
HOMO RES SACRA

Russian-Armenian University

Printed by the decision of the Academic Council of RAU

V E S T N I K

OF RUSSIAN-ARMENIAN UNIVERSITY

(SERIES: PHYSICAL-MATHEMATICAL AND NATURAL SCIENCES)

(42)
RAU University Press
No. 2/2022

Российско-Армянский университет

Печатается по решению
Ученого Совета РАУ

В Е С Т Н И К

РОССИЙСКО-АРМЯНСКОГО УНИВЕРСИТЕТА

(СЕРИЯ: ФИЗИКО-МАТЕМАТИЧЕСКИЕ И ЕСТЕСТВЕННЫЕ НАУКИ)

(42)
Издательство РАУ
№ 2/2022

Вестник РАУ, №2. – Ер.: Изд-во РАУ, 2022.– 89 с.

Журнал основан в 2003 году и входит в перечень периодических изданий, зарегистрированных ВАК РА и РИНЦ

Российско-Армянский университет, 2022г.
ISSN1829-0450

Vestnik of the RAU, No.2. – Yer.: RAU Publishing House, 2022. – 89 p.

The journal founded in 2003 is included in the list of periodicals registered by HAC RA and RSCI.

Russian-Armenian University, 2022
ISSN1829-0450

СОДЕРЖАНИЕ

CONTENT

Математика и информатика

Механика

МАТЕМАТИКА И ИНФОРМАТИКА

DOI 10.48200/1829-0450_pmn_2022_2_7
UDC 519.174.7

Поступила: 29.03.2022г.
Сдана на рецензию: 30.03.2022г.
Подписана к печати: 13.04.2022г.

VERTEX-DISTINGUISHING EDGE COLORINGS OF SOME COMPLETE TRIPARTITE GRAPHS

T. Petrosyan, A. Drambyan

Russian-Armenian University

tigran.petrosyan@student.rau.am, ardrambyan@student.rau.am

ABSTRACT

A proper edge coloring of a graph G is a mapping $f: E(G) \longrightarrow \mathbb{Z}_{\geq 0}$ such that $f(e) \neq f(e')$ for every pair of adjacent edges e and e' in G. A proper edge coloring f of a graph G is called vertex-distinguishing if for any different vertices $u, v \in V(G)$, $S(u, f) \neq S(v, f)$, where $S(v, f) = \{f(e) \mid e = uv \in E(G)\}$. The minimum number of colors required for a vertex-distinguishing proper edge coloring of a simple graph G is denoted by $\chi'_{vd}(G)$. A graph G is called *complete r-partite* ($r \geq 2$) if its vertices can be partitioned into r non-empty independent sets $V_1, \dots, V_r$ such that each vertex in V_i is adjacent to all the other vertices in V_j for $1 \leq i < j \leq r$. Problems in which we are interested are particular cases of the great variety of different ways of labeling a graph. In this paper we provide lower and upper bounds on $\chi'_{vd}(G)$ for some complete tripartite graphs.

Keywords: edge coloring, vertex-distinguishing coloring, chromatic index, complete multipartite graph.

Introduction

All graphs considered in this paper are finite and simple, and we use West's book [6] for terminologies and notations not defined here. Let $G = (V, E)$ be a graph of order n with the vertex set $V = V(G)$ and the edge set $E = E(G)$. A k-edge coloring of a graph G is an assignment of k colors to the edges of G. Let $f(e)$ be the color of the edge e. Denote by $S(v, f) = \{f(e) \mid e = uv \in E(G)\}$ the multiset of colors assigned to the set of edges incident to v. The coloring f is proper if no two adjacent edges are assigned the same color and vertex-distinguishing proper coloring (abbreviated $VDP - coloring$), if it is proper and $S(u, f) \neq S(v, f)$ for any two distinct vertices u and v.

The $VDP - coloring$ has been considered in many papers. It was introduced and studied by Burris and Schelp in [2] and, independently, as observability of a graph, by Cerny et al., Hornãk and Sotãk [3]. In [2,4], the $VDP - coloring$ is also computed for some families of graphs, such as complete graphs K_n, bipartite complete graphs $K_{m,n}$, paths P_n, and cycles C_n. The following results has been proved by Burris and Schelp [2]. by Bazgan et al. [5].

Theorem 1. *Let n be any natural number. Then*

$$\chi_{vd}'(K_n) = \begin{cases} n & if \quad n \ is \ odd; \\ n+1 & if \quad n \ is \ even. \end{cases}$$

Theorem 2. *Let m and n be any natural numbers. Then*

$$\chi_{vd}'(K_{m,n}) = \begin{cases} n+1 & if \quad n > m \geq 2; \\ n+2 & if \quad n = m \geq 2. \end{cases}$$

The original motivation of study is generalizing results for $VDP - coloring$. In this work we obtain some results on vertex-distinguishing edge colorings of complete 3-,4- partite and multipartite graphs.

Main results

Let $K_{n_1,n_2,\dots,n_r}$ denote a complete r-partite graph with independent sets $V_1, V_2, \dots, V_r$ of sizes $n_1, n_2, \dots, n_r$. If f is a vertex distinguishing proper edge-coloring of G and $v \in V(G)$, then $S(v,f)$ denotes the set of colors of labeled edges incident to v.

Lemma 1. For any natural numbers m and n

$$\chi_{vd}{}'(K_{1,m,n}) = \begin{cases} m+n & if \quad n \geq m > 1; \\ m+n+1 & if \quad n \geq m = 1. \end{cases}$$

Proof

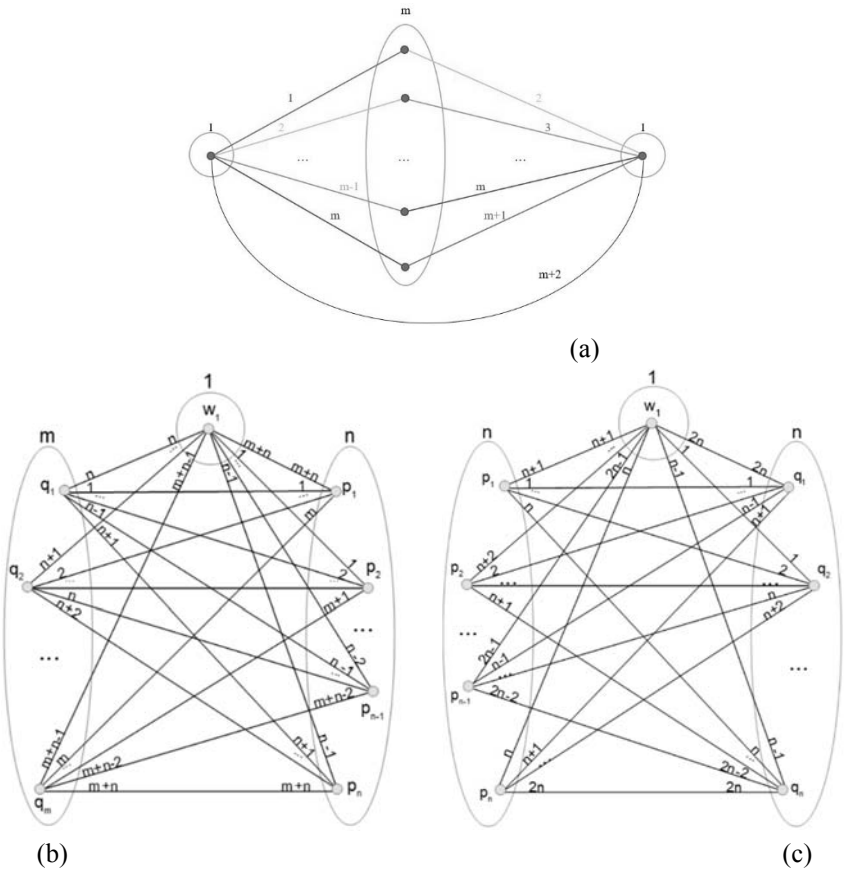


Fig. 1. The VDP-coloring of graph $K_{m,n,1}$.

Let $V(G) = \{w_1\} \cup P \cup Q$ be the vertex set of graph G, where w_1 is the 1-partition vertex, $P = \{p_1, p_2, \ldots, p_n\}$ is the set of n-part vertices and $Q = \{q_1, q_2, \ldots, q_m\}$ is the set of m-part vertices and $E(G) = \{w_1 p_i \mid 1 <= i <= n\} \cup \{w_1 q_i \mid 1 <= i <= m\} \cup \{p_i q_j \mid 1 <= i <= n \,\&\, 1 <= j <= m\}$ be the set of edges. We shall consider the following three cases.

Case 1. $m \geq n$ and $n = 1$

The algorithm of coloring with $m + 2$ colors for this case is presented in *Fig. 1a*. For each edge $v_i v_j \in E(K_{m,1,1})$, define a color $f(v_i v_j)$ as follows:

$$f(v_i v_j) = \begin{cases} i & if \quad v_i \in Q, \quad v_j = p_1; \\ i+1 & if \quad v_i \in Q, \quad v_j = w_1; \\ m+2 & if \quad v_i = p_1, \quad v_j = w_1. \end{cases}$$

Let us prove that f is a vertex distinguishing $(m + 2)-$coloring of $K_{m,1,1}$.

Note that

$$S(v_i, f) = \begin{cases} [1, m] \cup \{m+2\} & if \quad v_i \in P; \\ [2, m+2] & if \quad v_i \in W; \\ \{i; i+1\} & if \quad v_i \in Q. \end{cases}$$

Therefore, for each pair of vertices v_i and v_j $S(v_i, f) \neq S(v_j, f)$.

Finally, let's prove that we need at least $m + 2$ colors for VDP-coloring. There are $m + 1$ edges incident to each of the 1-partition vertices, so we need at least $m + 1$ colors for labeling edges. As the set of colors incident to that vertices must be different, we need to use at least $m + 2$ colors.

Case 2. $n \neq m$ and $m > 1$

The algorithm of coloring with $m + n$ colors for this case is presented in *Fig. 1b*. For each edge $v_i v_j \in E(K_{m,n,1})$, define a color $f(v_i v_j)$ as follows:

$$f(v_iv_j)=\begin{cases} i+j-1 & if \quad v_i\in Q, \quad v_j\in P;\\ n+i-1 & if \quad v_i\in Q, \quad v_j=w_1;\\ m+n & if \quad v_i=p_0, \quad v_j=w_1;\\ i & if \quad v_i\in P\backslash\{p_0\}, \quad v_j=w_1. \end{cases}$$

By the definition of f, we have

$$S(v_i,f)=\begin{cases} [i,n+i] & if \quad v_i\in Q;\\ [i,m+i-1]\cup\{m+n\} & if \quad v_i=p_0;\\ [i-1,m+i-1] & if \quad v_i\in P\backslash\{p_0\};\\ [1,m+n] & if \quad v_i\in W. \end{cases}$$

1-partition vertex is connected to $m+n$ vertices from different partitions, so we need to use at least $m+n$ colors for proper coloring.

Case 3. $n=m$ and $m>1$

The algorithm of coloring for this case, using $2n$ colors is presented in *Fig. 1c*. For each edge $v_iv_j\in E(K_{n,n,1})$, define a color $f(v_iv_j)$ as follows:

$$f(v_iv_j)=\begin{cases} i+j-1 & if \quad v_i\in P, \quad v_j\in Q;\\ n+i & if \quad v_i\in P\backslash\{p_n\}, \quad v_j=w_1;\\ n & if \quad v_i=p_n, \quad v_j=w_1.\\ i-1 & if \quad v_i\in Q\backslash\{q_1\}, \quad v_j=w_1;\\ 2n & if \quad v_i=q_1, \quad v_j=w_1; \end{cases}$$

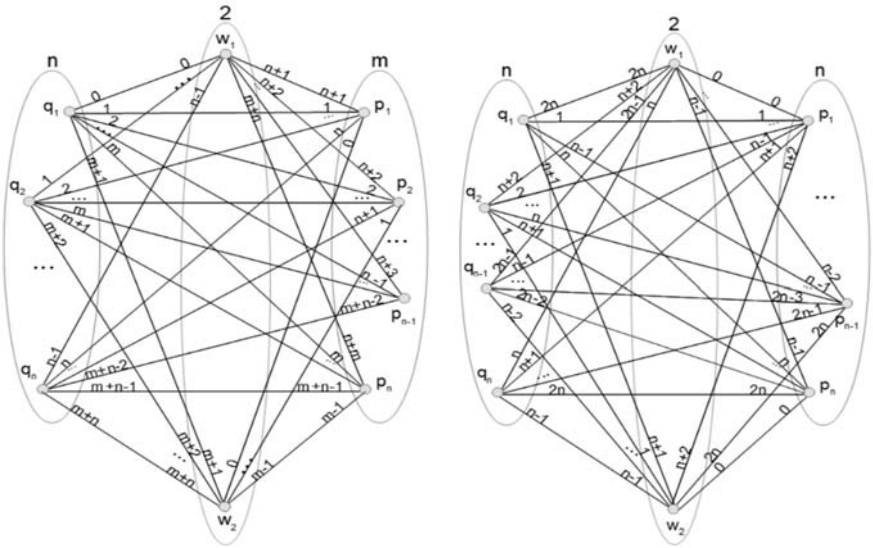
By the definition of f, we obtain

$$S(v_i,f)=\begin{cases} [i,n+i] & if \quad v_i\in P;\\ [1,n-1]\cup\{n+1\}\cup\{2n\} & if \quad v_i\in P;\\ [i,n+i-1]\cup\{n+i+1\} & if \quad v_i\in Q;\\ [0,j-1]\cup[j+1,n+m] \ \ (j\in\{n,m\}) & if \quad v_i\in W. \end{cases}$$

1-partition vertex is connected to $2n$ vertices from different partitions, so we need to use at least $2n$ colors for proper coloring.

Lemma 2. For any natural numbers m and n, we have:

$$\chi_{vd}{}'(K_{2,m,n}) = \begin{cases} m+n+1 & for \quad n \geq m > 2 \ or \ n > m = 2 \\ 6 & for \quad n = m = 2 \end{cases}$$



(a) (b)

Fig. 2. The VDP-coloring of graph $K_{m,n,2}$.

Proof

If any partition has 1 vertex, then It comes to *Lemma 1.*

Let $V(G) = W \cup P \cup Q$ be the vertex set of graph G, where $W = \{w_1, w_2\}$ is set of 2-partition vertices, $P = \{p_1, p_2, \ldots, p_n\}$ is set of n-part vertices and $Q = \{q_1, q_2, \ldots, q_m\}$ is set of m-part vertices and $E(G) = \{w_i p_j \mid i \in \{1;2\} \ \& \ 1 <= j <= n\} \cup \{w_i q_j \mid i \in \{1;2\} \ \& \ 1 <= j <= m\} \cup \{p_i q_j \mid 1 <= i <= n \ \& \ 1 <= j <= m\}$ be the set of edges. We shall consider the following three cases.

Case 1. $n > m$ and $m \geq 2$

The algorithm of coloring with $m+n+1$ colors for this case is presented in *Fig. 2a*. For each edge $v_iv_j \in E(K_{2n})$, define a color $f(v_iv_j)$ as follows:

$$f(v_iv_j)=\begin{cases} i+j-1 & if \quad v_i \in Q, \quad v_j \in P; \\ (j-1)\cdot(m+1)+i-1 & if \quad v_i \in Q, \quad v_j \in W; \\ (2-j)\cdot(n+1)+i-1 & if \quad v_i \in P, \quad v_j \in W. \end{cases}$$

Let us prove that f is an VDP-coloring of $K_{m,n,2}$.

Note that

$$S(v_i,f)=\begin{cases} [i-1,m+i] & if \quad v_i \in Q; \\ [i-1,n+i] & if \quad v_i \in P; \\ [0,j-1]\cup[j+1,n+m] \ \ (j\in\{n,m\}) & if \quad v_i \in W. \end{cases}$$

It is easy to see that $S(v_i,f) \neq S(v_j,f)$ for any vertices v_i and v_j from different partitions.

Case 2. $n=m\geq 2$

$$f(v_iv_j)=\begin{cases} i+j-1 & if \quad v_i \in Q \ \ (i\in[1,n-1]), \quad v_j \in P; \\ j\cdot n+i & if \quad v_i \in Q\backslash\{q_1\}, \quad v_j \in W; \\ j\cdot n+i & if \quad v_i \in P, \quad v_j \in W. \end{cases}$$

By the definition of f, we have

$$S(v_i,f)=\begin{cases} [1,n+1]\cup\{2n\} & if \quad v_i=q_0; \\ [i-1,n+i] & if \quad v_i \in Q\backslash\{q_0\}; \\ [i-1,n+i-1]\cup\{n+i;n+i+1 \ mod \ 2n\} & if \quad v_i \in P; \\ [0,n]\cup[n+2,2n] & if \quad v_i=w_1; \\ [0,n-1]\cup[n+1,2n] & if \quad v_i=w_2. \end{cases}$$

The algorithm of coloring for this case, using $2n+1$ colors is presented in *Fig. 2b*. It is easy to see that we have $VDP-coloring$ that is

using set of color $[0,2m]$ that is $2n+1$ colors. But we need at least that amount of colors that is why equality for chromatic index holds.

Case 3. $n = m = 2$

First let's prove that there is no $VDP-coloring$ of graph $K_{2,2,2}$ with 5 colors. Our graph has 6 vertices, which are incident to 4 edges from other partitions. The maximum number of 4-element sets with 5 elements, is $\binom{5}{4} = 5$. So at least 2 vertices will have the same set of edge colors. So we need to use at least 6 colors for proper coloring of our graph. Denote vertices of our graph by $W = \{w_0, w_1, w_2, w_3, w_4, w_5\}$ and define coloring f as $f(w_i w_j) = (i+j) \ mod \ 6$ for each $w_i, w_j \in W$. It is easy to see that f is a VDP-coloring for our graph.

Theorem 3. *For any natural numbers l,m and n, we have*

$$\chi_{vd}'(K_{l,m,n}) \quad l+m+n.$$

Note that if $min(m,n,l) = 1$ then it comes to *Lemma. 1*.

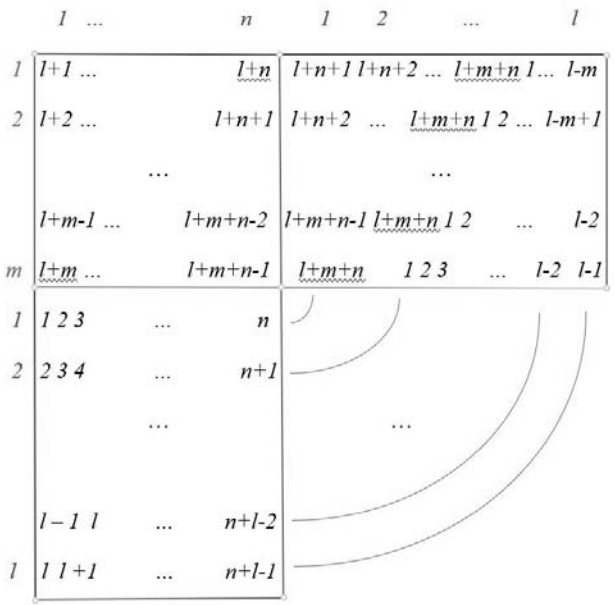


Fig. 3. The VDP-coloring of $K_{m,n,l}$ graph with $m+n+l$ colors.

Without loss of generality, let $l \geq m \geq n$ and $V(G) = W \cup P \cup Q$ be the vertex set of graph G, where $W = \{w_1, w_2, \ldots, w_l\}$ is set of l-partition vertices, $P = \{p_1, p_2, \ldots, p_n\}$ is set of n-part vertices and $Q = \{q_1, q_2, \ldots, q_m\}$ is set of m-part vertices. $E(G) = \{w_i p_j \mid 1 <= i <= l \ \& \ 1 <= j <= n\} \cup \{w_i q_j \mid 1 <= i <= l \ \& \ 1 <= j <= m\} \cup \{p_i q_j \mid 1 <= i <= n \ \& \ 1 <= j <= m\}$ is the set of graph edges.

The algorithm of coloring is presented in *Fig. 3*, where first n columns are set of edge colors, incident to vertices of $n - partition$, and first m rows are set of colors of edges, incident to vertices of $m - partition$. The colors set union of $m + i$-th column with $n + i$-th row is set of colors incident to i-th vertex in $l - partition$.($i \in [1, l]$).

