $(s_{i1}, s_{i2}, \dots, s_{im})$, where $s_{ij} = 1$, if the edge is $e_j \in S_i$, and $s_{ij} = 0$, if $e_j \notin S_i$. For the fundamental system of cuts, we can write the *matrix of cuts* $S(G) = [s_{ij}]$, where $i = 1, \dots, \nu^*$, $j = 1, \dots, m$, which, by rearranging the columns, also converts to canonical form

$$\hat{S}(G)_{\nu^*} \sim \begin{pmatrix} b_{11} & \dots & b_{1\nu} & 1 & 0 & \dots & 0 \\ b_{21} & \dots & b_{2\nu} & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{\nu^*1} & \dots & b_{\nu^*\nu} & 0 & 0 & \dots & 1 \end{pmatrix} = [\mathbf{S}^* | \mathbf{E}_{v^*}], \tag{2}$$

where $S^* = \begin{pmatrix} b_{11} & \dots & b_{1\nu} \\ b_{21} & \dots & b_{2\nu} \\ \dots & \dots & \dots \\ b_{\nu^*1} & \dots & b_{\nu^*\nu} \end{pmatrix}$, $b_{ij} \in \{0,1\}$, for $i = 1, \dots, \nu^*$, $j = 1, \dots, \nu$.

Similarly, here we build a matrix of cuts for a graph, and not for a digraph. Note that the matrix of cuts is also not a unique representation of the graph.

The rows of the matrix of cycles are called *the cycle vectors* of the graph G, the rows of the matrix of cuts are called *the vectors of the cuts*.

In [16] algorithms of construction of basic cycles of minimum length are given.

3. Main part

We will consider abstract connected graphs with number of vertices $n \geq 3$.

For the graph $G = (V,E)$, the Boolean (the set of all subsets) of the set E, including the empty set $\emptyset$, is denoted by W_G. The set W_G forms an Abelian group with operation of addition modulo 2, provided that all elements are represented as rows of 0 and 1 length $|E| = m$ by the following rule. If the edge e_i belongs to a subset, then the i-th coordinate is 1, if it does not belong to — 0. Addition is performed by coordinate. Multiplying by 0 gives the zero line corresponding to the empty set. If we add the operation of multiplying rows by elements of the Galois field $GF(2) = \{0,1\}$, then all the axioms of the linear space for the set W_G will be fulfilled. In this case, the dimension of this space is equal to $|E| = m$, and as one of the bases we can take the rows corresponding to the edges of the graph (Theorem 4.2, [12]).

The set of all simple cycles, including the null-graph and the union of edge-disjoint simple cycles of the graph over the field $GF(2)$, forms a linear subspace W_C of dimension $\nu = m - n + k$ of the space W_G (Theorem 4.3, [12]). We will call it the cycle space. Similarly, the set of all cuts corresponding to the selected spanning tree of the graph and their edge-disjoint unions of cuts over the field $GF(2)$ is a linear subspace W_S of dimension $\nu^* = n - k$ of the space W_G (Theorems 4.4, 4.5, [12]). We will call it the cut space. Moreover, the set of basic cycles and the set of basic cuts with respect to some spanning tree of a connected graph are bases, respectively, of the space of all simple cycles of the graph and their edge-disjoint unions, as well as the space of all cuts and their edge-disjoint unions, respectively. They are called cycle basis and fundamental cutsets accordingly (or cut basis).

Linear subspaces of graph: the cycle space W_C and the cut space W_S are orthogonal, moreover, they are orthogonal complements of each other (theorems 4.9 and 4.10, [12]). And any graph can be represented as a direct sum of the cycle space and the cut space (theorem 4.11, [12]).