For each edge $v_i, v_j \in E(K_{l,m,n})$, define a color $f(v_i v_j)$ as follows:

$$f(v_i v_j) = \begin{cases} l + i + j - 1 & if \quad v_i \in P, \quad v_j \in Q; \\ i + j - 1 & if \quad v_i \in P, \quad v_j \in W; \\ [(l + n + i + j - 1) \ mod \ (l + m + n)] + 1 & if \quad v_i \in Q, \quad v_j \in W. \end{cases}$$

Then we have.

$$S(v_i, f) = \begin{cases} [i, l + m + i - 1] & if \quad v_i \in P; \\ [1, l - m + 1] \cup [l + m - i, l + m + n] & if \quad v_i \in Q; \\ [1, n + i - 1] \cup [l + n + i, l + m + n] & if \quad v_i \in W (i \leq m); \\ [i - m, i - 1] \cup [i, n + i - 1] & if \quad v_i \in W (i > m): \end{cases}$$

We constructed a coloring using $m + n + l$ colors. It is easy to see that the edge coloring is vertex-distinguishing. Vertices of $l - partition$ are connected with $m + n$ vertices from $m - partition$ and $n - partition$, and sets of edge colors for vertices are different by $VDP - coloring$ definition, so we need to use at least $m + n + 1$ colors. Thus, we have $m + n + 1 \leq \chi_{vd}'(K_{l,m,n})$. In general, this result can't be improved, because for some 3-partite complete graphs chromatic index is equal to $m + n + l$ (See *Lemma. 1 Case 1.*).

Theorem 4. *Let l, m and n be natural numbers, such that $m \geq n + l$ $(n > l > 1)$. Then*

$$\chi_{vd}'(K_{m,n,l}) = m + n + 1$$

Let $V(G) = W \cup P \cup Q$ be the vertex set of graph G, where $W = \{w_1, w_2, \ldots, w_l\}$ is set of l-partition vertices, $P = \{p_1, p_2, \ldots, p_n\}$ is set of n-partition vertices and $Q = \{q_1, q_2, \ldots, q_m\}$ is set of m-partition vertices and $E(G) = \{w_i p_j \mid 1 <= i <= l \,\&\, 1 <= j <= n\} \cup \{w_i q_j \mid 1 <= i <= l \,\&\, 1 <= j <= m\} \cup \{p_i q_j \mid 1 <= i <= n \,\&\, 1 <= j <= m\}$ be the set of graph edges.

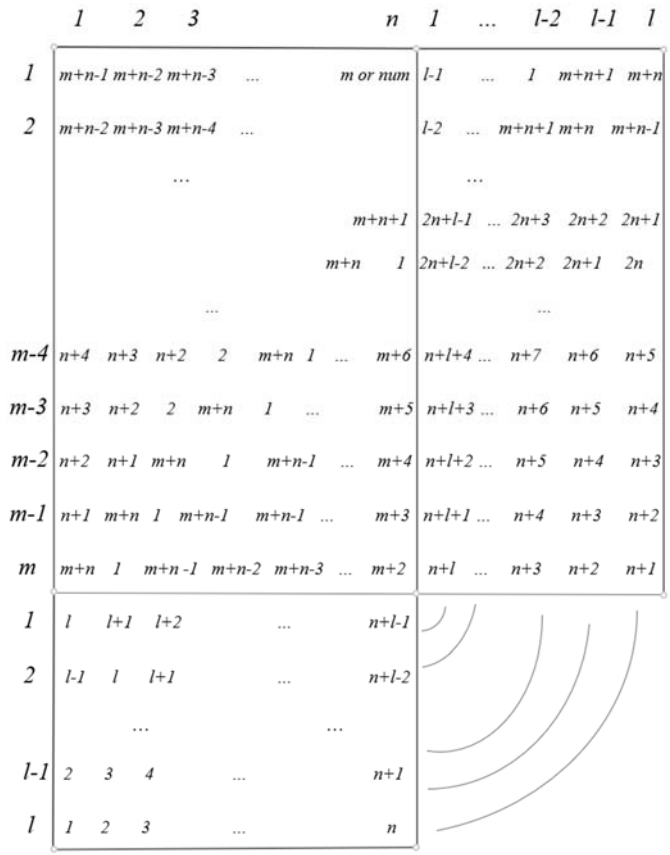


Fig. 4. The VDP-coloring of $K_{m,n,l}$ graph with $m + n + 1$ colors.

The algorithm of coloring is presented in *Fig. 4*.

For each edge $v_iv_j \in E(K_{m,n,l})$, define a color $f(v_iv_j)$ as follows:

$$f(v_iv_j) = \begin{cases} m+n-i-2 & if \quad v_i \in P, \quad v_j \in Q, i > m-j+2; \\ 1 & if \quad v_i \in P, \quad v_j \in Q, i = m-j+2; \\ m+n & if \quad v_i \in P, \quad v_j \in Q, i = m-j+3; \\ l-j+1+i & if \quad v_i \in P, \quad v_j \in W; \\ j \cdot n+i & if \quad v_i \in Q, \quad v_j \in W. \end{cases}$$

Then we have.

$$S(v_i, f) = \begin{cases} [1, m+n+1]/\{(m+n+l-1)\mathrm{mod}(m+n+2)\} & if \quad v_i \in W; \\ [1, l+i-1] \cup \{m+n\} \cup [n+\lfloor\frac{i+1}{2}\rfloor, m+n-i] & if \quad v_i \in P; \\ [n+\left\lfloor\frac{i}{2}\right\rfloor, n+l+i-1] \cup [1, \lfloor\frac{i}{2}\rfloor] & if \quad v_i \in Q. \end{cases}$$

The algorithm of coloring is shown in *Fig.4*. Each vertex of $l-partition$ is connected with $m+n$ vertices and we have $l > 1$, so for VDP-coloring we need to use at least $m+n+1$ colors. In this case we found an exact value of chromatic index.

Acknowledgment. I would like to thank Petros Petrosyan for expertise and assistance throughout all aspects of my research.

REFERENCES

1. *Burri A.* Vertex-distinguishing edge-colorings. Ph.D. Dissertation, Memphis State University (1993).
2. *Burris A., Schelp R.* Vertex-distinguishing proper edge-colorings. J. Graph Theory 26, 73–82 (1997).
3. *Cerny J., Hornak, M., Sotak R.* Observability of a graph. Math. Slovaca 46, 21–31 (1996).
4. *Hornak, M., Sotak, R.* Observability of complete multipartite graphs with equipotent parts. Ars Comb. 41, 289–301 (1995).

5. *Hornak M., Sotak R.* Asymptotic behavior of the observability of Q_n, Discrete Math. 176, 139–148 (1997).
6. *West D.* Introduction to Graph Theory - Prentice-Hall, New Jersey, 2001.

ВЕРШИННО-РАЗЛИЧАЮЩИЕ РЁБЕРНЫЕ РАСКРАСКИ ПОЛНЫХ ТРЁХДОЛЬНЫХ ГРАФОВ

Т. Петросян, А. Драмбян

Российско-Армянский университет

АННОТАЦИЯ

Для графа G функция $f: E(G) \to N$ называется рёберной раскраской графа G. Рёберная раскраска f графа G называется правильной, если для любых смежных рёбер $e, e' \in E(G)$, $f(e) \neq f(e')$. Если f – правильная раскраска графа G и $v \in V(G)$, то обозначим через $S(v, f)$ множество цветов рёбер, инцидентных вершин v. Правильная раскраска f графа G называется вершинно-различающей, если для любых различных вершин $u, v \in V(G)$, $S(u, f) \neq S(v, f)$. Наименьшее число цветов, необходимое для вершинно-различающей рёберной раскраски графа G называется вершинно-различающим хроматическим индексом и обозначается $\chi'_{vd}(G)$. Граф G вершины которой можно представить в виде объединения r независимых, непустых множеств вершин $V_1, \dots, V_r$, таких что каждая вершина из V_i смежна со всеми вершинами из V_j для $1 \leq i < j \leq r$ называется полный $r-$ дольный граф. В данной работе найдены некоторые верхние оценки вершинно-различающего хроматического индекса полных трёхдольных графов.

Ключевые слова: раскраска ребер, вершинно-различающие раскраски, хроматический индекс, полный многодольный граф.

DOI 10.48200/1829-0450_pmn_2022_2_19
УДК 621.391.15

Поступила: 07.03.2022г.
Сдана на рецензию: 29.03.2022г.
Подписана к печати: 28.04.2022г.

A PROBLEM ON THE AMOUNTS OF THE SAME CONSECUTIVE PAIRS IN BOOLEAN VECTORS

N. Khandanyan, H. Sahakyan, Zh. Margaryan

Yerevan State University

khandanyan1998@gmail.com, hovhannes1417@gmail.com, jiromr@mail.ru

ABSTRACT

In this paper, we map the set of n-dimensional boolean vectors onto the set of non-negative integer quartets, where each number is the count of consecutive boolean pairs (00, 01, 10, 11). Also, we provide a theorem with constructive proof that shows from which types of quartets the n-dimensional boolean vector can be reconstructed. For each type of quartet, we calculate how many boolean vectors exist that are mapped to the quartet. Furthermore, we calculate the number of quartets that can be mapped from boolean vectors of length n. We provide similar theorems and calculations for cyclical n-dimensional boolean vectors, meaning last and first values are also considered as a pair. Moreover, we map n-dimensional boolean vectors to the set of non-negative integer octads, where each number is the count of consecutive boolean triples (e.g. 000, 001, 010, ...). As for quartets, we provide a constructive theorem that shows from which types of octads the n-dimensional boolean vector can be reconstructed.

Keywords: neighbourings, neighbor pairs, neighbor triplets.

Let we have a $\alpha \in B^n$ vector: α has *x 11*, y *10*, *z 01,* and *t 00* neighbourings. Obviously, $x + y + z + t = n - 1$. Let us find out whether there exists an α vector for the given quad (x, y, z, t), and if yes, find the amount of them. We will also discuss the case when the vector α is cyclic; naturally, $x + y + z + t = n$ for this case.

Theorem 1. *For the given quad* (x, y, z, t) *there exists an α vector if* $|y - z| \leq 1$ *and if* $y = z = 0$; *then* $x \times t = 0$.

Proof. First, we will show that if exists α vector, then $|y - z| \leq 1$. Let's assume equal numbers exist in consecutive places $(x + t \neq 0)$ when we remove one of them and recalculate x, y, z, t for the resulting vector y and z will stay the same, and $x + t$ will decrease by one. We will do this till $x + t$ is zero. Now resulting vector is alternating ones and zeros, e.g., 101010101. Obviously, $|y - z| \leq 1$.

$y - z = -1$, if the first number of α is one and last is zero

$y - z = 0$, if first and last numbers of α are equal

$y - z = 1$, if the first number of α is 0 and last is 1

And if $y = z = 0$ meaning there are no 01 or 10 pairs, the vector is either only ones or only zeros which is equivalent to $x \times t = 0$.

Now the second part, we should construct vector α that has *x 11*, y *10*, *z 01* and *t 00*,

If $y = z = t = 0$, then α is a vector with only ones and length x,

If $y = z = x = 0$, then α is a vector with only zeros and length t,

If $y + z \neq 0$, then we take alternating ones and zeros, that has y 10 and z 01. Then we take some 1 and add x ones, same with t zeros. This vector will satisfy described conditions.

Let us find out the amount of α vectors corresponding to the quad (x, y, z, t)

Theorem 2. *Let we have a* (x, y, z, t) *quad. Then the amount of the* α *vectors are:*

$$\begin{cases} 1 & \text{, if } y+z=0 \text{ and } x\times t=0 \\ C_{x+y-1}^{y-1}\times C_{t+y-1}^{y-1} & \text{, if } y-z=1 \\ C_{x+y-1}^{y-1}\times C_{t+y}^{y}+C_{x+y}^{y}\times C_{t+y-1}^{y-1} & \text{, if } y=z=0 \\ C_{x+y}^{y}\times C_{t+y}^{y} & \text{, if } y-z=-1 \end{cases}$$

Proof. For proof, we will use the following lemma:

Lemma: *We can represent some number n, as a sum of non-negative numbers C_{n+k-1}^{k-1} ways.*

If $y+z=0$ and $x\times t=0$, then there is obviously only one such vector.

If $y-z=1$, when $x=t=0$ then the alternating vector will be

$$\underbrace{101010\ldots101010}_{2y}$$

We want to calculate how many ways we can insert ones and zeros to have a vector corresponding to (x,y,z,t) quartet. We need to insert x ones in y places, so we need to represent, using lemma, we know there are C_{x+y-1}^{y-1}ways, same with t zeros C_{t+y-1}^{t-1}, so overall there are $C_{x+y-1}^{y-1}\times C_{t+y-1}^{y-1}$ways.

If $y-z=-1$ we can do the same calculations swapping zeros and ones and we know that $y=z-1$,

$$C_{x+z-1}^{z-1}\times C_{t+z-1}^{z-1}=C_{x+y}^{y}\times C_{t+y}^{y}$$

If $y-z=0$ then there are two possibilities for $x=t=0$

$$\underbrace{10101\ldots10101}_{2y+1} \text{ and } \underbrace{01010\ldots01010}_{2y+1}$$

For the first one, we should insert x ones in $y+1$ places. There are C_{x+y}^{y} possible ways, and t zeros in y places, C_{t+y-1}^{y-1} ways. So, for the first case, we have $C_{x+y}^{y}\times C_{t+y-1}^{y-1}$ possible cases. For the second case, we can do similar calculations, inserting x ones in y places and t zeros in $y+1$

places. Overall, we would have $C_{x+y-1}^{y-1} \times C_{t+y}^{y} + C_{x+y}^{y} \times C_{t+y-1}^{y-1}$ possible ways to construct α vector.

Theorem 3. *For the given number n, the amount of corresponding quads is* $3k^2 + 2k + 2$, *if* $n = 2k + 1$ *and* $3k^2 - k + 2$, *if* $n = 2k$.

Proof. First, when $n = 2k + 1$, here we have 3 possibilities: $y - z = 0, y - z = -1, y - z = 1$

If $y - z = 0$ then $x + 2y + z = n - 1 = 2k$ $(x, y, z \geq 0)$.

We just need to guarantee that $x + t$ is an even number in $[0, 2k]$ range, and y can be calculated.

$x + z = 0$ one possibility

$x + z = 2$ three possibilities

…

$x + z = 2(k - 1)$ $2k - 1$ possibilities

$x + z = 2k$ 2 possibilities ($x \times z = 0$ when $y = z = 0$)

Overall, $1 + 3 + 5 + \cdots + (2k - 1) + 2 = k^2 + 2$.

If $y - z = 1$ then $x + 2z + t = 2k - 1$ $(x, z, t) \geq 0)$.

Here we need to guarantee that $x + t$ is odd and in $[0, 2k]$ range.

$x + z = 1$ two possibilities

$x + z = 3$ four possibilities

…

$x + z = 2k - 1$ $2k$ possibilities

$2 + 4 + \cdots + 2k = k(k + 1)$ possibilities.

If $y - z = -1$ we will do the same calculations that we did in the previous case.

So, for $n = 2k + 1$ we have $k^2 + 2 + k(k + 1) + k(k + 1) = 3k^2 + 2k + 2$. Similarly, for $n = 2k$ we would have $3k^2 - k + 2$.

Let us formulate the theorems for the corresponding cyclic vectors. The three theorems presented below will answer these questions. Following theorems can be proved in the same way as non-cyclic counterparts.

Theorem 4. *There exists a cyclic α vector for the quad (x, y, z, t) iff $(y + z = 0$ and $x \times t = 0)$ or $y = z$.*

Theorem 5. *The amount of the cyclic vectors α corresponding to the quad (x, y, z, t) is:*

$$\begin{cases} 1 & \text{, if } y + z = 0 \text{ and } x \times t = 0 \\ C_{x+y-1}^{y-1} \times C_{t+y-1}^{y-1} & \text{, if } y = z. \end{cases}$$

Theorem 6. *For the given number n, the amount of the corresponding quads is $k^2 + 2$, if $n = 2k$ and $k^2 + k + 2$, if $n = 2k + 1$.*

Now let us consider the same problem for the triplets. We have the vector $\alpha \in B^n$. α has c_{000} *000*, c_{001} *001* and so on. It is clear that $\sum_i c_i = n - 2$. Let us find out for what c_i values there exists an α vector. We make the following denotations:

$$b_{00} \equiv c_{100} - c_{001},$$
$$b_{01} \equiv c_{001} + c_{101} - c_{011} - c_{010},$$
$$b_{10} \equiv c_{010} + c_{110} - c_{101} - c_{100},$$
$$b_{11} \equiv c_{011} - c_{110}$$

Theorem 7. *For the given octad $(c_{000}, c_{001}, c_{010}, c_{011}, c_{100}, c_{101}, c_{110}, c_{111})$ there exists an α vector iff the following conditions are satisfied:*

1. *If $c_{001} = c_{100} = 0$, then either $c_{000} = n - 2$, or $c_{000} = 0$,*
2. *If $c_{110} = c_{011} = 0$, then either $c_{111} = n - 2$, or $c_{111} = 0$,*
3. *$|b_i| \leq 1$,*
4. *only one of the numbers b_i can have the value 1,*
5. *only one of the numbers b_i can have the value -1,*

Proof. Let us show if the octad that satisfies conditions 1-5 is given, then we can construct a vector whose corresponding octad will be the given one. First, we will build a vector that will have $(c_{001}, c_{010}, c_{011}, c_{100}, c_{101}, c_{110})$ values, and then we will add 000 and 111 triplets. For constructions, we will use a directed graph (Figure 1) where each node

represents a triple and for any v, u nodes vu is an edge if the last two digits of u are equal to first two digits of v (e.g. (101, 011) is an edge). Each directed walk in the graph that goes through node j c_j times for any j corresponds to some vector α that has hexad $(c_{001}, c_{010}, c_{011}, c_{100}, c_{101}, c_{110})$. For example, $011 \to 110 \to 100 \to 001$ walk corresponds to vector 011001.

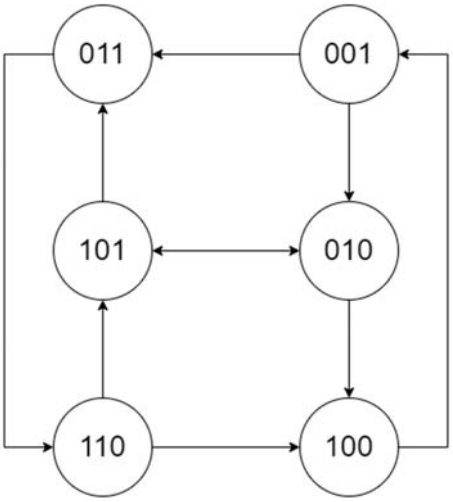


Figure 1.

Lemma 1: *If* $\forall i\ b_i = 0$, *then hexad will have this form* $(x, x - y + z, y, x, z, y)$, *where* x, y, z *are non-negative integers.*

Proof. From b_i definitions we have $c_{100} = c_{001}$, $c_{001} + c_{101} = c_{011} + c_{010}$, $c_{010} + c_{110} = c_{101} + c_{100}$, $c_{011} = c_{110}$. If we define $c_{001} \equiv x, c_{011} \equiv y$, $c_{101} \equiv z$ and use the above equations, every c_j can be calculated.

Lemma 2: *If* $\forall i\ b_i = 0$ *and some* $c_j \geq 0$ *(*j *is one of the following triplets* $011, 001, 101, 010, 110, 100$*) then we can construct a vector that will start with* j *and will correspond to the given hexad.*

Proof. As proof, we will describe a walk on the graph. Each time going out from node, we decrement c of that node by one. If all c_k are positive, we walk on cycle $011 \to 110 \to 101 \to 010 \to 100 \to 001$ until

one of c_k becomes zero, if one of the c_k is already zero, we will skip this step. Because we decremented each c by one, new b'_i values also will be zero.

If zero node is one of $011, 001, 110, 100$ nodes, using $b'_i = 0$ conditions and removing zero nodes, we will have a graph that looks like Figure 2. Now we can walk on this graph from any node resolving all c_k values to zero. If we are on v_2 we go by cycle $v_2 \to v_1 \to v_4 \to v_2$ p times, then we have v_2 with value $q - p$ and v_3 with the same value, we do $v_2 \to v_3 \to v_2$ cycle $q - p$ times. If we are on v_1 or v_4 we do half cycle and go to v_2 then the second cycle $q - p$ times, then return and complete cycle, and finally do $p - 1$ times cycle $v_2 \to v_1 \to v_4 \to v_2$ (from v_1 or v_4). Note if $p = 0$ then we couldn't be on v_1 or v_4.