According to the process of construction of basic cycles, it can be seen that only simple cycles can be basic. Since each of them will contain only one chord (theorem 2.10, [12]) and edges of the graph spanning tree. Moreover, each chord will be included only in one basic cycle. Therefore, the fundamental system of cycles (cycle basis) is a linearly independent system of vectors. Similarly, each cut from the fundamental cutsets of graph contains only one edge of the spanning tree (theorem 2.9, [12]). The specified edge belongs to a single cut from the fundamental cutset of graph, which ensures their linear independence.

The elements of the linear subspaces W_C and W_S are found as linear combinations of the vectors of these linearly independent systems of vectors, therefore the cycle basis and the fundamental cutsets form the bases of the corresponding vector subspaces (Theorem 4.6, [12]).

Theorem 2 [14]. For a simple graph G, any row of the matrix $C = C(G)_{\nu\times m}$ is orthogonal to any row of the matrix $S = S(G)_{\nu^*\times m}$

$$C \cdot S^T = S \cdot C^T = \mathbf{0},$$

where C^T, S^T are transposed matrices, $\mathbf{0}$ is the zero matrix of the corresponding dimension.

A similar theorem for a digraph was proved in [12] by Theorem 6.6.

Theorem 3 [14]**.** Let for some spanning tree T of a connected graph $G = (V,E)$, $|V| = n$, $|E| = m$, $k = 1$, $n \geq 3$, the matrix of cycles $C(G)$ is constructed and transformed to the form $C(G) = [\mathbf{E}_{m-n+k} \mid \mathbf{C}^*]$. Then the canonical form of the matrix of cuts can be defined as $S(G) = [\mathbf{S}^* | \mathbf{E}_{n-k}]$, where $\mathbf{S}^* = (\mathbf{C}^*)^T$.

Theorem 4 [15]**.** 1. The order of the linear subspace W_C of all simple cycles of the graph, including the null-graph, and their edge-disjoint unions (cycle space), is $2^{m\ -\ n\ +\ k}$.

2. The order of the linear subspace W_S (cut space) of all cutsets of the graph and their edge-disjoint unions (cut space) is 2^{n-k}.

Corollary 1. The number of all non-zero simple cycles and their edge-disjoint graph associations is $2^{m-n+k} - 1$. The number of all non-zero cutsets of a graph and their edge-non-intersection associations is $2^{n-k} - 1$.

Thus, using the matrix of cycles $C(G)$, we construct the matrix of fundamental cutsets of the graph $S(G)$. Using the matrices $C(G)$ and $S(G)$, we can construct the cycle space W_C of all simple cycles and their edge-disjoint unions and the cut space W_S of all cuts and their edge-disjoint unions of the graph G as the set of all possible linear combinations of rows of matrices by adding them modulo 2 over the Galois field $GF(2)$.

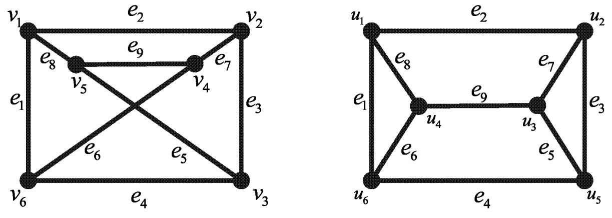
Theorem 5. The isomorphic graphs are the same:

1) ordered numerical sequences of lengths of all simple cycles and their edge-disjoint unions,

2) ordered numerical sequences of lengths of all cutsets and their edge-disjoint unions.

The proposed numerical sequences can be used as two invariants upon detection of graphs isomorphism. Using only the sequence of lengths of all simple cycles and their edge-disjoint unions does not work on graphs with vertices of degree 1.