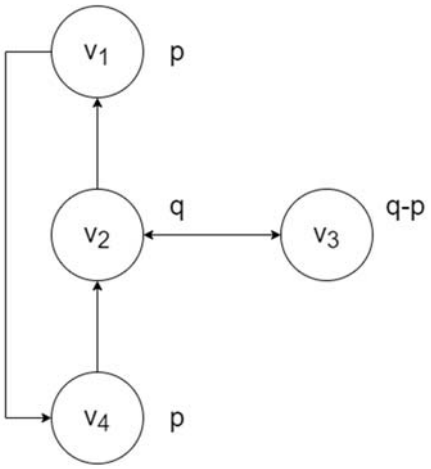


Figure 1.

If the zero node is 101 or 010, with the same reasoning, we will get to the graph in Figure 3. Here we have 2 cycles $v_1 \to v_5 \to v_4 \to v_3 \to v_2 \to v_1$ and $v_4 \to v_2 \to v_3 \to v_4$. If we are on node v_1 or node v_5 then we do half cycle to node v_4 then second cycle $q - p$ times, then finish the first cycle and do $p - 1$ times first cycle. If we are on node v_4 or node v_2 then we do the first cycle p times, then the second cycle $q - p$ times. And finally,

if we are on node v_3 we go to v_4 then to v_2 then do the first cycle p times, which we will finish on v_2 then we go to v_3 and do the second cycle $q - p - 1$ times.

Lemma 3: *If* $\forall i\ b_i = 0$ *and some* $c_j \geq 0$, *then we can construct a vector that will end with* j *and correspond to the given hexad.*

Proof. Let us "mirror" our c values:

$$c'_{001} = c_{100} \quad c'_{010} = c_{010}$$
$$c'_{011} = c_{110} \quad c'_{100} = c_{001}$$
$$c'_{101} = c_{101} \quad c'_{110} = c_{011}$$

And notice that the graph in figure 1 will be the same (isomorph). Using the previous lemma, we can construct a vector, which will start with j, and reverse it.

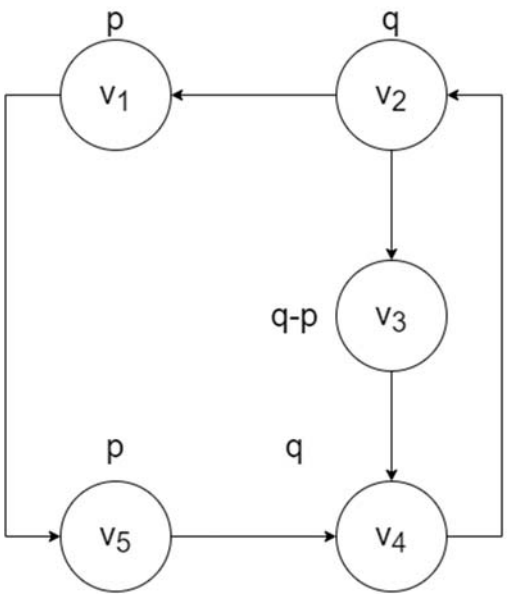


Figure 2.

Lemma 4: *If* $\forall i\ b_i = 0$ *and at least one* c_j *is not zero, then we can construct a vector that starts with* 10 *and we can construct a vector that starts with* 01.

Proof. From **Lemma 1** we know that hexad is $(x, x-y+z, y, x, z, y)$. If c_{101} or c_{100} is not zero, then using **Lemma 2** we will construct a vector that will start with 101 or 100. Let's assume $c_{101} = 0$ and $c_{100} = 0$, meaning $z = 0$ and $x = 0$. The hexad will take the following form $(0, -y, y, 0, 0, 0)$, c_{010} can't be less than zero, which means y also should be zero, which contradicts the given condition. The same reasoning can be applied to the second case.

Lemma 5: *If* $\forall i\ b_i = 0$ *and at least one* c_j *is not zero, then we can construct a vector that ends with* 10, *and we can construct a vector that ends with* 01.

Proof. Let us again "mirror" our c values:

$$c'_{001} = c_{100} \quad c'_{010} = c_{010}$$
$$c'_{011} = c_{110} \quad c'_{100} = c_{001}$$
$$c'_{101} = c_{101} \quad c'_{110} = c_{011}$$

And calculate b values for this new hexad

$$b'_{00} = c'_{100} - c'_{001} = -b_{00} = 0$$
$$b'_{01} = c'_{001} + c'_{101} - c'_{011} - c'_{010} = -b_{10} = 0$$
$$b'_{10} = c'_{010} + c'_{110} - c'_{101} - c'_{100} = -b_{10} = 0$$
$$b'_{11} = c'_{011} - c'_{110} = -b_{11} = 0$$

Conditions for the previous lemma are satisfied, so we can construct a vector that starts with 10 (01) and mirror it (e.g., 10111 → 11101). This new vector will end with 01 (10) and correspond to the given hexad.

Lemma 6: *If* $b_i = 1, b_j = -1$ *and we constructed vector, then we can construct vector for when* $b_{\bar{\imath}} = 1, b_{\bar{\jmath}} = -1$*, where* $\bar{x}$ *is* complement x *(e.g.* $\overline{01} = 10$*).*

Proof. The proof is very similar to the previous, but instead of mirroring, we use complement.

$$b'_{00} = c'_{100} - c'_{001} = b_{00}$$
$$b'_{01} = c'_{001} + c'_{101} - c'_{011} - c'_{010} = b_{10}$$
$$b'_{10} = c'_{010} + c'_{110} - c'_{101} - c'_{100} = b_{10}$$
$$b'_{11} = c'_{011} - c'_{110} = b_{11}$$

We construct for these new values and then complement the result.

Now let us prove the main theorem. From conditions 3-5 and knowing from the definition that the sum of all b_j is zero, we can conclude that there are $A_4^2 + 1 = 13$ possible cases.

1. $b_{00} = 1, b_{01} = -1$.
   $b_{00} = c_{100} - c_{001} = 1$ therefore $c_{100} > 0$. And $b_{01} = c_{001} + c_{101} - c_{011} - c_{010} = -1$ meaning either $c_{011} > 0$ or $c_{010} > 0$. If $c_{010} > 0$ we construct vector using **lemma 5** that ends with 01 for hexad $(c_{001}, c_{010} - 1, c_{011}, c_{100} - 1, c_{101}, c_{110})$, obviously recalculated b_j vectors are zero. And then add to the end of the vector 00, doing so increments c_{100} and c_{010} by one. If $c_{011} > 0$, from $b_{11} = 0$ we know that $c_{011} = c_{110} > 0$, and $(c_{001}, c_{010}, c_{011} - 1, c_{100} - 1, c_{101}, c_{110} - 1)$, again new b_j are zero, and we can use **lemma 5** and construct vector ending with 01, and then add 100 to the end of the vector.
2. $b_{11} = 1, b_{10} = -1$. It can be proven easily using previous case and **lemma 6**.
3. $b_{00} = -1, b_{01} = 1$.
   Then from definitions of b_j we know that $c_{001} > 0$. Using **lemma 4** for hexad $(c_{001} - 1,\ c_{010}, c_{011}, c_{100}, c_{101}, c_{110})$ starting from 01, we add 0 to the start of the vector.

4. $b_{11} = -1, b_{10} = 1$. **Lemma 6** and case 3.
5. $b_{00} = 1, b_{10} = -1$.
   Then from definitions of b_j we know that $c_{100} > 0$. Using **lemma 5** for hexad $(c_{001}, c_{010}, c_{011}, c_{100} - 1, c_{101}, c_{110})$ ending with 10, we add 0 to the end of the vector.
6. $b_{11} = 1, b_{10} = -1$. **Lemma 6** and case 5.
7. $b_{00} = -1, b_{10} = 1$.
   $b_{00} = c_{100} - c_{001} = -1$ therefore $c_{001} > 0$. And $b_{10} = c_{010} + c_{110} - c_{101} - c_{100} = 1$ meaning either $c_{010} > 0$ or $c_{110} > 0$. If $c_{010} > 0$ we construct vector using **lemma 4** that starts with 01 for hexad $(c_{001} - 1, c_{010} - 1, c_{011}, c_{100}, c_{101}, c_{110})$. And add to the start of the vector 00. If $c_{110} > 0$, from $b_{11} = 0$ we know that $c_{011} = c_{110} > 0$, and $(c_{001} - 1, c_{010}, c_{011} - 1, c_{100}, c_{101}, c_{110} - 1)$, again new b_j are zero, and we can use **lemma 4** and construct vector starting with 10, and then add 001 to the start of the vector.
8. $b_{11} = -1, b_{01} = 1$. **Lemma 6** and case 7.
9. $b_{01} = 1, b_{10} = -1$.
$$b_{01} = c_{001} + c_{101} - c_{011} - c_{010} = 1$$
$$b_{10} = c_{010} + c_{110} - c_{101} - c_{100} = -1$$
   From b_{01} we see that either $c_{001} > 0$ or $c_{101} > 0$. If $c_{101} > 0$ we construct vector using **lemma 4** that starts with 01 for hexad $(c_{001}, c_{010}, c_{011}, c_{100}, c_{101} - 1, c_{110})$ and add 1 to the start. If $c_{001} > 0$, from $b_{00} = 0$ we know that $c_{100} = c_{100} > 0$, and can do same for hexad $(c_{001} - 1, c_{010}, c_{011}, c_{100} - 1, c_{101}, c_{110})$ and add 10 to the start.
10. $b_{10} = 1, b_{01} = -1$. **Lemma 6** and case 9.
11. $b_{00} = 1, b_{11} = -1$.
    From definitions, we know that $c_{100} > 0$ and $c_{110} > 0$. If $c_{001} > 0$ we will start from 001 using **lemma 2** for hexad $(c_{001}, c_{010}, c_{011}, c_{100} - 1, c_{101}, c_{110} - 1)$, and then add 11 to the start. If $c_{011} > 0$ we will end on 100 using **lemma 3** for the same hexad and add 00 to the end. If $c_{001} = c_{011} = 0$, then from b_j

definitions our hexad looks like this $(0, x, 0, 1, x, 1)$, then our vector is 11 01 01 ... 01 00, where the number of 01 pairs is x, for example for $x = 2$, the vector is 11010100.

12. $b_{11} = 1, b_{00} = -1$. **Lemma 6** and case 11.
13. $b_{00} = b_{01} = b_{10} = b_{11} = 0$ is proven in **lemma 2**.

Now we will add into consideration c_{000} and c_{111}. For that, we first construct vector without them then find 00 or 11 in the constructed vector and add c_{000} times new 0 and c_{111} times new 1. For example, if $c_{000} = 2, c_{111} = 1$ and the vector is 100110, then our new vector will be 100001110. But what if in our vector there is not a 00 pair, then form condition 1 we know that either $c_{000} = 0$ or everything else is zero ($c_{000} = n - 2$), same for c_{111}.

Now let us prove that for any vector with octad ($c_{000}, c_{001}, c_{010}, c_{011}, c_{100}, c_{101}, c_{110}, c_{111}$), conditions $1 - 5$ are true.

Conditions 1 and 2 are obviously true, if there is not a 100 or 001, then either vector is zeros, or there is not a 000 triple, same for 111.

Let us show that $|b_{01}| \leq 1$, $b_{01} = c_{001} + c_{101} - c_{011} - c_{010}$, a left neighbour of triplets 011 and 010 can only be 001 and 101, and a right neighbour of triplets 001 and 101 can only be 011 and 010. Meaning $c_{001} + c_{101} = c_{011} + c_{010}$, except when the vector starts with 01, then $b_{01} = -1$, or when vector ends with 01, then $b_{01} = 1$. The same conclusions can be applied to b_{10}, also to b_{00} and b_{11}, if we write them as $b_{00} = c_{000} + c_{100} - c_{000} - c_{001}$ and $b_{11} = c_{011} + c_{111} - c_{110} - c_{111}$. From all b_j values only one can be -1, and only one can be 1, because only one triplet is at the start of the vector, and only one can be in the end.

REFERENCES

1. *Hamming R.* Error detecting and error correcting codes – The Bell System Technical Journal. 1950. V. 29. № 2. PP. 149–154.
2. *Golay M. J. E.* Notes on digital coding. IEEE Information Society Newsletter. 1949. V. 37. № 6. P. 657.

3. *Leontiev V., Movsesyan G., Margaryan Zh.* Codes in Additive Channels. Dokladi of AN of Armenia. 2010. V.110. № 4. PP. 334–339.
4. *Leontiev V., Movsesyan G., Margaryan Zh.* Constant Weight Perfect and D-Representable Codes. Proceedings of the YSU A: Physical and Mathematical Sciences. 2012. V. 46. № 1. PP. 16–19.
5. *Leontiev V., Movsesyan G., Margaryan Zh.* Code Volume Boundaries in The Additive Channel. Proceedings of the YSU A: Physical and Mathematical Sciences. 2012. V. 46. № 2. PP. 14–21.

ЗАДАЧА О КОЛИЧЕСТВЕ ОДИНАКОВЫХ ПОСЛЕДОВАТЕЛЬНЫХ ПАР В БУЛЕВЫХ ВЕКТОРАХ

Н.А. Ханданян, Ж.Г. Маргарян, О.К. Саакян

АННОТАЦИЯ

В данной статье отображается набор n-мерных булевых векторов на наборы неотрицательных целочисленных квартетов, где каждое число представляет собой количество последовательных булевых пар (00, 01, 10, 11). Кроме того, приводим теорему с конструктивным доказательством, показывающую, из каких типов квартетов можно восстановить n-мерный булевой вектор. Для каждого типа квартета мы вычисляем, сколько существует булевых векторов, отображаемых в квартет. Кроме того, мы вычисляем количество квартет, которые можно отобразить из булевых векторов длины n. Мы приводим аналогичные теоремы и вычисления для циклических n-мерных булевых векторов, где цикличный означает, что последнее и первое значения также рассматриваются как пара. Кроме того, мы отображаем n-мерные логические векторы в набор неотрицательных целых октадов, где каждое число представляет собой количество последовательных булевых троек (например, 000, 001, 010, ...). Как и для квартетов, мы приводим конструктивную теорему, показывающую, из каких типов октадов можно восстановить n-мерный логический вектор.

Ключевые слова: пары, соседние пары, соседние тройки.

DOI 10.48200/1829-0450_pmn_2022_2_32
УДК 519.174.7

Поступила: 22.10.2022г.
Сдана на рецензию: 29.10.2022г.
Подписана к печати: 27.11.2022г.

ON ANTIMAGIC EDGE COLORINGS OF CERTAIN GRAPHS

H. Mikaelyan

Yerevan State University

hamletm2000@gmail.com

ABSTRACT

For a given graph G and a proper edge t-coloring α defined on G, denote by $Sum_G(v,\alpha)$ the sum of colors of edges neighboring v, where $v \in V(G)$. In that case, α is called an antimagic edge t-coloring of graph G, if for every pair of distinct vertices $v_1, v_2 \in V(G)$, $Sum_G(v_1,\alpha) \neq Sum_G(v_2,\alpha)$. The set of graphs G, for which there exists some t, such that G admits an antimagic edge t-coloring, is denoted by $\mathcal{AM}$. For any graph $G \in \mathcal{AM}$, denote by $\omega_{am}(G)$ the least positive integer t, for which G admits an antimagic edge t-coloring, and by $\Omega_{am}(G)$ the biggest integer t, for which G admits an antimagic edge t-coloring. In this paper we obtain some estimations and some exact results on $\omega_{am}(G)$ and $\Omega_{am}(G)$ for wheels, some Halin graphs, and complete graphs.

Keywords: antimagic edge-coloring, edge-coloring.

Introduction

Throughout the paper all graphs are finite, undirected, simple and connected. Let $V(G)$ and $E(G)$ denote the sets of vertices and edges of G,

respectively. For $v \in V(G)$, we define $N_G(v)$ as the set of neighbors of v: $N_G(v) = \{u \mid uv \in E(G)\}$. The number of vertices in this set is called the degree of the vertex v and is denoted by $d_G(v)$. All the terms and concepts that are not defined in this paper can be found in [11, 12].

A proper edge t-coloring of a graph G is a surjective mapping $\alpha : E(G) \to \{1, 2, \dots, t\}$ such that $\alpha(e) \neq \alpha(e')$ for any pair of adjacent edges $e, e' \in E(G)$. Consider a proper edge coloring α. We define the spectrum of a vertex v in the graph G with respect to the coloring α as the set of colors of edges adjacent to v and denote it by $S_G(v, \alpha)$ ($S_G(v, \alpha) = \{\alpha(uv) \mid u \in N_G(v)\}$). Similarly, we denote by $Sum_G(v, \alpha)$ the sum of colors of edges adjacent to v ($Sum_G(v, \alpha) = \sum_{u \in N_G(v)} \alpha(uv)$). In some cases, we will use the notations $S(v, \alpha)$ $\big(S(v)\big)$ and $Sum(v, \alpha)$ $(Sum(v))$ instead of $S_G(v, \alpha)$ and $Sum_G(v, \alpha)$ respectively if the graph (along with the coloring) is obvious from the context.

A proper edge t-coloring α of a graph G, for which all the values $Sum_G(v, \alpha), v \in V(G)$ are distinct, is called an antimagic edge t-coloring. If graph G admits an antimagic edge $|E(G)|$-coloring, then G is called an antimagic graph.

In 1990, Hartsfield and Ringel conjectured the following:

Conjecture [8]. All simple connected graphs except for K_2 are antimagic.

Interestingly, the conjecture remains open even for large classes of graphs such as trees and bipartite graphs. However, the conjecture has undergone numerous researches. Here are some significant results: trees which don't have any vertex of degree 2 are antimagic [9]; connected k-regular graphs ($k \geq 2$) and their Cartesian products are antimagic [2, 3, 4, 6]; simple cycles, paths with length greater than 1 are antimagic [5, 8, 10]. There are also some results on graphs depending on their minimum degree, maximum degree or average degree [1, 7].

The conjecture above has been formulated by the authors in the terms of edge labelings (assignments of edges to numbers $1, 2, \dots, |E(G)|$). This

paper offers to consider more generalized problem taking into account that edge labeling is a particular case of edge coloring.

Thus, we introduce a class of graphs $\mathcal{AM}$ which contains graphs G for which there exists some integer t, such that G admits antimagic edge t-coloring. Then, for any graph $G \in \mathcal{AM}$, we denote by $\omega_{am}(G)$ the least positive integer t, for which G admits an antimagic edge t-coloring, and by $\Omega_{am}(G)$ the largest integer t, for which G admits antimagic an edge t-coloring. Obviously, if the conjecture is true, then $\mathcal{AM}$ consists of all connected graphs except for K_2, and for each of those graphs, $\Omega_{am}(G) = |E(G)|$.

Let us consider these particular types of graphs: complete graphs, wheels, and Halin graphs. Let n be an integer number greater than or equal to 4. We call $W_n = K_1 + C_{n-1}$ a wheel of n vertices. A graph $H = T \cup C$, where T is a tree without vertex of degree 2 and C is a simple cycle consisting of the leaves of T, is called a Halin graph. If T is a double star, and each star has n leaves ($n \geq 2$), then we will denote the Halin graph by H_n. In the figure 1, you can see examples of H_n graphs for $n = 3$ (on the left) and $n = 5$ (on the right).

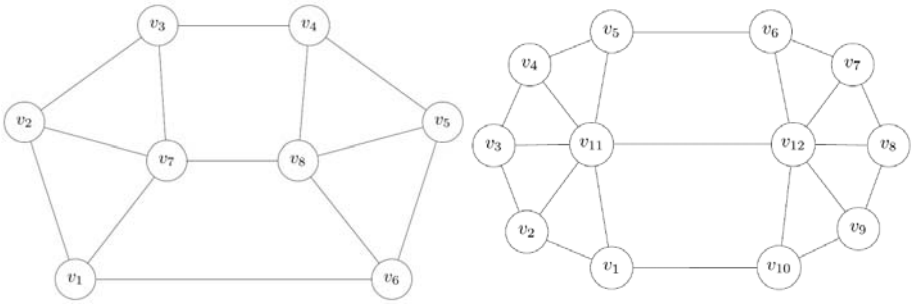


Figure 1.

The fact that complete graphs and wheels are antimagic graphs, are presented as exercises in [8], and can be derived also from theorems in [1], [2], [3]. The following theorems immediately imply from the latter:

Theorem 1: For any integer $n \geq 3$, $K_n \in \mathcal{AM}$ and $\Omega_{am}(K_n) = |E(K_n)| = \frac{n(n-1)}{2}$.