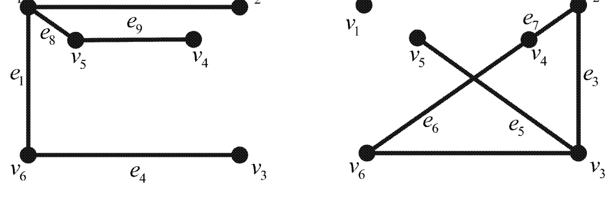
Example 1. We consider the use of the proposed invariants on a known pair of non-isomorphic graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ (figure 1).



a) G_1 b) G_2

Fig. 1. Non-isomorphic graphs G_1 and G_2

For graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ we have $|V_1| = |V_2| = 6$, $|E_1| = |E_2| = 9$, $k_1 = k_2 = 1$. Note that the graphs G_1 and G_2 are regular of degree 3, that is, the vectors of the vertices powers of the graphs are $(3;3;3;3;3;3)$. These invariants do not give an answer to the question about the isomorphism of these graphs.



a) T_1 b) T_2

Fig. 2. The spanning tree T_1 and the spanning co-tree T_1^* for the graph G_1

We construct the cycle space $W_C(G_1)$ for the graph G_1. The cyclomatic rank of the graph G_1 is $\nu(G_1) = 9 - 6 + 1 = 4$. Consequently, any spanning tree of a graph contains 5 edges, and spanning co-tree — 4 chords (edges). Take an arbitrary spanning tree of the graph T_1 (figure 2, a) and appropriate spanning co-tree T_1^* (figure 2, b). Note that the spanning tree T_1 is presented in the form of a connected tree, and the spanning co-tree of T_1^* is forest.

Attaching the chord (edge) e_3 of the spanning co-tree T_1^* to the spanning tree T_1, we obtain the cycle $C_1 = \{e_1,\ e_2,\ e_3,\ e_4\} = (111100000)$, whose length is 4, $l(C_1) = 4$. Similarly for the chord e_5 cycle is $C_2 = \{e_5,\ e_4,\ e_1, e_8\} = (100110010)$, $l(C_2) = 4$, for the chord e_6 cycle is $C_3 = \{e_6,\ e_1,\ e_8,\ e_9\} = (100001011)$, $l(C_3) = 4$, for the chord e_7 cycle is $C_4 = \{e_2,\ e_7,\ e_9,\ e_8\} = (010000111)$, $l(C_4) = 4$.

The constructed basic cycles relative to the spanning tree T_1 form a fundamental system of cycles (cycle basis). Rewrite them in the form of vectors that form the rows of the matrix of cycles:

$$C(G_1)_{4\times 9} = \begin{array}{c} \begin{array}{ccccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 & e_9 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} \end{array}.$$

Transform the matrix $C(G_1)_{4\times 9}$ to the canonical form, first writing down the columns corresponding to the spanning co-tree chords T_1^*:

$$\hat{C}(G_1)_{4\times 9} = \begin{array}{c} \begin{array}{ccccccccc} e_3 & e_5 & e_6 & e_7 & e_1 & e_2 & e_4 & e_8 & e_9 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \end{array} = [\mathbf{E}_4|\mathbf{C}^*].$$

According to Theorem 3, the matrix of basic cuts (fundamental cutsets) in the canonical form is equal to

$$S(G_1)_{5\times 9} = \begin{array}{c} \begin{array}{ccccccccc} e_3 & e_5 & e_6 & e_7 & e_1 & e_2 & e_4 & e_8 & e_9 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array} = [(\mathbf{C}^*)^T|\mathbf{E}_5].$$

The dimension of the space $W_C(G_1)$ of simple cycles and their edge-disjoint unions for the G_1 graph is 2^4, and the elements of the space are represented by vectors obtained from the matrix $C(G_1)_{4\times 9}$ by modulo-2 addition of all possible combinations of the matrix rows:

$C_5 = (011010010), C_6 = (011101011), C_7 = (101100111),$
$C_8 = (000111001), C_9 = (110110101), C_{10} = (110011100),$
$C_{11} = (111011001), C_{12} = (001010101), C_{13} = (010111110),$
$C_{14} = (001101100), C_{15} = (101011110).$

In this case, we take into account the trivial zero cycle $C_{16} = (000000000)$.