Theorem 2: For any integer $n \geq 4$, $W_n \in \mathcal{AM}$ and $\Omega_{am}(W_n) = |E(W_n)| = 2n - 2$.

In this paper we consider $\omega_{am}(G)$ parameter for types of graphs mentioned above, $\Omega_{am}(G)$ parameter for H_n graph, and obtain some results on their exact values, or lower and upper bounds.

Main results

Lemma: Let G is a graph in $\mathcal{AM}$. Let for some d $(1 \leq d \leq \Delta(G))$, n_d is the number of vertices with degree d in G $(n_d > 0)$. In that case, $\omega_{am}(G) \geq \left\lceil \frac{n_d - 1}{d} \right\rceil + d$.

Proof: Let $\omega_{am}(G) = t$. Consider any antimagic t-coloring of the graph G. Consider $Sum(v)$ values for all such vertices $v \in V(G)$ for which $d_G(v) = d$. Since $S(v) \subseteq \{1, 2, \dots, t\}$ and $|S(v)| = d$, $1 + 2 + \cdots + d = \frac{d(d+1)}{2} \leq Sum(v) \leq (t - d + 1) + (t - d + 2) + \cdots + (t - d + d) = (t - d) \cdot d + \frac{d(d+1)}{2}$. It means that the set of possible values for $Sum(v)$ lies in $\left[\frac{d(d+1)}{2}, (t - d) \cdot d + \frac{d(d+1)}{2}\right]$, so the number of distinct values cannot exceed $(t - d)d + \frac{d(d+1)}{2} - \frac{d(d+1)}{2} + 1 = (t - d)d + 1$. It implies that $n_d \leq (t - d)d + 1 \Rightarrow t \geq \left\lceil \frac{n_d - 1}{d} \right\rceil + d$.

We use the lemma above as the main lower bound for $\omega_{am}(G)$ parameter in the following theorems. For the upper bound of $\Omega_{am}(G)$ we use the obvious fact $\Omega_{am}(G) \leq |E(G)|$, which implies from the surjectivity of the proper $\Omega_{am}(G)$-coloring. For the upper bound of $\omega_{am}(G)$ as well as for the lower bound of $\Omega_{am}(G)$ parameter we prove our results by constructing respective examples.

Theorem 3: For any integer $n \geq 4$, $W_n \in \mathcal{AM}$ and

$$\omega_{am}(W_n) = \begin{cases} 5, & if\ n = 4 \\ n-1, & if\ n \geq 5 \end{cases}.$$

Proof: Firstly, let us prove that

$$\omega_{am}(W_n) \geq \begin{cases} 5, & if\ n = 4 \\ n-1, & if\ n \geq 5 \end{cases}.$$

In order to do that, we use the lemma considering $d = n - 1$.

If $n = 4$, all the vertices of the wheel have a degree of $d = n - 1 = 3$, so $n_d = 4$. Hence, $\omega_{am}(W_4) \geq \left\lceil \frac{4-1}{3} \right\rceil + 3 = 4$. Note that it is also impossible to color with 4 colors keeping all $Sum(v), v \in V(G)$ values distinct. Suppose that there is such coloring and let us consider spectrums of all vertices of the wheel. The number of subsets with cardinality $d = 3$ of the set $\{1, 2, 3, 4\}$ is exactly $n_d = 4$. So the set of spectrums of all vertices should be $\{S(v) \mid v \in V(G)\} = \{\{1, 2, 3\}, \{1, 2, 4\}, \{1, 3, 4\}, \{2, 3, 4\}\}$. But in this set, each number appears 3 times, whereas for any proper edge coloring each color c must persist in that set twice the number of edges colored with c times. It implies that an antimagic coloring with 4 colors doesn't exist, so $\omega_{am}(W_4) \geq 5$.

If $n \geq 5$, one of the vertices has a degree of $n - 1$, and others have a degree of 3 ($3 < n - 1$), so $n_d = 1 \Rightarrow \omega_{am}(W_n) \geq \left\lceil \frac{1-1}{n-1} \right\rceil + n - 1 = n - 1$.

So the lower bound of $\omega_{am}(W_n)$ is proven and in order to complete the proof we need to bring examples with corresponding $\omega_{am}(W_n)$ values.

Let us denote the vertices of W_n with $v_1, v_2, \dots, v_n$ and enumerate them in such order that $E(W_n) = \{v_1v_i \mid i = 2, \dots, n\} \cup \{v_iv_{i+1} \mid i = 2, \dots, n-1\} \cup \{v_2v_n\}$.

For $n = 4$, we construct the coloring α in the following way: Say that $\alpha(v_1v_2) = 1, \alpha(v_1v_3) = 2, \alpha(v_1v_4) = 5, \alpha(v_2v_3) = 3, \alpha(v_3v_4) = 4, \alpha(v_4v_2) = 2$ (figure 2). It is easy to see that the coloring is an antimagic 5-coloring.

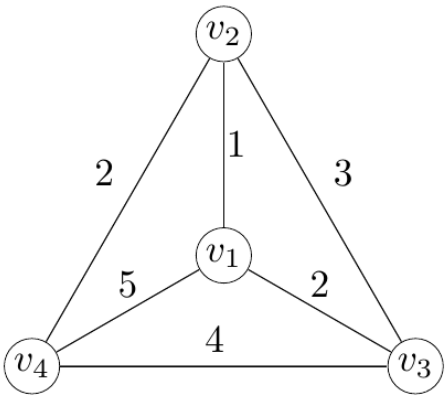


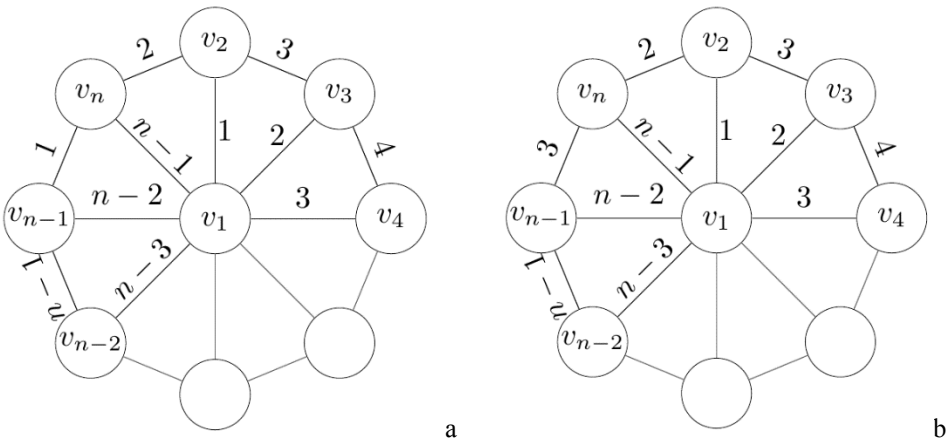
Figure 2.

For $n \geq 5$, to construct the examples of colorings we consider two cases: $n \not\equiv 1\ (mod\ 3)$ and $n \equiv 1\ (mod\ 3)$.

1. $n \not\equiv 1\ (mod\ 3)$

In this case we construct the required coloring α in the following way (figure 3a):

$$\alpha(v_1v_i) = i - 1\ (i = 2, 3, \dots, n),$$
$$\alpha(v_iv_{i+1}) = i + 1\ (i = 2, 3, \dots, n-2),$$
$$\alpha(v_{n-1}v_n) = 1, \qquad \alpha(v_nv_2) = 2.$$



a b

Figure 3.

Since $\alpha(v_1v_i) = i - 1$ $(i = 2, 3, \dots, n)$, all the colors from 1 to $n - 1$ appear in the coloring, so the mapping is surjective. It is also easy to see that the adjacent edges have different colors, so α is a proper edge $n - 1$-coloring. The spectrums of the vertices will look like this:

$$S_{W_n}(v_1, \alpha) = \{1, 2, \dots, n-1\},$$
$$S_{W_n}(v_i, \alpha) = \{i-1, i, i+1\}\ (i = 2, 3, \dots, n-2),$$
$$S_{W_n}(v_{n-1}, \alpha) = \{n-2, n-1, 1\}, \qquad S_{W_n}(v_n, \alpha) = \{n-1, 1, 2\}.$$

To prove that α is an antimagic $n - 1$-coloring, it remains now to calculate the $Sum(v_i)$ for each $i \in \{1, 2, \dots, n\}$ and make sure that they are pairwise distinct.

$$Sum_{W_n}(v_1, \alpha) = \frac{n(n-1)}{2},$$
$$Sum_{W_n}(v_i, \alpha) = 3i\ (i = 2, 3, \dots, n-2),$$
$$Sum_{W_n}(v_{n-1}, \alpha) = 2n - 2, \qquad Sum_{W_n}(v_n, \alpha) = n + 2.$$

The following expressions show that $Sum_{W_n}(v_1, \alpha)$ is different from the others:

$$n \geq 5 => Sum_{W_n}(v_1, \alpha) = \frac{n(n-1)}{2} \geq 2n > 2n - 2 = Sum_{W_n}(v_{n-1}, \alpha),$$
$$n \geq 5 => Sum_{W_n}(v_1, \alpha) = \frac{n(n-1)}{2} \geq 2n > n + 2 = Sum_{W_n}(v_n, \alpha),$$
$$n = 5 => Sum_{W_n}(v_1, \alpha) = 10 \neq 3i\ (i = 2, \dots, n-2),$$
$$n = 6 => Sum_{W_n}(v_1, \alpha) = 15 \neq 3i\ (i = 2, \dots, n-2),$$
$$n \geq 7 => Sum_{W_n}(v_1, \alpha) = \frac{n(n-1)}{2} \geq 3n > 3i\ (i = 2, \dots, n-2).$$

It is obvious that $Sum_{W_n}(v_i, \alpha) = 3i$ $(i = 2, 3, \dots, n-2)$ values are pairwise distinct and also differ from $Sum_{W_n}(v_{n-1}, \alpha) = 2n - 2$ and $Sum_{W_n}(v_n, \alpha) = n + 2$ since $n \not\equiv 1\ (mod\ 3)$. Finally, $2n - 2 > n + 2$, so

all $Sum_{W_n}(v_i, \alpha)$ $(i = 1, \dots, n)$ are pairwise distinct and the coloring is antimagic.

2. $n \equiv 1 \ (mod\ 3)$

In this case we construct the coloring β in the following way (figure 3b):

$$\beta(v_1v_i) = i - 1 \ (i = 2, 3, \dots, n),$$
$$\beta(v_iv_{i+1}) = i + 1 \ (i = 2, 3, \dots, n-2),$$
$$\beta(v_{n-1}v_n) = 3, \qquad \beta(v_nv_2) = 2.$$

Obviously, the coloring is a proper edge $n-1$-coloring.

$$Sum_{W_n}(v_1, \beta) = \frac{n(n-1)}{2},$$
$$Sum_{W_n}(v_i, \beta) = 3i \ (i = 2, 3, \dots, n-2),$$
$$Sum_{W_n}(v_{n-1}, \beta) = 2n, \qquad Sum_{W_n}(v_n, \beta) = n + 4.$$

Similarly, taking into account the $n \geq 7$, it is easy to see that the coloring is antimagic.

Theorem 4: For any integer $n \geq 2$, $H_n \in \mathcal{AM}$ and

$$\omega_{am}(H_n) = \begin{cases} 5, & if\ n = 2 \\ n + 2, & if\ n \geq 3 \end{cases}.$$

Proof: Let $V(H_n) = \{v_1, v_2, \dots, v_{2n+2}\}$ and $E(H_n) = \{v_iv_{i+1} \mid i = 1,2, \dots, 2n-1\} \cup \{v_{2n+1}v_i \mid i = 1,2, \dots, n\} \cup \{v_{2n+2}v_i \mid i = n+1, n+2, \dots, 2n\} \cup \{v_{2n}v_1, v_{2n+1}v_{2n+2}\}$.

Firstly, we prove that

$$\omega_{am}(H_n) \geq \begin{cases} 5, & if\ n = 2 \\ n + 2, & if\ n \geq 3 \end{cases}.$$

We use the lemma with $d = n + 1$. For $n = 2$, $n_d = 6$ since each vertex will have a degree of 3, which implies $\omega_{am}(H_2) \geq \left\lceil \frac{n_d - 1}{d} \right\rceil + d =$

$\left\lceil\frac{6-1}{3}\right\rceil + 3 = 2 + 3 = 5$. For $n \geq 3$, $d_{H_n}(v_i) = 3$ $(i = 1, 2, \dots, 2n)$, $d_{H_n}(v_{2n+1}) = d_{H_n}(v_{2n+2}) = n + 1 = d \Rightarrow n_d = 2 \Rightarrow$ $\omega_{am}(H_n) \geq \left\lceil\frac{2-1}{n+1}\right\rceil + n + 1 = n + 2$.

Now we show examples in which the lower bound is reachable. For $n = 2$ and $n = 3$ examples are shown in the figure 5.

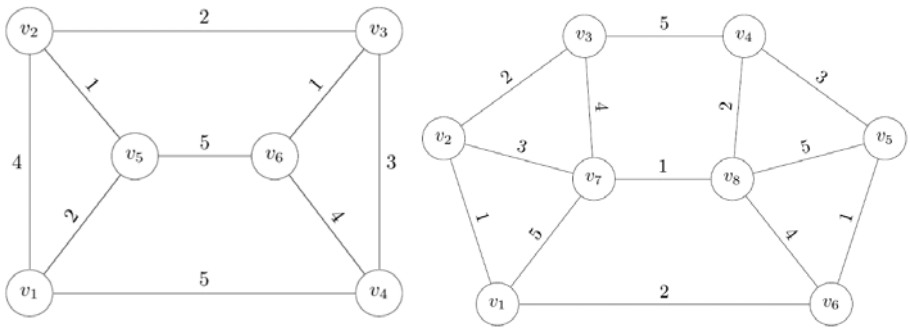


Figure 4.

For $n \geq 4$, we consider three separate cases and construct the coloring α depending on the remainder of n by 3.

1. $n \equiv 1\ (mod\ 3)$

$$\alpha(v_i v_{i+1}) = i\ (i = 1,2, \dots, n-1),$$
$$\alpha(v_i v_{i+1}) = 2n + 2 - i\ (i = n+1, n+2, \dots, 2n-1),$$
$$\alpha(v_{2n+1} v_i) = i + 1\ (i = 1,2, \dots, n-1),$$
$$\alpha(v_{2n+2} v_i) = 2n - i\ (i = n+1, n+2, \dots, 2n-1),$$
$$\alpha(v_n v_{n+1}) = n + 2, \alpha(v_{2n} v_1) = n, \alpha(v_{2n+1} v_n) = 1, \alpha(v_{2n+2} v_{2n}) = n + 2,$$
$$\alpha(v_{2n+1} v_{2n+2}) = n + 1.$$

2. $n \equiv 2\ (mod\ 3)$

$$\alpha(v_i v_{i+1}) = i\ (i = 1,2, \dots, n),$$
$$\alpha(v_i v_{i+1}) = 2n + 2 - i\ (i = n+1, n+2, \dots, 2n-1),$$
$$\alpha(v_{2n+1} v_i) = i + 1\ (i = 1,2, \dots, n-1),$$
$$\alpha(v_{2n+2} v_i) = 2n - i\ (i = n+1, n+2, \dots, 2n-1),$$

$$\alpha(v_{2n}v_1) = n-1, \alpha(v_{2n+1}v_n) = 1, \alpha(v_{2n+2}v_{2n}) = n+1, \alpha(v_{2n+1}v_{2n+2}) = n+2.$$

3. $n \equiv 0 \ (mod\ 3)$

$$\alpha(v_iv_{i+1}) = i \ (i = 1,2,\dots,n),$$
$$\alpha(v_iv_{i+1}) = 2n+2-i \ (i = n+1, n+2, \dots, 2n-1),$$
$$\alpha(v_{2n+1}v_i) = i+1 \ (i = 1,2,\dots,n-1),$$
$$\alpha(v_{2n+2}v_i) = 2n-i \ (i = n+1, n+2, \dots, 2n-1),$$
$$\alpha(v_{2n}v_1) = n-2, \alpha(v_{2n+1}v_n) = n+2, \alpha(v_{2n+2}v_{2n}) = n, \alpha(v_{2n+1}v_{2n+2}) = n+1.$$

Theorem 5: For any integer $n \geq 2$, $\Omega_{am}(H_n) = |E(H_n)| = 4n+1$.

Let $V(H_n) = \{v_1, v_2, \dots, v_{2n+2}\}$ and $E(H_n) = \{v_iv_{i+1} \mid i = 1,2,\dots,2n-1\} \cup \{v_{2n+1}v_i \mid i = 1,2,\dots,n\} \cup \{v_{2n+2}v_i \mid i = n+1, n+2, \dots, 2n\} \cup \{v_{2n}v_1, v_{2n+1}v_{2n+2}\}$. It is sufficient to construct an antimagic $|E(H_n)| = 4n+1$-coloring of H_n (again we denote it by α):

$$\alpha(v_iv_{i+1}) = i, (i = 1,2,\dots,2n-1), \alpha(v_{2n}v_1) = 2n, \alpha(v_{2n+1}v_{2n+2}) = 4n+1,$$
$$\alpha(v_{2n+1}v_i) = 4n+1-i \ (i = 1,2,\dots,n),$$
$$\alpha(v_{2n+2}v_i) = 4n+1-i \ (i = n+1, n+2, \dots, 2n).$$

Obviously, the coloring corresponds to the required conditions.

In the figure 6, you can see the illustration of the coloring for H_5.

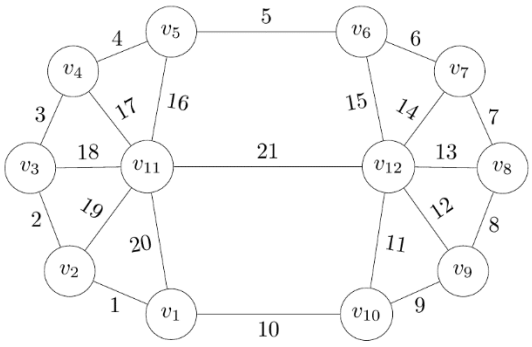


Figure 5.

The two theorems above show that the graph H_n $(n \geq 2)$ is an antimagic graph and we have found the exact values for the parameters $\omega_{am}(G)$ and $\Omega_{am}(G)$ of this type of graph.

Theorem 6: If $n \geq 3$ is an odd number, then $\omega_{am}(K_n) = n$.

Proof: Using the lemma $(d = n-1,\ n_d = n)$, $\omega_{am}(K_n) \geq \left\lceil \frac{n-1}{n-1} \right\rceil + n - 1 = n$. Let $V(G) = \{v_1, v_2, \dots, v_n\}$ and let us construct the mapping α as follows: for each v_i, v_j $(i \neq j)$, $\alpha(v_i v_j) = \big((i+j)\ mod\ n\big) + 1$, where $a\ mod\ b$ denotes the remainder of division of a by b.

Obviously, the mapping is subjective. Suppose that there are adjacent edges with the same color. It means, that there are $1 \leq i, j, k \leq n$ pairwise distinct indices for which $\alpha(v_i v_j) = \alpha(v_i v_k)$. In that case, $\big((i+j)\ mod\ n\big) + 1 = \big((i+k)\ mod\ n\big) + 1 => (i+j)\ mod\ n = (i+k)\ mod\ n => \big((i+j) - (i+k)\big) \vdots n => (j-k) \vdots n$, but since $1 \leq j \neq k \leq n$, it leads to a contradiction. So α is a proper n-coloring. For the spectrums of vertices we have $S(v_i) = \{1, 2, \dots, n\} \setminus \{2i\ mod\ n + 1\}$ $(i = 1, 2, \dots, n)$, so $Sum(v_i) = \frac{n(n+1)}{2} - ((2i\ mod\ n) + 1)$, and since n is odd, $(2i\ mod\ n) \neq (2j\ mod\ n)$ if $i \neq j$. □

Theorem 7: If $n \geq 4$ is an even number, and if $\nexists\ k \in \mathbb{Z}$, such that $n = 12k + 2$, then $\omega_{am}(K_n) = n + 1$.

Proof: Consider a graph K_n where n satisfies the conditions of the theorem $(V(K_n) = \{v_1, v_2, \dots, v_n\})$. We prove that it is impossible to construct an antimagic n-coloring of graph K_n. Since there are n vertices, where spectrum of each vertex should be unique, the set of spectrums should be the following: $\{S(v_i) \mid i = 1,2, \dots, n\} = \{\{1,2, \dots, n\} \setminus \{i\} \mid i = 1,2, \dots, n\}$. Here, each number appears in $n-1$ sets, which leads to a contradiction since each color c must persist in that set twice the number of edges colored with c times. Hence, $\omega_{am}(K_n) \geq n + 1$. We divide the rest part of the solution into two cases:

1. $\nexists\ k \in \mathbb{Z},\ n = 12k + 8$.

In this case we define the coloring β in the following way: for each v_i and v_j $(i \neq j)$, $\beta(v_i v_j) = \big((i+j)\ mod\ (n+1)\big) + 1$. It is easy to see that β is a proper $n+1$-coloring and that $Sum(v_i) = \frac{(n+1)(n+2)}{2} - (i+1) - ((2i\ mod\ (n+1)) + 1)$ $(i = 1,2,\dots,n)$. Those values are pairwise distinct iff $i + (2i\ mod\ (n+1))$ are so $(i = 1,2,\dots,n)$. When $1 \leq i \leq \frac{n}{2}$, the latter equals $3i$, when $\frac{n}{2} < i \leq n$, it equals $3i - n - 1$. Since either $n = 6k$ or $n = 6k+4$ for some $k \in \mathbb{Z}$, $3i - n - 1$ is not divisible by 3, so the values $Sum(v_i)$ are unique, which implies that the coloring is an antimagic $n+1$-coloring.

2. $\exists\, k \in \mathbb{Z},\ n = 12k + 8$.

Consider the following mapping:

$$\beta(v_i v_j) = \begin{cases} 1, & if\ i+j = 3, \\ 1, & if\ i+j = n+3, \min(i,j)\ is\ even, \\ 2, & if\ i+j = n+3, \min(i,j)\ is\ odd, \\ i+j-1, & if\ i+j < n+3, \\ i+j-1-n, & otherwise. \end{cases}$$

Considering the colors of neighboring edges of $v_1, v_i\left(2 \leq i \leq \frac{n}{2}+1\right) and\ v_j\left(\frac{n}{2}+2 \leq j \leq n\right)$ separately, it is easy to derive that the mapping is a proper $n+1$-coloring and the respective values of $Sum(v_i)$ for $i = 1,2,\dots,n$ are pairwise distinct.

Theorem 8: If $n = 12k+2,\ k > 0,\ k \in \mathbb{Z}$, then $\omega_{am}(K_n) \leq n+2$.

Proof: It is sufficient to construct an antimagic $(n+2)$-coloring (let us call it γ) of K_n when n satisfies the theorem conditions. We define γ in the following way:

$$\gamma(v_i v_j) = \begin{cases} n+2, & if\ i+j = \frac{n}{2}+1\ or\ i+j = \frac{3n}{2}+1, \\ \big((i+j)\ mod\ (n+1)\big) + 1, & otherwise. \end{cases}$$

Let us calculate for each vertex which colors are not present in its neighboring edges. It is easy to see that for $i = 1, 2, \dots, \frac{n}{2}$, the missing colors in neighboring edges of v_i are $i+1, 2i+1$, and $\begin{cases} n+2, & if\ i = \frac{n+2}{4}, \\ \frac{n}{2}+2, & otherwise. \end{cases}$ For $i = \frac{n}{2}+1, \frac{n}{2}+2, \dots, n$, the missing colors are $i+1, 2i-n$, and $\begin{cases} n+2, & if\ i = \frac{3n+2}{4}, \\ \frac{n}{2}+1, & otherwise. \end{cases}$ For each v_i, the missing numbers are distinct, moreover, for each v_i and v_j $(i \neq j)$, the sums of missing numbers are distinct, so γ is an antimagic $(n+2)$-coloring.

REFERENCES

1. *Alon N., Kaplan G., Lev A., Roditty Y. and R. Yuster*, Dense graphs are antimagic, J. Graph Theory 47 (2004). PP. 297–309.
2. *Bérczi K., Bernáth A. and Vizer M.* Regular graphs are antimagic, The Electronic Journal of Combinatorics, 22, 2015.
3. *Chang F., Liang Y.-C., Pan Z. and Zhu X.* Antimagic labeling of regular graphs, J. Graph Theory, 82 (2016). PP. 339–349.
4. *Cheng Y.* A new class of antimagic Cartesian product graphs, Discrete Math., 308 (2008). PP. 6441–6448.
5. *Cheng Y.* Lattice grids and prisms are antimagic, Theoretical Computer Science 374 (2007). PP. 66–73.
6. *Cranston D., Liang Y.-C. and Zhu X.* Regular graphs of odd degree are antimagic, J. Graph Theory 80 (2015). PP. 28–33.
7. *Eccles T.* Graphs of large linear size are antimagic, J. Graph Theory, 81 (2016). PP. 236–261.
8. *Hartsfield N. and Ringel G.* Pearls in Graph Theory, Academic Press, INC., Boston, 1990. PP. 108–109, Revised version, 1994.
9. *Kaplan G., Lev A. and Roditty Y.* On zero-sum partitions and anti-magic trees, Discrete Math., 309 (2009). PP. 2010–2014.
10. *Wang T.-M.* Toroidal grids are anti-magic, Computing and combinatorics, Lecture Notes in Comput. Sci., 3595, Springer, Berlin (2005) PP. 671–679.
11. *West D.* Introduction to Graph Theory, Prentice-Hall, New Jersey, 2001.

12. *Պետրոսյան Պ.Ա., Մկրտչյան Վ.Վ., Քամալյան Ռ.Ռ.* Գրաֆների տեսություն, ուսումն. ձեռն., Եր. ԵՊՀ հրատ., 2015:

ОБ АНТИМАГИЧЕСКИХ РЕБЕРНЫХ РАСКРАСКАХ НЕКОТОРЫХ ГРАФОВ

Г. Микаелян

Ереванский Государственный университет

АННОТАЦИЯ

Для данного графа G и приведенной на нем правильной реберной t-раскраски α, обозначим через $Sum_G(v,\alpha)$ сумму цветов ребер инцидентных вершине $v \in V(G)$. Тогда α называется антимагической реберной t-раскраской графа G, если для каждой пары различных вершин $v_1, v_2 \in V(G)$, $Sum_G(v_1,\alpha) \neq Sum_G(v_2,\alpha)$. Множество графов G, для которых существует некоторое целое число t такое, что G допускает антимагическую реберную t-раскраску, обозначается через $\mathcal{AM}$. Для произвольного графа G из множества $\mathcal{AM}$, обозначим через $\omega_{am}(G)$ наименьшее положительное целое число t, для которого G допускает антимагическую реберную t-раскраску, а через $\Omega_{am}(G)$ наибольшее целое число t, для которого G допускает антимагическую реберную t-раскраску. В данной работе найдены некоторые оценки и некоторые точные результаты параметров $\omega_{am}(G)$ и $\Omega_{am}(G)$ для колес, некоторых Халин графов и полных графов.

Ключевые слова: антимагическая реберная раскраска, реберная раскраска.

DOI 10.48200/1829-0450_pmn_2022_2_46
УДК 51-76

Поступила: 01.12.2022г.
Сдана на рецензию: 02.12.2022г.
Подписана к печати: 15.12.2022г.

COMPARISON OF DATA MATCHING METHODS ON BIOMEDICAL DATASETS

T. Galstyan

Russian-Armenian University

tigran@yerevann.com

ABSTRACT

We study the efficiency of recently developed state-of-the-art entity resolution methods on real-life biomedical datasets. We first formalize the entity resolution problem, also known as dataset matching. We consider following matching problem settings: without outliers, outliers in only one dataset, outliers in both datasets. We proceed to analyze and carefully preprocess biomedical dataset pairs considered in our experiments. We show that recent algorithms constantly outperform the original greedy algorithm in all settings. We also examine a newly proposed procedure which estimates the unknown number of inliers with no additional information and we successfully show that algorithms using this estimation procedure almost match the performance of the algorithms which were given the true number of inliers as an input.

Keywords: entity resolution, record linkage, single-cell transcriptomics, spatial proteomics.

1. Introduction

Dataset matching, also known as record linkage or entity resolution, is a problem of finding the same entities across different datasets. This task

arises in many practical applications such as object detection, camera position estimation, biomedical data analysis, etc. [1, 2]. In object detection, dataset matching is used to match local keypoint descriptors on different images, assuming they contain the same object. Ideally, two descriptors are matched if they represent the projections of the same 3D point in two images.

In single-cell biology research, it is common to have datasets containing measurements on overlapping sets of cells, with similar (but different) experimental protocols and post/pre-processing steps. So it is both possible and non-trivial to match entities from different biomedical datasets to get an enhanced description of each entity.

In this paper, we use the following statistical framework to tackle the dataset matching problem.

Let $X = \{X_1, X_2, \dots, X_n\}$ and $X^{\#} = \{X^{\#}{}_1, X^{\#}{}_2, \dots, X^{\#}{}_m\}$ be sets of d-dimensional vectors of sizes $n, m > 1$, respectively. We will assume that the data is generated according to the following model:

$$\begin{aligned} X_i &= \theta_i + \sigma\xi_i, i = 1, \dots, n \\ X^{\#}{}_j &= \theta^{\#}{}_j + \sigma\xi^{\#}{}_j, j = 1, \dots, m \end{aligned} \quad (1)$$

Here $\Theta = \{\theta_1, \dots, \theta_n\}$ and $\Theta^{\#} = \{\theta^{\#}{}_1, \dots, \theta^{\#}{}_n\}$ are sets of original unobserved vectors, ξ_i and $\xi^{\#}{}_j$ are i.i.d. standard Gaussian random d-dimensional vectors and σ is the noise magnitude. We only observe the noisy sets X and $X^{\#}$. We also assume that there is an underlying injective mapping π^* from the set $S^* \subset [n]$ (where $[n] = \{1, 2, \dots, n\}$) of size k^* ($k^* \leq n$) into the set $[m]$: $\pi^*: S^* \to [m]$ such that for all $i \in S^*, \theta_i = \theta^{\#}{}_{\pi^*(i)}$. We will call the observations from the set S^* and their corresponding matches (the support and the image of π^*) *inliers* and the rest of the vectors *outliers*. Our objective will be detecting unknown π^* as accurately as possible having access to only X and $X^{\#}$. In this paper, we will concentrate on biological datasets. In particular, we will analyze two pairs of datasets, a

pair of single-cell RNA sequencing datasets and a pair of proteomics datasets. We will discuss datasets in detail in Section 2.

Matching problem under condition (1) has been studied first by Collier and Dalalyan [3] under more strict restrictions: $n = m$, no outliers. The extension of this work on a more general case in the presence of outliers was done in [4] and [5]. Chen et al. [6] proposed a method called *LAPS* (Linear Assignment with Projected Signals). Chen et al. showed that *LAPS* can be successfully applied on present-day biomedical datasets.

The goal of the current paper is to further analyze several state-of-the-art data matching algorithms on real-life biomedical datasets. The rest of the paper will go as follows: first we will discuss datasets and matching algorithms in detail in Sections 2 and 3, then Section 4 will discuss experiments and results.

2. Experimental Setup and Datasets

This section describes different problem settings of the matching problem we included in our empirical evaluation and steps we took to process the data to ensure that they are consistent with the problem settings. We can divide experiments performed in this paper into three categories: no outliers ($k^* = n = m$), outliers in only one set ($k^* = n < m$) and outliers in both sets ($k^* < n, k^* < m$). We used two pairs of biological datasets.

2.1 Single-cell RNA sequencing data

The first dataset pair under consideration is a pair of single-cell RNA sequencing datasets from human pancreatic islets. It comprises a dataset (referred as *Celseq*) obtained using CEL-seq2 technology [7] and a second one (*Smartseq*) obtained with Smart-seq2 [8]. Celseq data has 34363 RNAs measured across 2285 cells, while Smartseq has 26178 RNAs measured across 2394 cells. Both raw datasets can be found in *SeuratData* package of R [9]. We applied the following standard preprocessing steps identically to both Celseq and Smartseq datasets using Python Scanpy package [10]. First,

we filtered out RNAs that appeared in less than 25 cells; this filtered around half of the RNAs. Then we normalized both datasets across cells with a target count sum of 10000, meaning that after normalization each cell had 10000 total counts over all RNAs. Then we proceeded to select 5000 most active RNAs per dataset and filtered out the rest. As the RNA names are available in the data, we selected RNAs which appear in both datasets and again filtered out the rest. We ended up with two sets of sizes 2285 x 2808 and 2394 x 2808 respectively. Top 10 most active RNAs per dataset are shown in Fig 1.

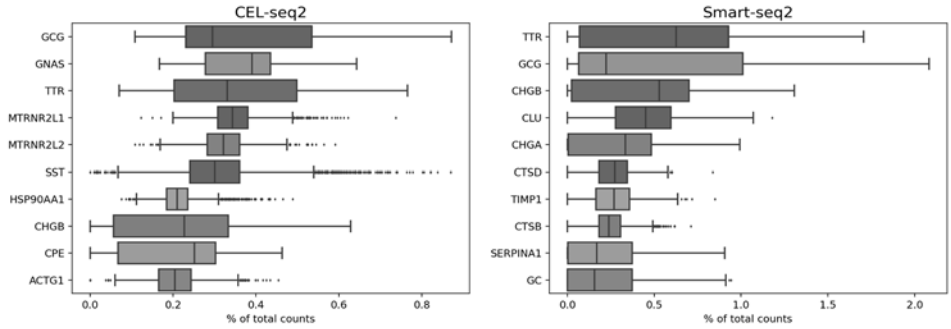


Figure 1. Highest expression RNAs for Celseq (left) and Smartseq (right) datasets.

As there is no ground-truth one-to-one matching for cells in these datasets, we worked on cell-type level, meaning that we assumed a match is correct if it matches cells of the same type. Human annotation of cell types is available for both datasets. As cell types are not perfectly aligned in datasets we performed following type-balancing steps. For each type, first we checked which dataset has fewer cells of this particular type, then we randomly downsampled cells from the other dataset of that type to leave exactly the same number of cells per type. Eventually we achieve two datasets of size (1935 x 2808). Cell type distributions after balancing are displayed in Fig 2. This corresponds to the description of the experiment with no outliers ($k = n = m$), as we will aim to match cells and all of

them can theoretically match a cell of the same type (no outliers). Steps for transforming this data for other setups will be discussed in later sections.

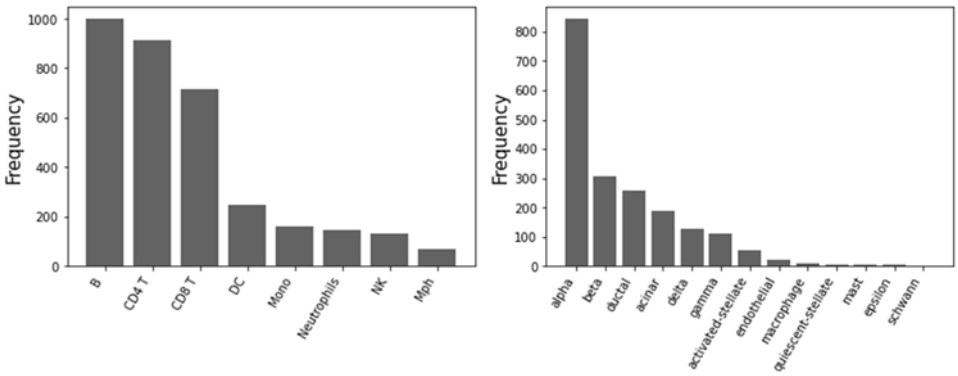


Figure 2. Number of cells-per-type for murine spleen proteomics (left) and human pancreas (right) dataset pairs.

2.2. Murine Spleen Proteomics data

Second pair of datasets was obtained from murine spleen using CITE-seq [9] and CODEX [11] technologies. As different technologies surface different biological information (e.g. CODEX provides spatial coordinates of collections of cells on the slice of tissue cut) aligning this kind of datasets can serve as an easy method for enriching results of biological experiments with knowledge from existing datasets. Raw CITE-seq dataset contains measurements of 208 proteins in 7601 cells, while the CODEX dataset measures 31 proteins in 48332 cells. Both raw datasets can be downloaded from the github page of MARIO[1] proteomic data integration pipeline. Similar to the previous pair, preprocessing and celltype-balancing steps were applied to CITE-seq and CODEX datasets resulting in a pair of sets of size (3381 x 28). CITE-seq-CODEX pair is significantly harder to match

[1] https://github.com/shuxiaoc/mario-py

because first its lower-dimensional ($d = 28$) and signal-to-noise ratio here is significantly lower. For this pair also, evaluation was done on celltype-level, as it's the only human-annotated label for the data. Distribution of the cell types after balancing are presented in Fig 2.

3. Data Matching Algorithms

The most straightforward approach to trying to recover π^* is to match vectors from X to $X^{\#}$ greedily. There are several greedy algorithms for this problem. The most simple and easy-to-implement algorithm works as follows: vectors are taken from the set X one-by-one, in original order, and matched to their closest vector from $X^{\#}$ by some predefined distance measure. This algorithm is used in OpenCV (as BFMatcher) [12] to match keypoint descriptors of images. In this work we used a slightly more sophisticated greedy algorithm, which on every step matches the pair of vectors $X_i \in X$ and $X_j{}^{\#} \in X^{\#}$ such that the distance between them is minimal ($argmin_{i,j} \left|\left|X_i - X_j{}^{\#}\right|\right|_2$). Data points matched at each step are not considered in the following steps. Notice, the only difference is that the second algorithm doesn't go over the set X in some random or pre arranged order, but takes the vector X_i which has the closest possible match. We chose the second algorithm to use in this work because it doesn't depend on the ordering of the set X and works consistently while still achieving similar results, being fast and straightforward to implement.

Another matching method from [4] can be defined by the following optimization problem:

$$\hat{\pi}^{LSS} \in argmin \sum_1^n \left|\left|X_i - X_{\pi(i)}{}^{\#}\right|\right|_2^2 \quad (2)$$

We will call (2) *Least Sum of Squares* (LSS) estimator. LSS can be computed using the Hungarian algorithm [13]. It is a polynomial-time algorithm ($O(n^3)$) to solve the assignment problem.

The greedy algorithm can be easily extended to the cases with outliers. The Hungarian algorithm has a natural extension for the case where there are outliers on one side ($k = n < m$) [14]. We will discuss the extension to the general case, where two sides contain outliers below.

3.1 Assignment as a Minimum-Cost-Flow Problem

As described in [5], if we want to solve (2) for matchings of fixed size $k < n$, we can reformulate (2) as a minimum-cost flow problem on a weighted directed bipartite graph with two additional nodes. The graph is constructed as follows. First, all vectors from both X and $X^{\#}$ are assigned a node in the graph. All nodes of vectors from X are connected to all nodes from $X^{\#}$ with edges with cost of $\left|\left|X_i - X_j{}^{\#}\right|\right|_2$ and capacity equal to 1 for all pairs i, j, creating a bipartite graph. Then two additional nodes are created: a *source* node and a *sink* node. The *source* node is connected to all nodes of X with edges of capacity 1 and zero cost. Similarly, the *sink* node is connected to all nodes of $X^{\#}$ (edges directed to the *sink*), again with 1-capacity zero-cost edges. The resulting graph is shown in Fig. 3.

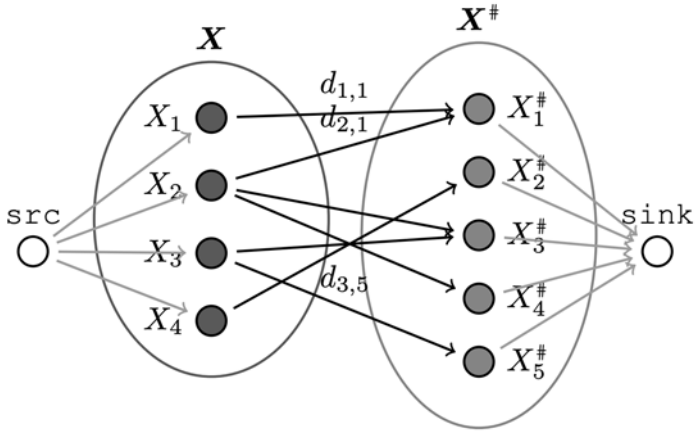


Figure 3. Illustration of the flow network corresponding to the optimization problem described in (2). Illustration from [5].