The ordered sequence of lengths of all constructed simple cycles and their edge disjoint unions of the space $W_C(G_1)$ has the form:

$$(0,4,4,4,4,4,4,4,4,6,6,6,6,6,6,6).$$

Similarly, for the graph G_2 the matrix of cycles $C(G_2)$ and the matrix of cuts $S(G_2)$ in the canonical form are constructed on the spanning tree T_2 and on the spanning co-tree T_2^* (Figure 3):

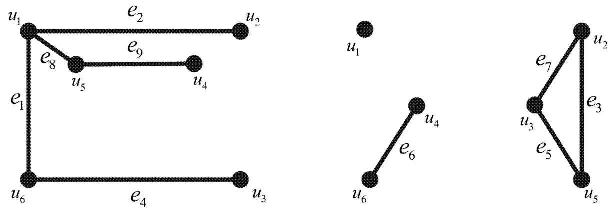
$$\hat{C}(G_2) = \begin{array}{c} \begin{array}{ccccccccc} e_3 & e_5 & e_6 & e_7 & e_1 & e_2 & e_4 & e_8 & e_9 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \end{array},$$

$$\hat{S}(G_2) = \begin{array}{c} \begin{array}{ccccccccc} e_3 & e_5 & e_6 & e_7 & e_1 & e_2 & e_4 & e_8 & e_9 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array}.$$

The ordered sequence of lengths of all constructed simple cycles and their edge disjoint unions of the cycle space $W_C(G_2)$ has the form:

$$(0,3,3,4,4,4,4,5,5,5,5,6,6,6,6,6).$$

At this step, it can be noted that the ordered sequences of the lengths of all simple cycles and their edge-disjoint unions of the cycle spaces $W_C(G_1)$ and $W_C(G_2)$ do not coincide. Therefore, there is no need to find the space of all cuts of the graph. And on the basis of the necessary condition (Theorem 5), these graphs are not isomorphic.



a) T_2 b) T_2^*

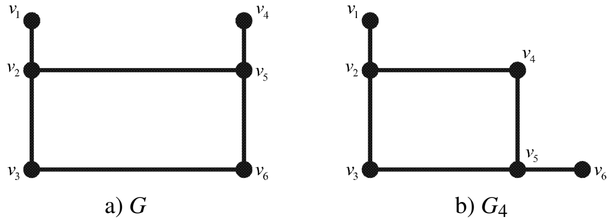
Fig. 3. The spanning tree T_2 and the spanning co-tree T_2^* for the graph G_2

Thus, to detect the isomorphism of graphs, one should construct a matrix of cycles of the graph along one of the spanning trees of the graph. With its help, you can get in the graph the whole set of simple cycles and their edge-disjoint unions. To do this, you need to find all possible sums of rows of the graph matrix of cycles according to the definition of the basis of the linear space of W_C. Such nonzero sums will be $2^{\nu-1}$, where $\nu = m - n + k$ is the cyclomatic rank of the graph. Of course, the representations of these cycles will depend on the numbering of the vertices of the graph and the order in which they are considered. But we can first determine the lengths of all simple cycles in a graph and their edge-disjoint unions. Of course, for isomorphic graphs, ordered by non-decreasing sequences of the lengths of all simple cycles and their edge-disjoint unions must coincide completely. The discrepancy allows you to immediately answer the absence of isomorphism between the graphs.

Naturally, all vertices of degree 1 will not participate in the recording of cycles of a graph, since the hanging vertices of a connected graph do not enter into one cycle and will not be represented by units in the matrix of cycles. Therefore, it is also advisable here to consider the matrix of basic cuts, since the hanging vertices will necessarily fall into the set of all cutsets of the graph and their edge-disjoint unions. Since all such cutsets and their unions form a linear space of W_S, then using the matrix of basic cuts in $2^{n-k} - 1$ steps (summation modulo 2 of all possible variants of rows of the matrix S), all cutsets of the graph and their edge-disjoint unions can be obtained.