4. Experimental Results

First experiment is performed in the simplest setup of no outliers in both datasets. As we have two pairs of datasets of equal sizes, all entities were matched. The results are shown in Fig. 4 for the Celseq-Smartseq pair and in Fig. 5 for the CITE-seq-CODEX pair. One can observe that LSS consistently outperforms the greedy algorithm on both pairs. The second experiment is designed the following way to measure algorithms' performance in case of the presence of outliers in one of the sets. To ensure the presence of outliers we select one specific cell type and completely remove cells of that type from one of the datasets: *Smartseq* in human pancreas pair and *CODEX* in the murine spleen proteomics pair. By doing so we can guarantee the presence of outliers in the opposite set, because all of the corresponding matches of a certain type have been removed. The results are presented in Fig. 6 and Fig. 7 for each pair of datasets respectively.

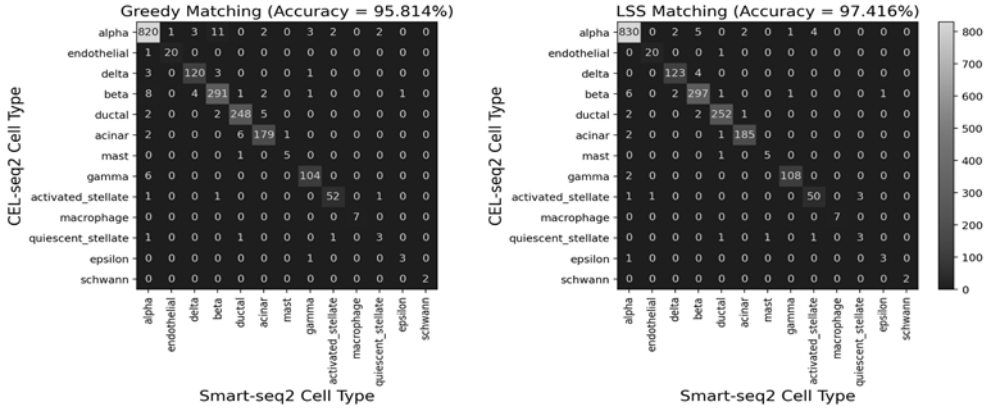


Figure 4. Confusion matrices for RNA-seq dataset pair matchings (greedy algorithm on the left-hand matrix, LSS on the right-hand). Celltype-level accuracy is reported.

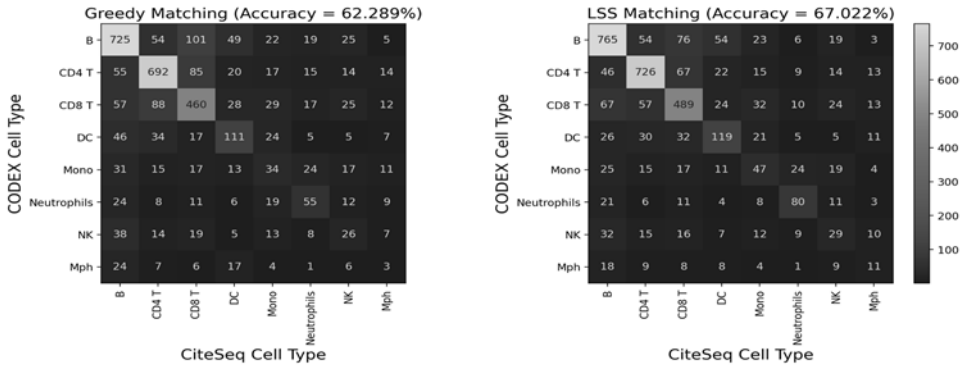


Figure 5. Confusion matrices for murine spleen proteomics dataset pair matchings (greedy algorithm on the left-hand matrix, LSS on the right-hand). Celltype-level accuracy is reported.

Again we see LSS consistently outperforming the greedy algorithm in terms of matching accuracy regardless of the outlier cell type. On the other hand, the outlier detection accuracy is nearly the same for both algorithms in all of the cases. Outlier detection accuracy here means the rate of outliers left out of the matching (as $k = n < m$, there will be $(m - k)$ vectors from $X^{\#}$ left out of a chosen matching π of size k).

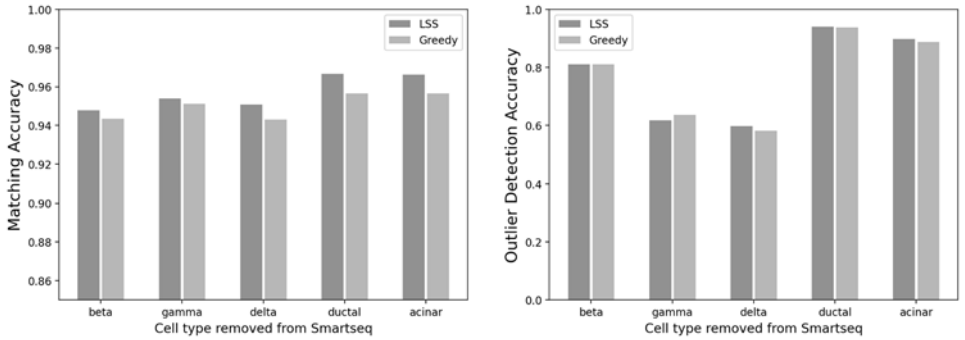


Figure 6. Left: The accuracy of different matchers on human pancreas dataset pair in case of one outlier cell type in Celseq dataset (removed from Smartseq). Right: Percentage of outliers in Celseq dataset not included in the matching, higher the better.

We chose top cell types by the number of occurrences for this experiment: left out type '*alpha*' for the first pair because it constitutes almost half of the dataset.

The third and last experimental setting is similar to the previous one, except that there are outliers in both X and $X^{\#}$ ($k < n, k < m$). In this case, we will be removing all cells of a fixed cell types from both X and $X^{\#}$, one per dataset. The main point of interest of this experiment is to evaluate the algorithm for estimating the number of inliers k proposed by [5] in Section 3.

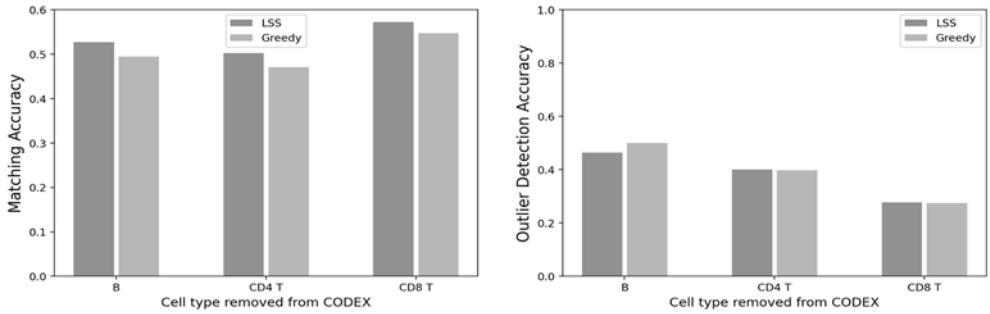


Figure 7. Left: The accuracy of different matchers on murine spleen pair in case of one outlier cell type in CITE-seq dataset (removed from CODEX).
Right: Percentage of outliers in CITE-seq dataset not included in the matching (left unmatched), the higher the better.

As the murine spleen dataset pair has a very low signal-to-noise ratio and is very far from satisfying the constraints of Theorem 1 of [5], here we concentrate only on RNA-sequencing datasets. For a fair comparison, we did not compare results with the estimated number of inliers with vanilla LSS / Greedy algorithms, as they are forced to match every vector (including outliers). We compare the algorithm with the estimated number of inliers with LSS / Greedy forced to match k^* overall entities (correct number of inliers). This will put the initial algorithm in a disadvantageous position as it will have less initial information than the algorithms that will

know the correct number of inliers. Nevertheless, as one can see on Fig 8., even with the disadvantage the algorithm from [5] manages to achieve similar (even sometimes better) results.

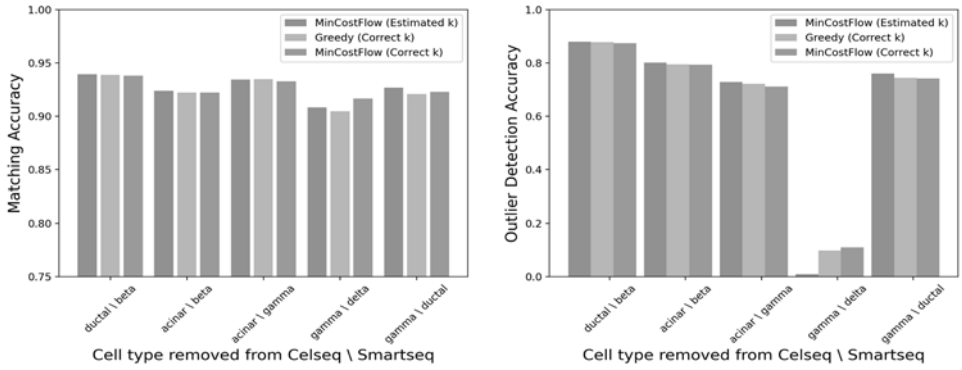


Figure 8. The comparison of an algorithm unaware of the number of inliers (MinCostFlow estimated k) and algorithms which had the correct number of inliers as input.

All three methods achieve overall mean accuracy of 91.8% – 92.1%. Results of this experiment provide additional experimental verification of the methods proposed in [5] and show that they can be used on real-life datasets with realistic settings, even without any prior knowledge about the true number of inliers / outliers.

5. Conclusion

We have analyzed the problem of dataset matching in real-life settings. Our analysis, carried out on two biomedical datasets, showed that the algorithms based on the *LSS-Min Cost Flow* methodology consistently outperform greedy algorithms in simple cases of no outliers and outliers in one of the sets. Furthermore, the reported results demonstrate the consistency of the procedure of estimation of the number of inliers. Indeed, its results are comparable with the *oracle* algorithms that use the correct number of inliers. Therefore, a high level conclusion of this paper is that the

algorithm obtained by combining the LSS-Min Cost Flow methodology with the model selection approach can be recommended for use in practice.

REFERENCES

1. *Jin Y., Mishkin D., Mishchuk A. et al.* Image Matching across Wide Baselines: From Paper to Practice, International Journal of Computer Vision, 2020.
2. *Barkas N., Petukhov V., Nikolaeva D. et al.* Joint analysis of heterogeneous single-cell RNA-seq dataset collections, Nature Methods, 2019.
3. *Collier O., Dalalyan A.* Minimax rates in permutation estimation for feature matching, The Journal of Machine Learning Research, 162–192, 2016.
4. *Galstyan T., Minasyan A., Dalalyan A.* Optimal detection of the feature matching map in presence of noise and outliers, Electronic Journal of Statistics, 16(2), 5720–5750, 2022.
5. *Minasyan A., Galstyan T., Hunanyan S., Dalalyan, A.* Matching Map Recovery with an Unknown Number of Outliers, arXiv preprint arXiv: 2210.13354, 2022.
6. *Chen S., Jiang S., Ma Z.*, et al. One-Way Matching of Datasets with Low Rank Signals, arXiv preprint arXiv:2204.13858, 2022.
7. *Hashimshony T., Senderovich N., Avital G., Rozenblatt-Rosen O*., et al. Cel-seq2: sensitive highly-multiplexed single-cell rna-seq. Genome biology, 17(1): 1–7, 2016.
8. *Picelli S., Björklund Å. K., Faridani O. R.,* et al. Smart-seq2 for sensitive full-length transcriptome profiling in single cells, Nature methods, 10 (11): 1096–1098, 2013.
9. *Stoeckius M., Hafemeister C., Stephenson* W. et al. Simultaneous epitope and transcriptome measurement in single cells. Nature Methods, 14(9): 865–868, 2017.
10. *Wolf F., Angerer P., Theis F.* SCANPY: large-scale single-cell gene expression data analysis, Genome Biology, 2018.
11. *Goltsev Y., Samusik N., Kennedy-Darling J.,* et al. Deep profiling of mouse splenic architecture with codex multiplexed imaging, Cell, 174 (4): 968–981, 2018.
12. *Bradski G.* The OpenCV Library, Dr. Dobb's Journal of Software Tools, 2000.
13. *Harold W. Kuhn*. A tale of three eras: The discovery and rediscovery of the Hungarian method, European Journal of Operational Research, 219(3): 641–651, 2012.
14. *Bourgeois F., Lassalle J.C.* An extension of the Munkres algorithm for the assignment problem to rectangular matrices, Communications of the ACM 14.12: 802–804, 1971.

СРАВНЕНИЕ МЕТОДОВ РАЗРЕШЕНИЯ СУЩНОСТЕЙ НА БИОМЕДИЦИНСКИХ НАБОРАХ ДАННЫХ

Т. Галстян

АННОТАЦИЯ

В данной статье изучена эффективность современных методов разрешения сущностей на реальных наборах биомедицинских данных. Сначала мы формализуем проблему разрешения сущностей, также известную как «сопоставление данных». Мы рассматриваем следующие постановки задачи сопоставления: без выбросов, выбросы только в одном наборе данных, выбросы в обоих наборах данных. Мы анализируем и предварительно тщательно обрабатываем пары наборов биомедицинских данных, рассматриваемых в наших экспериментах. Мы показываем, что современные алгоритмы превосходят исходный жадный алгоритм во всех настройках. Мы также рассматриваем недавно предложенную процедуру, которая оценивает неизвестное количество выбросов без дополнительной информации, и мы успешно показываем, что алгоритмы, использующие эту процедуру оценки, почти не уступают по эффективности алгоритмам, которые получили истинное количество выбросов в качестве входных данных.

Ключевые слова: разрешение сущностей, сопоставление данных, транскриптомика одиночных клеток, структурная протеомика.

DOI 10.48200/1829-0450_pmn_2022_2_59
UDC 519.174.7

Поступила: 29.03.2022г.
Сдана на рецензию: 30.03.2022г.
Подписана к печати: 15.04.2022г.

STRONG EDGE COLORING OF HAMMING GRAPHS IN SPECIAL CASE

A. Drambyan

Russian-Armenian University

ardrambyan@student.rau.am

ABSTRACT

An edge coloring ϕ of a graph G is called strong if any two edges at distance at most 2 receive different colors. The minimum number of colors required for a strong edge coloring of a graph G is called strong chromatic index of graph G and denoted by ${\chi_s}'(G)$. Hamming graph $H(n,m)$ is the Cartesian product of n complete graphs K_m. In this paper, for Hamming graphs $H(n,m)$, we show that $\frac{(2n-1)m(m-1)}{2} \leq \chi_s'(H(n,m)) \leq \frac{nm^2(m-1)}{2}$. Besides we construct strong edge coloring for $H(n,4)$ that improves upper bound to $12n-6 \leq \chi_s'(H(n,4)) \leq 12n$.

Keywords: Strong chromatic index, Strong edge coloring, Hamming graphs.

Introduction

All graphs considered in this paper are finite and simple. We denote by $V(G)$ and $E(G)$ the sets of vertices and edges of a graph G, respectively.

The degree of a vertex $v \in V(G)$ is denoted by $d(v)$ and the maximum degree of vertices in G by $\Delta(G)$.

An *edge coloring* of a graph G is a mapping $\phi: E(G) \to \mathbb{N}$. ϕ is called *strong* if any two edges at distance at most 2 receive different colors. The minimum number of colors required for a strong edge coloring of a graph G is called *strong chromatic index* of graph G and denoted by $\chi_s'(G)$.

Strong edge coloring of graphs was introduced by Fouquet and Jolivet in 1983 [3]. Later, during seminar in Prague, Erdős and Nešetřil proposed the following conjecture.

Conjecture 1. *For every graph G with maximum degree $\Delta(G)$,*

$$\chi_s'(G) \leq \begin{cases} \frac{5}{4}\Delta(G)^2, & if \quad \Delta(G) \ is \ even, \\ \frac{1}{4}(5\Delta(G)^2 - 2\Delta(G) + 1), & if \quad \Delta(G) \ is \ odd. \end{cases}$$

Conjecture was proved for $\Delta(G) = 3$ by [1] and [4] independently. For $\Delta(G) = 4$ it's shown that $\chi_s'(G) \leq 22$ [2], while the Conjecture 1 says that $\chi_s'(G) \leq 20$. Hurley et al. [5] showed that for graph G with sufficiently large $\Delta(G)$, $\chi_s'(G) \leq 1.772\Delta(G)^2$ and this is the currently best known upper bound for general graphs.

Graph, where each pair of vertices are connected with an edge, is called *complete* and denoted by K_n. It is obvious that $\chi_s'(K_n) = |E(K_n)| = \frac{n(n-1)}{2}$. The *Cartesian product* $G \quad H$ of graphs G and H is a graph with set of vertices $V(G) \times V(H)$, and 2 vertices $u = (u_1, u_2)$ and $v = (v_1, v_2)$ are adjacent if $u_1 = v_1$ and u_2 and v_2 are adjacent in H or $u_2 = v_2$ and u_1 and v_1 are adjacent in G. *Hamming graph* $H(n, m)$ is the Cartesian product of n complete graphs K_m. Vertices of the Hamming graph can be represented as a tuples of length n, where each position can take a value from 0 to $m - 1$, and 2 vertices are adjacent if and only if they are equal in all but one position.

Main Results

We begin our considerations with strong edge coloring of Hamming graphs.

Theorem 1. *Let $H(n,m)$ be a Hamming graph with $m \geq 2$. Then*

$$\frac{(2n-1)m(m-1)}{2} \leq \chi_s'(H(n,m)) \leq \frac{nm^2(m-1)}{2}$$

Proof In order to proof the lower bound let us consider m vertices $x_1 = (0,0,\ldots,0)$, $x_2 = (1,0,\ldots,0)$, ..., $x_m = (m-1,0,\ldots,0)$. They all are at distance 1 from each other and all the edges adjacent to that vertices should receive different colors. For any vertex x from $H(n,m)$, $d(x) = n(m-1)$. We get $\chi_s'(H(n,m)) \geq md(x) - \frac{m(m-1)}{2} = nm(m-1) - \frac{m(m-1)}{2} = \frac{(2n-1)m(m-1)}{2}$.

For the upper bound let us construct edge classes $E_{k,i,j,r} = \{xy \mid$ where x and y differs at position k, $x_k = i, y_k = j$ and the sum of all positions of x (or y) except k-th position, by $mod\ m$, equals to $r\}$, where $1 \leq k \leq n$, $1 \leq i < j \leq m$, $0 \leq r < m$. Any edge from $H(n,m)$ belongs to exactly one of these sets and edges from the same edge class are at distance more than 2 from each other, which means that each set can be colored using one color. We get $\chi_s'(H(n,m)) \leq \frac{nm^2(m-1)}{2}$.

Notice that $H(n,2)$ is the n-dimensional cube. And the upper bound from Theorem 1 matches with the result from [6]. Next, we consider strong edge coloring of Hamming graphs when $m = 4$.

Theorem 2. *Let $H(n,4)$ be a Hamming graph. Then*

$$12n - 6 \leq \chi_s'(H(n,4)) \leq 12n.$$

Proof

Lower bound follows from Theorem 1. We will call strong edge colorings ϕ_1 and ϕ_2 of the graph G *combinable*, if after joining same vertices of ϕ_1-colored G and ϕ_2-colored G to each other, coloring stays strong for the newly formed graph. Let us construct 4, combinable between each other strong edge colorings ϕ_1, ϕ_2, ϕ_3 and ϕ_4 of $H(n,4)$, that use $12n$ colors in total by induction on n. For base case $n = 1$:

$$\phi_1(e) = \begin{cases} 1 & e = ((0),(1)) \\ 2 & e = ((2),(3)) \\ 3 & e = ((0),(3)) \\ 4 & e = ((1),(2)) \\ 5 & e = ((0),(2)) \\ 6 & e = ((1),(3)) \end{cases} \qquad \phi_2(e) = \begin{cases} 1 & e = ((2),(3)) \\ 2 & e = ((0),(1)) \\ 3 & e = ((1),(2)) \\ 4 & e = ((0),(3)) \\ 5 & e = ((1),(3)) \\ 6 & e = ((0),(2)) \end{cases}$$

$$\phi_3(e) = \begin{cases} 7 & e = ((0),(1)) \\ 8 & e = ((2),(3)) \\ 9 & e = ((0),(3)) \\ 10 & e = ((1),(2)) \\ 11 & e = ((0),(2)) \\ 12 & e = ((1),(3)) \end{cases} \qquad \phi_4(e) = \begin{cases} 7 & e = ((2),(3)) \\ 8 & e = ((0),(1)) \\ 9 & e = ((1),(2)) \\ 10 & e = ((0),(3)) \\ 11 & e = ((1),(3)) \\ 12 & e = ((0),(2)) \end{cases}$$

We will use these colorings. Denote by $(a_1, a_2, \dots, a_k)H(n-k,m)$ sub-graph of $H(n,m)$ that contains vertices x, for which first k positions of the tuple are equal to $a_1, a_2, \dots, a_k$ accordingly, $0 \le a_i < m$ for $i = 1, \dots, k$. $H(n,4)$ can be represented as a combination of $(0)H(n-1,4)$, $(1)H(n-1,4)$, $(2)H(n-1,4)$ and $(3)H(n-1,4)$, where vertices from 2 components are connected if they differ only in the first position.

In the future we will refer to the edges, that connects vertices from different components, as *connector edges*. By induction we have 4 strong edge colorings ϕ_1, ϕ_2, ϕ_3 and ϕ_4 of $H(n-1,4)$ that use $12(n-1)$ colors in total. For the first coloring ϕ'_1 of $(H(n,4))$, we color $(0)H(n-1,4)$ using ϕ_1, $(1)H(n-1,4)$ using ϕ_2, $(2)H(n-1,4)$ using ϕ_3 and $(3)H(n-$

1,4) using ϕ_4. Easy to see that the set of connector edges $(i)H(n-1,4)-(j)H(n-1,4)$ $(i<j)$ is the same as $E_{1,i,j,0} \cup E_{1,i,j,1} \cup E_{1,i,j,2} \cup E_{1,i,j,3}$.

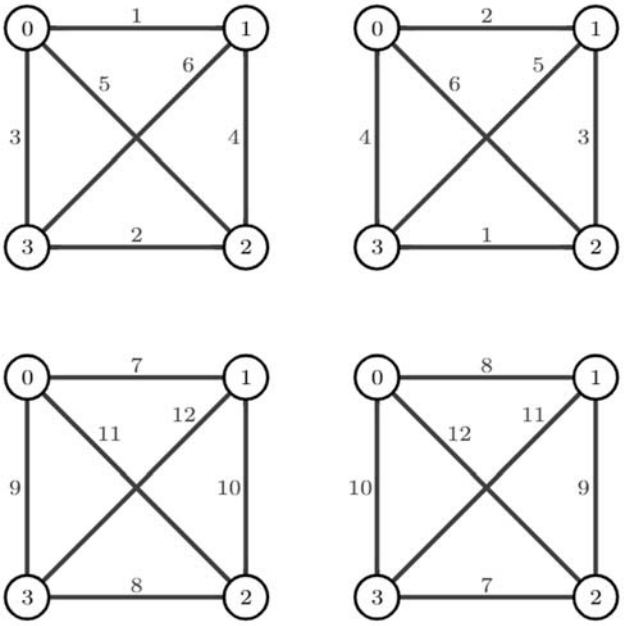


Figure 1. Visualization of ϕ_1, ϕ_2, ϕ_3 and ϕ_4 colorings for base case $n=1$.

In the future, under $(c_1,c_2,c_3,c_4)-coloring$ of $(i)H(n-1,4)-(j)H(n-1,4)$ connector edges, we will understand assignment of a color c_1 to $E_{1,i,j,0}$, color c_2 to $E_{1,i,j,1}$, color c_3 to $E_{1,i,j,2}$ and color c_4 to $E_{1,i,j,3}$ edge classes.We Complete ϕ'_1 coloring with:

$$\begin{cases}(12n-11,12n-10,12n-9,12n-8)-coloring\ of\ (0)H(n-1,4)-(1)H(n-1,4)\\(12n-7,12n-6,12n-5,12n-4)-coloring\ of\ (0)H(n-1,4)-(2)H(n-1,4)\\(12n-3,12n-2,12n-1,12n)-coloring\ of\ (0)H(n-1,4)-(3)H(n-1,4)\\(12n-8,12n-9,12n-10,12n-11)-coloring\ of\ (2)H(n-1,4)-(3)H(n-1,4)\\(12n-4,12n-5,12n-6,12n-7)-coloring\ of\ (1)H(n-1,4)-(3)H(n-1,4)\\(12n,12n-1,12n-2,12n-3)-coloring\ of\ (1)H(n-1,4)-(2)H(n-1,4)\end{cases}$$

Other 3 colorings can be constructed in a similar way:

$$\phi'_2 = \begin{cases} \phi_2 \ coloring\ of\ (0)H(n-1,4) \\ \phi_3 \ coloring\ of\ (1)H(n-1,4) \\ \phi_4 \ coloring\ of\ (2)H(n-1,4) \\ \phi_1 \ coloring\ of\ (3)H(n-1,4) \\ (12n-10,12n-9,12n-8,12n-11) - coloring\ of\ (0)H(n-1,4) - (1)H(n-1,4) \\ (12n-6,12n-5,12n-4,12n-7) - coloring\ of\ (0)H(n-1,4) - (2)H(n-1,4) \\ (12n-2,12n-1,12n,12n-3) - coloring\ of\ (0)H(n-1,4) - (3)H(n-1,4) \\ (12n-9,12n-10,12n-11,12n-8) - coloring\ of\ (2)H(n-1,4) - (3)H(n-1,4) \\ (12n-5,12n-6,12n-7,12n-4) - coloring\ of\ (1)H(n-1,4) - (3)H(n-1,4) \\ (12n-1,12n-2,12n-3,12n) - coloring\ of\ (1)H(n-1,4) - (2)H(n-1,4) \end{cases}$$

$$\phi'_3 = \begin{cases} \phi_3 \ coloring\ of\ (0)H(n-1,4) \\ \phi_4 \ coloring\ of\ (1)H(n-1,4) \\ \phi_1 \ coloring\ of\ (2)H(n-1,4) \\ \phi_2 \ coloring\ of\ (3)H(n-1,4) \\ (12n-9,12n-8,12n-11,12n-10) - coloring\ of\ (0)H(n-1,4) - (1)H(n-1,4) \\ (12n-5,12n-4,12n-7,12n-6) - coloring\ of\ (0)H(n-1,4) - (2)H(n-1,4) \\ (12n-1,12n,12n-3,12n-2) - coloring\ of\ (0)H(n-1,4) - (3)H(n-1,4) \\ (12n-10,12n-11,12n-8,12n-9) - coloring\ of\ (2)H(n-1,4) - (3)H(n-1,4) \\ (12n-6,12n-7,12n-4,12n-5) - coloring\ of\ (1)H(n-1,4) - (3)H(n-1,4) \\ (12n-2,12n-3,12n,12n-1) - coloring\ of\ (1)H(n-1,4) - (2)H(n-1,4) \end{cases}$$

$$\phi'_4 = \begin{cases} \phi_4 \ coloring\ of\ (0)H(n-1,4) \\ \phi_1 \ coloring\ of\ (1)H(n-1,4) \\ \phi_2 \ coloring\ of\ (2)H(n-1,4) \\ \phi_3 \ coloring\ of\ (3)H(n-1,4) \\ (12n-8,12n-11,12n-10,12n-9) - coloring\ of\ (0)H(n-1,4) - (1)H(n-1,4) \\ (12n-4,12n-7,12n-6,12n-5) - coloring\ of\ (0)H(n-1,4) - (2)H(n-1,4) \\ (12n,12n-3,12n-2,12n-1) - coloring\ of\ (0)H(n-1,4) - (3)H(n-1,4) \\ (12n-11,12n-8,12n-9,12n-10) - coloring\ of\ (2)H(n-1,4) - (3)H(n-1,4) \\ (12n-7,12n-4,12n-5,12n-6) - coloring\ of\ (1)H(n-1,4) - (3)H(n-1,4) \\ (12n-3,12n,12n-1,12n-2) - coloring\ of\ (1)H(n-1,4) - (2)H(n-1,4) \end{cases}$$

Easy to see that colorings ϕ'_1, ϕ'_2, ϕ'_3, ϕ'_4 are strong, combinable between each other, cover all the edges and use $12n$ colors in total.

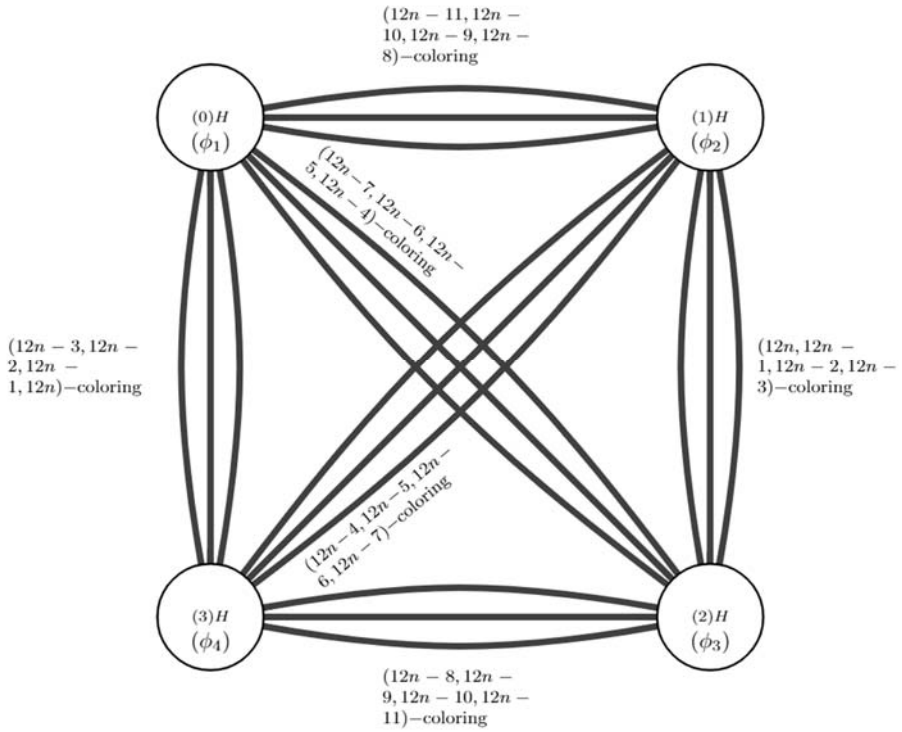


Figure 2. Illustration of ϕ'_1 coloring for $H(n, 4)$.

REFERENCES

1. *Lars Dovling Andersen*. The strong chromatic index of a cubic graph is at most 10. Discrete Mathematics, 108 (1–3): 231–252, 1992.
2. *Daniel W. Cranston*. Strong edge-coloring of graphs with maximum degree 4 using 22 colors. Discrete Mathematics, 306 (21): 2772–2778, 2006.
3. *Jean-Luc Fouquet and Jean-Loup Jolivet*. Strong edge-colorings of graphs and applications to multi-k-gons. Ars Combinatoria A, 16:141–150, 1983.
4. *Peter Hor´ak, He Qing, and William T Trotter*. Induced matchings in cubic graphs. Journal of Graph Theory, 17 (2): 151–160, 1993.

5. *Eoin Hurley, R´emi de Joannis de Verclos, and Ross J Kang*. An improved procedure for colouring graphs of bounded local density. In Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 135–148. SIAM, 2021.
6. *A. Gyarfas R. Faudree, R. Schelp and Zs. Tuza*. The strong chromatic index of graphs. Ars Combinatoria, 29: 205–211, 1990.

СИЛЬНАЯ РЕБЕРНАЯ РАСКРАСКА ГРАФОВ ХЭММИНГА В ЧАСТНОМ СЛУЧАЕ

А. Драмбян

Российско-Армянский университет

АННОТАЦИЯ

В данной статье рассматриваются только простые, конечные графы. Реберная раскраска ϕ графа G называется сильной, если любые 2 ребра на расстоянии не более двух окрашены в различные цвета. Минимальное количество цветов, необходимых для сильной реберной раскраски графа G, называется сильным хроматическим индексом графа G и обозначается $\chi_s{}'(G)$. Графом Хэмминга $H(n,m)$ называется прямое произведения n полных графов K_m. В этой статье, для графов Хэмминга $H(n,m$, показано что $\frac{(2n-1)m(m-1)}{2} \leq \chi_s'(H(n,m)) \leq \frac{nm^2(m-1)}{2}$. Кроме того, в статье показана раскраска для $H(n,4)$ которая улучшает верную оценку до $12n-6 \leq \chi_s'(H(n,4)) \leq 12n$.

Ключевые слова: сильный хроматический индекс, сильная реберная раскраска, графы Хэмминга.

DOI 10.48200/1829-0450_pmn_2022_2_67
УДК 004.8

Поступила: 07.12.2022г.
Сдана на рецензию: 08.12.2022г.
Подписана к печати: 22.12.2022г.

COMPARISON OF SINGLE OBJECT TRACKING ALGORITHMS ON VIDEO SEQUENCES CAPTURED FROM UAV

V. Melkonyan, V. Sahakyan, L. Kirakosyan, O. Hovhannisyan

Russian-Armenian University
Intellectual System and Robotics

vahagn.melkonyan@student.rau.am, vardan.sahakyan@student.rau.am,
kirakosyan.lilia@student.rau.am, olga.hovhannisyan@student.rau.am

ABSTRACT

In this paper, the problem of tracking one object is analyzed. A number of classic algorithms such as MEDIANFLOW tracker, MOSSE tracker, CSRT Tracker, SIFT were considered. And also algorithms based on deep learning, such as SiamFC, SiamTPN and TransT. These algorithms have been implemented on UAVs. Experiments have been carried out in the real world to reveal their strengths and weaknesses.

Keywords: Object Tracking, Computer Vision, Unmanned Aerial Vehicles, Neural Networks.

Intro

Object tracking has been one of the main tasks in the field of Computer Vision due to the complexity and wide range of applications in Video Surveillance, Robotics, Visual Servoing, etc. Changing illumination, pose and scale of the object, fast motions, continuing to track the object,

even if it has been out of the camera's view for some time: these are all non-trivial challenges associated with tracking. Single object tracking aims to continuously track previously initialized objects in the video feed. In the particular case of pre-initialized single object tracking, the situation is complicated further due to the fact that the previously initialized object is not under the limitation of belonging to the subset of previously known objects.

Object tracking methods

During the last 2 decades, a number of interesting approaches have been proposed to solve this task. As a first move, tracking algorithms implemented in the OpenCV [5] library was considered. There are seven classical implementations available: BOOSTING, MIL, KCF, TLD, MEDIANFLOW [1], MOSSE [2], and CSRT [3]. These online trackers combine motion models and appearance models to track the object from frame to frame. We've decided to experiment with the last 3 trackers due to real-time performance constraints.

MEDIANFLOW tracker – The algorithm is based on the Lucas-Kanade method. Internally, it tracks the movement of the object in forward and backward directions in time and measures the divergence between the two trajectories.

MOSSE tracker – Minimum Output Sum of Squared Error algorithm is based on the calculation of adaptive correlations in Fourier space. Created filters minimize the sum of squared errors between the actual correlation output and the predicted correlation output.

CSRT Tracker – The Discriminative Correlation Filter with Channel and Spatial Reliability algorithm uses a spatial reliability map to adjust filter support for a portion of a selected region from the tracking frame. This provides an ability to increase the search area and improve the tracking of non-rectangular objects.

Next, we experimented with SIFT [4], which is based on keypoint matching. The main idea of this algorithm is to encapsulate information about the frame using important key points (Figure 1), and then find the transformation that will match the group of important key points from frame to frame.

SIFT – This algorithm consists of 4 main parts:

- scale-space filtering based on Difference of Gaussian (DoG) to search local extrema over scale and space
- keypoints localization using Taylor series expansion of scale-space to adjust the location of extrema
- orientation assignment to each keypoint based on image gradient to provide rotation-invariance
- new descriptor generation for keypoints using magnitude and orientation of the gradient of image

Figure 1. Detected keypoints by SIFT

Deep learning-based object trackers – 3 methods, described below, are based on neural networks.

Siamese network-based architectures are amongst the most popular object tracking methods.

They consist of two joint neural networks with identical weights. First neural network takes as input the reference image and creates reference

vector embedding. Second neural network takes as input the search image (for ex. next frame), cuts it into pieces and performs the same operation creating search embedding. Finally, the closest matches are found and box coordinates in the search image extracted.

Figure 2 shows the SiamFC [6] architecture. It uses fully connected convolutional layers to compute embedding. For the reference image, it produces one embedding. In the final step cross correlation operator (* in Figure 2) is used to find the closest region in the search image.

SiamTPN [7] (Siamese Transformer Pyramid Networks) structure is shown in Figure 3.

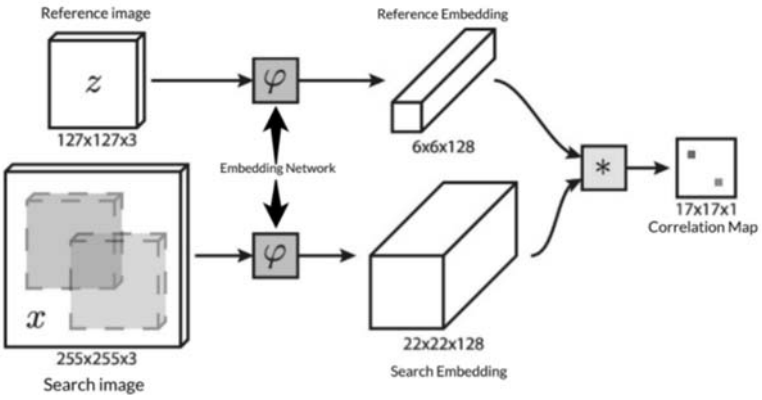


Figure 2. SiamFC Model

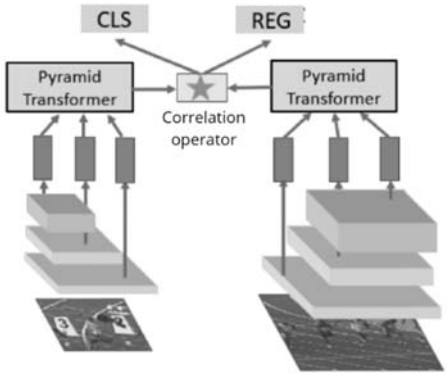


Figure 3. SiamTPN Model.

In contrast with SiamFC, it takes 3 embedding from reference and search images, which allows it to feed them into Pyramid Transformers and get a better representation of images. After that, it counts correlation between those two representations from the pyramid transformer, and passes output to Classifier and Regressor. Classifier predicts which box matches to our reference image most and returns center's coordinates, width and height (x, y, w, h). Regressor tries to tune that coordinates and sizes to include a reference image inside it more accurately.

Finally, let's take a look at TransT [8] architecture (Figure 4).

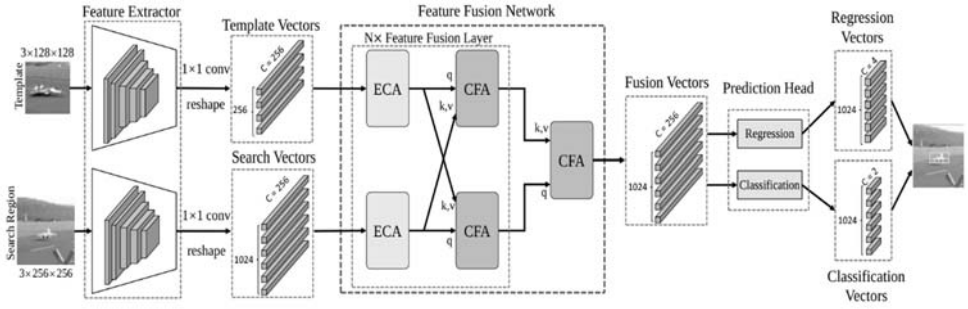


Figure 4. TransT Model.

As before, network inputs are Reference and Search images. At the first step, features from those two images are extracted and converted into vectors using convolution. In the Ego-Context Augment (ECA) block, the model tries to pay attention to semantics inside itself and find regions of interest (ROI) using a multi-head self-attention mechanism. Cross-Feature Augment (CFA) block takes from search image feature map and reference image feature map and combines them using multi-head cross-attention mechanism to see which region of first is correlated most with the second. In the end of this feature fusion block, Fusion vectors are obtained and used to classify and regress the final prediction box.

Siam-TPN and Trans-T both rely on transformer and attention mechanisms. To get more detailed information about these topics, the readers are encouraged to take a look at the following articles.

Results

All of the mentioned algorithms were implemented in a number of short video sequences taken from the drone [9]. In table 1 are the specifications of the drone.

Table 1.