Like the case with cycles, isomorphic graphs organized by non-decreasing sequences of all cutsets and their edge-disjoint unions must coincide completely. The discrepancy allows you to immediately answer the absence of isomorphism between the graphs.

Note that this invariant, like many well-known invariants, does not allow us to establish the absence of isomorphism for the graphs G_3 and G_4 in Figure 4.



a) G b) G_4

Fig. 4. Non-isomorphic graphs G_3 and G_4

The graphs G_3 and G_4 have the same number of vertices, edges, connected components, the same ordered power series of vertices, equal densities and non-densities, chromatic numbers, Hadwigers numbers. Also, both graphs have a single simple cycle of length 4 and 15 non-zero cutsets, the ordered sequences of lengths of which coincide. This counterexample suggests that in case of coincidence of ordered sequences of lengths of vectors of spaces W_C and W_S, additional studies are required. Therefore, the coincidence of these new invariants is also only a necessary condition of graph isomorphism.

Thus, as invariants of a graph, it is recommended to consider the vector of degrees of the vertices of the graph, together with the ordered sequences of lengths of simple cycles and cutsets of the graph. However, with the full coincidence of these quantities, it is possible to make an assumption about the isomorphism of the graphs under consideration, but additional research will be required.

It is possible for graphs to establish an isomorphism using the following heuristic algorithm.

Let G and G^* are isomorphic graphs with different numbering of the vertices. It is necessary to construct vertex classification trees for establishing a one-to-one correspondence between the vertex sets of graphs G and G^*. Construction of the classification tree of the graph $G = (V, E)$, $V = \{v_1, v_2, \dots, v_n\}$, $|V| = n$, $|E| = m$, $k = 1$, consists of the following steps.

(1) construction of spaces W_C of simple cycles (and their edge-disjoint unions) and W_S of simple cuts (and their edge-disjoint unions) of graph G;

(2) establishing the incidence property of each vertex of the graph to some edges forming cycle of the space W_C or a cut of the space W_S;

(3) construction of matrix $X = [x_{ij}]$ "vertex-property". The cycles and the cuts of the spaces W_C and W_S respectively are considered as properties forming the rows of the matrix. The element x_{ij} of the matrix takes the value 1 if the vertex v_j is incident

to some edge of the cycle or cut located in the i-th row, and 0 otherwise, where $i = 1, \dots, p$, $j = 1, \dots, n$, $p = 2^{m-n+k} + 2^{n-k} - 2$;

(4) calculation of the sum of elements in each column of the matrix X: $Q_j = \sum_{i=1}^{p} x_{ij}$, $j = 1, \dots, n$;

(5) sorting the columns of the matrix X in non-decreasing order of vertex degrees $deg\ v_j$, $j = 1, \dots, n$. Columns corresponding to vertices with equal degrees, it is also desirable to sort in non-decreasing order of the sums of Q_j. Let's redefine the obtained sequence of vertices as $x_1, x_2, \dots, x_n$;

(6) construction of the similarity matrix $H = [h_{jk}]$ of vertices $x_1, x_2, \dots, x_n$, where the element h_{jk} is the "distance" between vertices x_j and x_k, calculated by the formula $h_{jk} = \sum_{i=1}^{p} |x_{ij} - x_{ik}|$ for $j, k = 1, \dots, n$;

(7) dividing the set of vertices using the similarity matrix, assuming that initially all vertices belong to the same class: $K_1 = \{x_1, x_2, \dots, x_n\}$. To do this, two vertices x_l and x_q are defined, the difference between them is the greatest, that is, $h_{jk} = max\{h_{ij}\}$. The resulting vertices are considered to be the centers of two new classes K_l and K_q. Other vertices are divided into these classes by the degree of proximity to their centers. If the centers of the classes are the vertices x_l and x_q forming the classes K_l and K_q respectively, then the vertex $x_r \in K_l$ if $h_{rl} < h_{rq}$, or $x_r \in K_q$ otherwise. As a result, all vertices will be distributed between two classes K_l and K_q. Further separation of vertices classes K_l and K_q takes place by a similar procedure. If at some step it turned out that there are several vertices with equal distances to the centers in the classes, then the degree of vertex deg x_j and Q_j should be taken into account when dividing them.

The split process ends when there is only one vertex in each class.

This procedure allows us to construct a classification tree K^*, whose hanging vertices correspond to the vertices $x_1, x_2, \dots, x_n$ and, consequently, to the vertices $v_1, v_2, \dots, v_n$ of the original graph G.

Similarly, we construct a classification tree K^* of the graph $G^* = (U, E^*)$, $U = \{u_1, u_2, \dots, u_n\}$, $|U| = n$, $|E| = m$, $k = 1$, whose hanging vertices are $u_1, u_2, \dots, u_n$.

With the help of known algorithms for establishing isomorphism of trees, a one-to-one mapping between sets of graph vertices is constructed.

The work of the proposed algorithm is considered by an example.

Example 2. Establish a one-to-one correspondence between vertices of isomorphic graphs $G_5 = (V, E_5)$ and $G_6 = (U, E_6)$, $|V| = |U| = 5$, $|E_5| = |E_6| = 6$, $k(G_5) = k(G_6) = 1$ (figure 5) using vector spaces of simple cycles (and their edge-disjoint unions) and cuts (and their edge-disjoint unions) W_C and W_S.

Choosing the spanning trees T_5 and T_6 of graphs G_5 and G_6 (figure 5), we obtain the following matrix of basis cycles

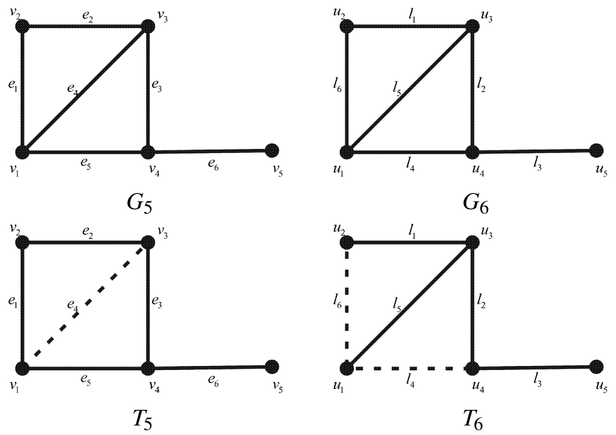


Fig. 5. Isomorphic graphs G_5 and G_6 and their spanning trees T_5 and T_6

$$C(G_5) = \begin{pmatrix} 1\,1\,1\,0\,1\,0 \\ 1\,1\,0\,1\,0\,0 \end{pmatrix},$$

$$C(G_6) = \begin{pmatrix} 0\,1\,0\,1\,1\,0 \\ 1\,0\,0\,0\,1\,1 \end{pmatrix}.$$

Next, we find the corresponding matrix of basic cuts

$$S(G_5) = \begin{pmatrix} 1\,1\,1\,0\,0\,0 \\ 1\,1\,0\,1\,0\,0 \\ 1\,0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,0\,1 \end{pmatrix},$$

$$S(G_6) = \begin{pmatrix} 0\,1\,1\,0\,0\,0 \\ 1\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,1\,0 \\ 1\,1\,0\,0\,0\,1 \end{pmatrix}.$$

Using matrices $C(G_5)$, $S(G_5)$ the corresponding spaces $W_C(G_5)$ and $W_S(G_5)$ are constructed. Let's form a matrix "vertex-property" $X(G_5)$ in accordance with point (3) of the algorithm

X(G_5)	v_1	v_2	v_3	v_4	v_5
C_1	1	1	1	1	0
C_2	1	1	1	0	0
C_3	1	0	1	1	0
S_1	1	1	1	1	0
S_2	1	1	1	1	0
S_3	1	0	1	1	0
S_4	0	0	0	1	1
S_5	1	1	1	0	0
S_6	1	1	1	1	0
S_7	1	1	1	1	1
S_8	1	1	1	1	0
S_9	1	1	1	1	1
S_{10}	1	0	1	1	1
S_{11}	1	1	1	1	0
S_{12}	1	1	1	1	1
S_{13}	1	1	1	1	1
S_{14}	1	1	1	1	1
S_{15}	1	1	1	1	1
Q_j	17	14	17	16	8

After ordering the vertices by non-decreasing degrees, we find the matrix of similarity of the vertices of the graph G_5

$$H(G_5) = \hat{S}(G_2) = \begin{array}{c} \begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \end{array} \\ \begin{pmatrix} 0 & 10 & 11 & 11 & 8 \\ 10 & 0 & 3 & 3 & 6 \\ 11 & 3 & 0 & 0 & 3 \\ 11 & 3 & 0 & 0 & 3 \\ 8 & 6 & 3 & 3 & 0 \end{pmatrix} \end{array},$$

where $x_1 \to v_5, x_2 \to v_2, x_3 \to v_1, x_4 \to v_3, x_5 \to x_4$.

Using the matrix $H(G_5)$, taking into account the values of Q_j, we construct a classification tree of vertices of the graph G_5 (figure 6, a). Similarly, we obtain the vertex classification tree of the graph G_6 (figure 6, b).

Comparing the classification trees $K(G_5)$ and $K(G_6)$ taking into account the automorphism of graph vertices, we obtain the following one-to-one mapping between the vertex sets of graphs G_5 and G_6 preserving adjacency: $v_1 \leftrightarrow u_1$, $v_2 \leftrightarrow u_5$, $v_3 \leftrightarrow u_4$, $v_4 \leftrightarrow u_1$, $v_5 \leftrightarrow u_3$.

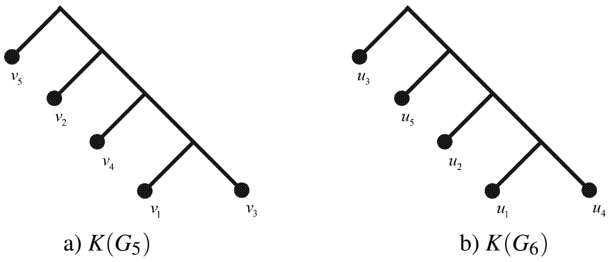


Fig. 6. Classification tree of the vertices in isomorphic graphs G_5 and G_6

The developed new invariant can be used in the analysis of graph structures of big data. The proposed heuristic algorithm makes it possible to establish a bijection between sets of vertices of isomorphic graphs when solving applied problems.

4. Conclusions

The article describes the following new statements.

1. The linear spaces of all simple cycles and their edge disjoint unions of a simple graph (cycle space), as well as of all cuts of the graph and their edge disjoint unions (cut space) are constructed. A estimate of the dimensions of these spaces is given.

2. The canonical form of matrices of basis cycles and basis cuts of the graph is described. The method of finding these matrices is given.

3. A new invariant of detecting isomorphism of graphs in the form of ordered by non-decreasing sequences of lengths of all simple cycles of the graph (and their edge-disjoint unions) and all sections of the graph (and their edge-disjoint unions) is proposed. It is shown that the coincidence of this invariants gives a new necessary condition for isomorphism of graphs.

4. We propose a heuristic algorithm for constructing a one-to-one correspondence between sets of vertices of isomorphic graphs using the linear spaces of all simple cycles and their edge disjoint unions of a simple graph (cycle space), as well as of all cuts of the graph and their edge disjoint unions (cut space).

References

[1] Нечепуренко М.И.: Алгоритмы и программы решения задач на графах и сетях Новосибирск, Наука, 1990.

[2] Карелин, В.П.: Теория и средства поддержки принятия решений в организационно-технологических системах: дис. ... д-ра техн. наука, Таганрог, ТРТУ, 1995.

[3] Бернштейн Л.С., Карелин В.П., Целых А.Н.: Модели и методы принятия решений в интегрированных интеллектуальных системах, Ростов/Д, Изд-во РГУ, 1999.

[4] Пинчук В.П.: Табличные инварианты на графах и их применение // Кибернетика и системный анализ, Институт кибернетики НАН Украины, 2001, № 4.

[5] Карелин В.П.: Задача распознавания изоморфизма графов. Прикладное значение и подходы к решению, Вестник Таганрогского института управления и экономики, 2015, № 1, С. 102–106.

[6] Пономаренко И.Н.: Проблема изоморфизма графов: Алгоритмические аспекты, СПб., 2010, 57 с.

[7] Погребной Ан.В., Погребной В.К.: Метод дифференциации вершин графа и решение проблемы изоморфизма // Известия Томского политехнического университета. Инжиниринг георесурсов. – 2015. – Т. 326. – № 6. – С. 34–45.

[8] Батенков К.А. Числовые характеристики структур сетей связи. Труды СПИИРАН, 2017, Вып. 53, С. 5–28.

[9] L. Babai, Eugene M. Luks: Canonical labeling of graphs. Proc. 15th ACM Symp. on Theory of Computing, 1983, pp. 171–183.

[10] Leszly Babai: Graph Isomorphism in Quasipolynomial Time . Submitted 19 January, 2016. –V. 1 submitted 11 December, 2015; originally announced December 2015. [https://arxiv.org/search/cs?searchtype=author&query=Babai%2C+L].

[11] Ландо С.К.: Графы и топология. М., ВШЭ, 2018., 78 с.

[12] Свами М., Тхуласираман К.: Графы, сети и алгоритмы. М., Книга по Требованию, 2013, 450 с.

[13] Носов В.И., Бернштейн Т.В., Носкова Н.В., Храмова Т.В. Элементы теории графов: Учебное пособие под редакцией В.И. Носова, СибГУТИ, Новосибирск, 2008, 107 с.

[14] Кристофидес Н. Теория графов. Алгоритмический подход. Издательство: Мир, 1978. 432 с.

[15] Горбатов В.А. Фундаментальные основы дискретной математики. Учебное пособие. М.: Наука. Физматлит, 544 с.

[16] Kolasinska E. On a Minimum Cycle Basis of a Graph, Zastosowania Matematyki, 16, 4 (1980), p. 631–639.

KILKA POMYSŁÓW NA TEMAT IZOMORFIZMU POŁĄCZONYCH WYKRESÓW

Streszczenie W artykule badamy istnienie izomorfizmów między grafami oraz poszukujemy niezmienników grafów spójnych. Tworzony jest nowy niezmienniczy graf. Metoda może słuïyć do wykrywania izomorfizmów między grafami spójnymi. W pracy użyto pojęcia przestrzeni wektorowej wszystkich prostych cykli grafu i ich sum względem rozłącznych krawędzi oraz przestrzeni wektorowej wszystkich zbiorów grafów uciętych i ich rozłącznych krawędziowo sum. Zbadano metodę konstruowania takich przestrzeni wektorowych: przestrzeni cyklicznej i przestrzeni cięcia. Podano nowe oszacowanie wymiarów tych tego typu przestrzeni wektorowych grafów. Otrzymany niezmiennik jest pokazany na konkretnym przykładzie. W pracy podano kontrprzykład, aby potwierdzić fakt, że zaproponowany niezmiennik może być użyty jako warunek konieczny, ale niewystarczający dla izomorfizmu grafów.

Słowa kluczowe: wykres, cykl, zestaw tnący, przestrzeń wektorowa, niezmienny, algorytm izomorfizmu