Testing drone information

drone frame, motors, ESC, and propellers	Holybro S500 kit	**telemetry**	SiK Telemetry Radio v3
Power module	PM06 v1	**remote controller**	FrSky Taranis X9D Plus (ACCESS) with SIYI FM30
flight controller	Pixhawk 4	**receiver**	FR receiver
GPS module	Holybro M9N GPS	**camera**	RunCam2 4K edition.

At the start of the video, identical initial box coordinates were given to the trackers as input. Predictions were collected and merged to form a single video file with predicted boxes from each tracker marked with different colors. Figure 5 shows one photo from such a sequence.

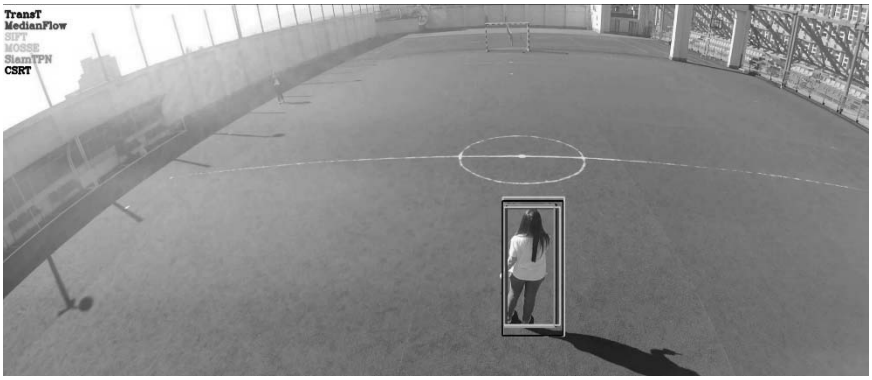


Figure 5. Comparing different methods.

Table 2.

Numerical results of Single Object Tracking for 3 video sequences (IoU metric)

Algorithms	Experiment 1	Experiment 2	Experiment 3
CSRT	80.60%	0.19%	17.74%
MOSSE	8.96%	0.04%	0.05%
MedianFlow	9.21%	0.03%	1.27%
SIFT	4.06%	0.01%	0.15%
SiamTPN	94.60%	85.43%	2.45%
TransT	99.75%	95.89%	84.24%

Conclusion

In this paper, we've analyzed several single object tracking algorithms. An end-to-end, general pipeline was built to initialize and track the object in the real-time video feed. The implementation of classical trackers was based on OpenCV, while the original papers and GitHub repositories were used to build deep learning trackers. All algorithms were tested on videos captured from a flying drone.

Here is an incomplete list of drawbacks of OpenCV-based trackers and SIFT: low speed, inability to stop tracking when the object is lost, inability to continue tracking after the object appears again, tracking similar objects instead of selected one, object loss at high speed of its movement. SiamFC worked better than traditional trackers. Here the problems begin when a tracked object goes behind another object and comes back. SiamTPN and TransT were the best trackers mentioned in the paper. These trackers were robust, and invariant to scale and shape changes. They showed the ability to find the object after losing it from the image. Although, there is still a lot of room for improvement, as can be seen from the videos and numerical results.

REFERENCES

1. MEDIANFLOW: *Zdenek Kalal, Krystian Mikolajczyk, Jiri Matas,* Forward-Backward Error: Automatic Detection of Tracking Failures 2010 - International Conference on Pattern Recognition, 2010.
2. MOSSE Tracker: *David S. Bolme J. Ross Beveridge Bruce A. Draper Yui Man Lui,* Visual Object Tracking using Adaptive Correlation Filters 2010 – CVPR, 2010.
3. CSRT Tracker: *Alan Lukezic, Tomas Vojır, Luka Cehovin Zajc, Jirı Matas and Matej Kristan,* Discriminative Correlation Filter Tracker with Channel and Spatial Reliability 2019 –international Journal of Computer Vision, 126(7), 671–688, 2018.
4. SIFT: *Lowe, David G,* Distinctive image features from scale-invariant key points – International Journal of Computer Vision 60. PP. 91–110, 2014.
5. OpenCV Object Tracking Implementations: https://pyimagesearch.com/2018/07/30/opencv-object-tracking/
6. SIAMFC: *Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, Philip H.S. Torr.* Fully-Convolutional Siamese Networks for Object Tracking 2021, ECCV, 2016.
7. SIAMTPN: *Daitao Xing, Nikolaos Evangeliou, Athanasios Tsoukalas, Anthony Tzes,* Siamese Transformer Pyramid Networks for Real-Time UAV Tracking 2021 – WACV, 2022.
8. TransT: *Xin Chen1, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang and Huczhuan Lu* Transformer Tracking 2021 – CVPR, 2021.

9. https://drive.google.com/drive/folders/1m1QyXsZeD2-ADa782Fym2x96C8O3oshl?usp=sharing

СРАВНЕНИЕ АЛГОРИТМОВ СЛЕЖЕНИЯ ЗА ОБЪЕКТОМ НА ВИДЕОПОСЛЕДОВАТЕЛЬНОСТИ ПОЛУЧЕННЫХ ИЗ БПЛА

В.Г. Мелконян, В.Р. Саакян, Л.А. Киракосян, О.А. Оганесян

АННОТАЦИЯ

В данной статье анализируется задача слежения за одним объектом. Был рассмотрен ряд классических алгоритмов, таких как трекер MEDIAFLOW, трекер MOSSE, трекер CSRT, SIFT. А также алгоритмы на основе глубокого обучения, такие как SiamFC, SiamTPN и TransT. Эти алгоритмы были реализованы на БПЛА. Были проведены эксперименты в реальном мире, чтобы выявить их сильные и слабые стороны.

Ключевые слова: слежение за объектами, компьютерное зрение, беспилотные летательные аппараты, нейронные сети.

МЕХАНИКА

DOI 10.48200/1829-0450_pmn_2022_2_76
УДК 539.376

Поступила: 08.12.2022г.
Сдана на рецензию: 08.12.2022г.
Подписана к печати: 12.12.2022г.

ГИСТЕРЕЗИС ПРИ ЗНАКОПЕРЕМЕННОМ ПЕРИОДИЧЕСКОМ НАГРУЖЕНИИ ГРУНТОВ

Т.Л. Петросян[1], С.Н. Хачатрян[2]

[1]Институт механики НАН РА
[2]Национальный универсситет Архитектуры и Строительства Армении

tlpetrosyan@mail.ru, sargis.xachatryann@mail.ru

АННОТАЦИЯ

В данной статье исследован процесс образования замкнутой петли гистерезиса грунтов в условиях знакопеременных периодических изменений напряжения. Исследование осуществлялось на базе наследственной теории ползучести, при экспоненциальном ядре. Получены уравнение петли гистерезиса и выражение для вычисления остаточной деформации зависимо от номера цикла. Получены изолинии нулевой остаточной деформации при знакопеременном периодическом нагружении.

Ключевые слова: гистерезис, диссипация, ползучесть, периодическое нагружение.

Известно, что в течение эксплуатации здания и сооружения, грунтовые материалы их оснований подвергаются знакопеременным периодическим нагружениям в результате приложения динамических напряжений – таких, как землетрясения, взрывы, ветровые нагрузки или вибрации машин. Исследование и анализ с целью оценки реакции зданий и сооружений на приложение вышеуказанных динамических нагружений находит все возрастающую необходимость в практике исследователей, проектировщиков и строительных инженеров [1]. Для описания грунтовых отложений и их реакций могут быть использованы различные идеализированные модели и аналитическая техника [2, 3, 4, 5]. Но, независимо от выбора этих моделей и техники, сначала необходимо определить соответствующее напряженно-деформированное состояние и поглощающие свойства материалов оснований. Для этих целей могут быть использованы различные лабораторные исследования. Соответствующий выбор в каком-либо данном случае зависит главным образом от диапазона деформаций. Многочисленные экспериментальные исследования показывают, что при низком уровне деформаций грунты обладают высоким модулем сдвига и низким затуханием. При высоком уровне деформаций нелинейность поведения грунта проявляется более сильно, приводя к более низким величинам модуля сдвига, но и к более высоким значениям затухания. Так как поведение грунтов зависит от величины их деформации, удобно представить различные способы исследований в соответствии с амплитудой относительной сдвиговой деформации. В случаях, когда сдвиговая деформация очень мала, для определения соответствующих динамических свойств грунтов могут быть использованы такие лабораторные методы исследования, как резонансный метод испытаний образцов грунта и метод импульсного прозвучивания [5, 7]. Оба методы (резонансный и прозвучивания) широко используются для определения скорости распространении волны напряжения, коэффициента затухания и модуля упругости при очень малых деформациях. Для изучения динамических свойств грунтов при более высоком уровне сдвиговых деформаций используется другие методы [4].

Во многих случаях сдвиговые деформации грунтов при землетрясении возбуждаются благодаря рраспространению сдвиговых волн от нижележащих слоев. В течение землетрясения как циклические сдвиговые напряжения, действующие на нижнюю и верхнюю грани, так и элемент грунта изменяют несколько раз свое направление [9]. Близкое лабораторное воспроизведение в этих условиях осуществляется в опытах как простой сдвиг [10]. Исследования, изложенные в работе [11], были проведены с целью получения данных по динамическим свойствам грунтов (модуль сдвига, гистерезисное затухание) при достаточно высоких амплитудах деформаций, которые могут встретиться на практике. В данной работе были проведены испытания грунтов на простой сдвиг с амплитудами в пределах 0.01– 0.5 процентами при 2 см высоты образцов. Частота воздействия была выбрана равной 1 гц. Согласно соответствующей величине напряжений и деформаций, построены графики зависимости $\tau - \gamma$ для конкретных циклов, образующих гистерезисную петлю. На Рис. 1 приведены изображения этих кривых, построенные для 1, 2, 10 и 300 циклов.

В работе [12] приведены экспериментальные данные о процессе образования замкнутой петли гистерезиса, полученные на глинистых грунтах при многократных ступенчатых приложениях компрессионного напряжения, чередующегося с полной разгрузкой. Получено, что линейная теория наследственности ползучести с экспоненциальным ядром описывает этот процесс достаточно точно. Приведены теоретические данные о гистерезисе при различных номерах цикла – в зависимости от периода нагружения.

Целью настоящей работы является исследование процесса образования замкнутой петли гистерезиса грунтов в условиях знакопеременных периодических изменений напряжения, его описания с использованием наследственной теорий ползучести в сравнении с экспериментальными данными, полученными в работе [11].

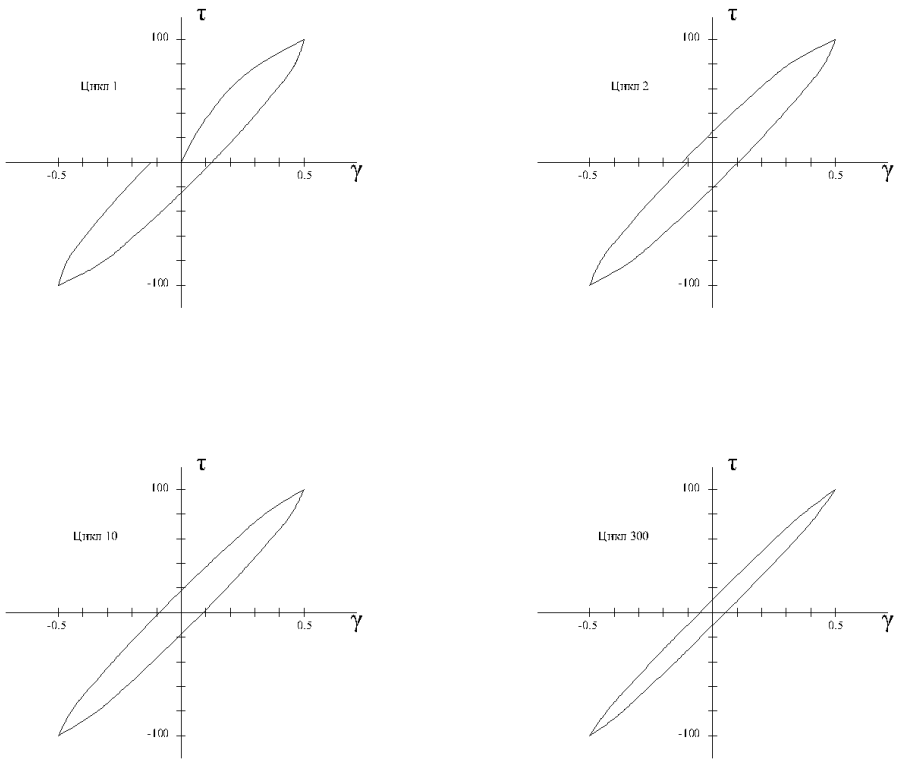


Рис. 1. Экспериментальные кривые $\tau - \gamma$ (петли гистерезиса), полученные в работе [7].

Обозначим через A «остаточную» деформацию A_nA_{n+1} (Рис. 2) остающуюся после соответствующего цикла нагружения, где n – целое число периодов T-нагружения. Подсчитывая A как функцию T и n, изучение процесса образования замкнутой петли гистерезиса приводит к получению выражения для A (T,n) при использовании наследственной теории ползучести.

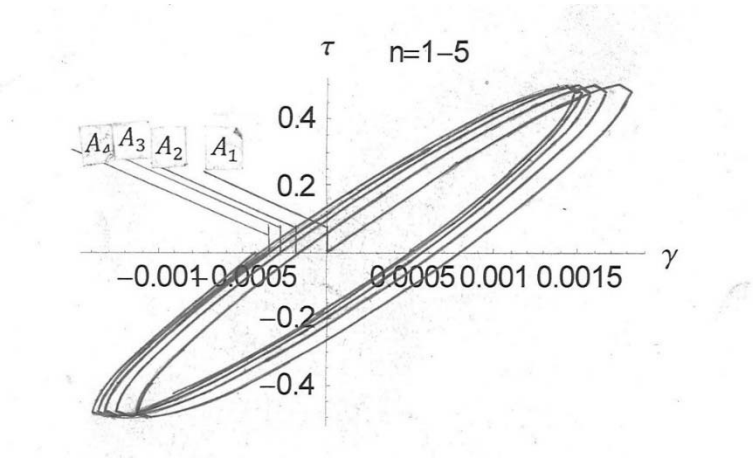


Рис. 2. Образование замкнутой петли гистерезиса при знакопеременном симметричном нагружении.

Для описания деформации при переменных касательных напряжениях $\tau(t)$, согласно линейной теории наследственности, будем иметь [13]

$$\gamma(t) = \frac{\tau(t)}{G} + \int_0^t \tau(t_1) \frac{\partial \delta(t-t_1)}{\partial t_1} dt_1 \qquad (1)$$

Допустим, что аппроксимация ползучести материала при единичном напряжении (мера ползучести) определяется формулой [14]

$$\delta(t) = \delta_0(1 - e^{-\alpha t}) \qquad (2),$$

где δ_0 и α – постоянные, определяемые из опыта.

Предположим, что изменение напряжения $\tau(t)$ имеет место по следующему закону:

$$\tau(t) = \tau_0[\sin(\omega t + \varphi_0) + \lambda] \qquad (3),$$

где $-\omega$-циклическая частота, φ_0-начальная фаза, λ – постоянная определяющая степень асимметрии циклического нагружения, которая изменяется в пределах $0 \le \lambda < 1$.

При использовании линейной теории наследственности в применении к (2) и (3) имеем:

$$\gamma(t) = \tau_0 \left\{ \frac{\sin(\omega t + \varphi_0) + \lambda}{G} + \frac{\delta_0 \alpha}{\alpha^2 + \omega^2} [\alpha \sin(\omega t + \varphi_0) - \omega \cos(\omega t + \varphi_0) + \omega \cos \varphi_0 e^{-\alpha t} - \alpha \sin \varphi_0 e^{-\alpha t}] + \lambda \delta_0 (1 - e^{-\alpha t}) \right\} \quad (4)$$

В формуле (4), подставляя $t = \mathrm{T}n$, получим значения деформаций $\gamma(t)$ в точках A_1, A_2, A_3 ... ... (при n=0,1,2,3……..)

$$\gamma(t) = \frac{\tau_0(\sin \varphi_0 + \lambda)}{G} + \frac{\delta_0 \alpha \tau_0}{\alpha^2 + \omega^2} (\alpha \sin \varphi_0 - \omega \cos \varphi_0 + \omega \cos \varphi_0 e^{-\alpha T n} - \alpha \sin \varphi_0 e^{-\alpha T n}) + \delta_0 \lambda \tau_0 (1 - e^{-\alpha T n}) \quad (5)$$

При этом функция A(T,n) определяется так:

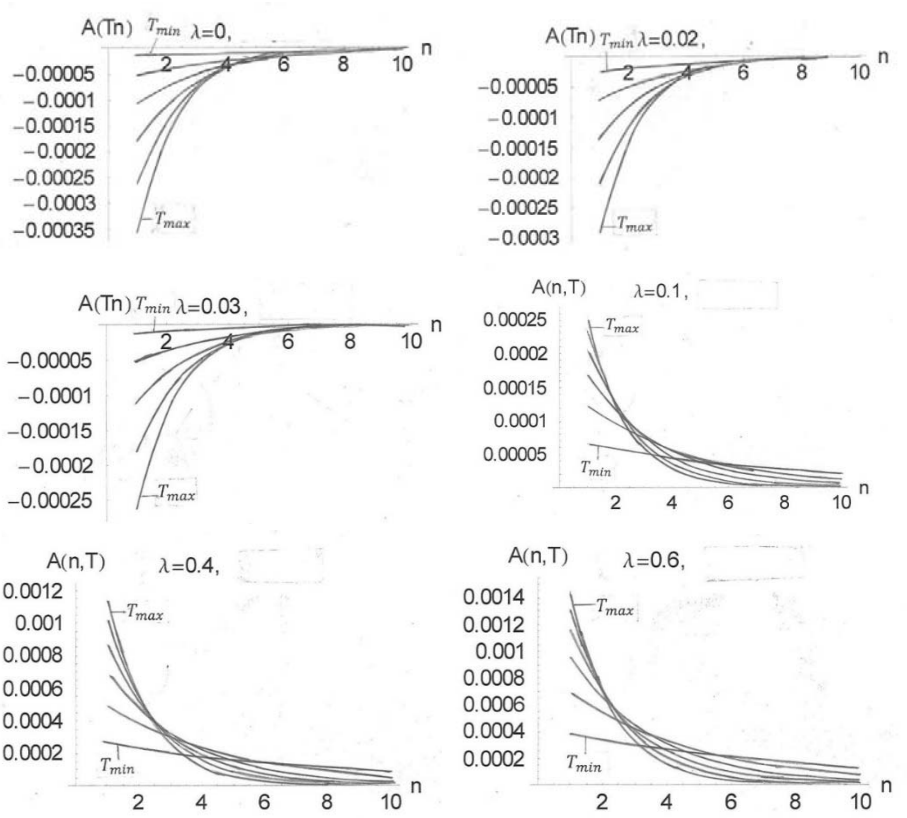
$$A(T,n) = \gamma(Tn) - \gamma\big(T(n-1)\big) = \delta_0 \tau_0 \left[\lambda - \left(\frac{\alpha\omega \cos \varphi_0}{\alpha^2+\omega^2} - \frac{\alpha^2 \sin \varphi_0}{\alpha^2+\omega^2}\right)\right] e^{-\alpha T(n-1)} (1 - e^{-\alpha T n}) \quad (6)$$

Параметры ползучести δ_0 и α, использованные в формуле (6) при расчете значения функции A (T,n), взяты из результатов исследо-

ваний, представленные в работе [10], которые получены при испытании на ползучесть глинистых грунтов на приборах кручения сплошных образцов. Указанные параметры имели следующие численные значения: $\delta_0 = 0.0112$, $\alpha = 0.126\ c^{-1}$.

Ниже рассматриваются графические представления и анализ зависимости остаточной деформации A от ряда характеристик периодического изменения напряжения, согласно теории наследственности.

На Рис. 3 приведены теоретические кривые A(n), построенные по формуле (6) для нескольких значений периода T при постоянном амплитуде τ_0. Ясно, что при такой программе нагружения, значение скорости изменения напряжения уменьшается с увеличением периода T.



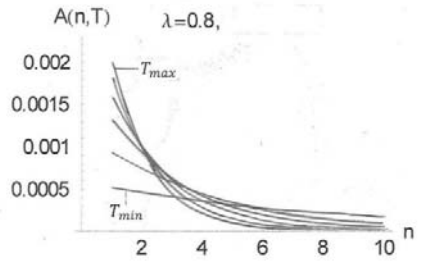
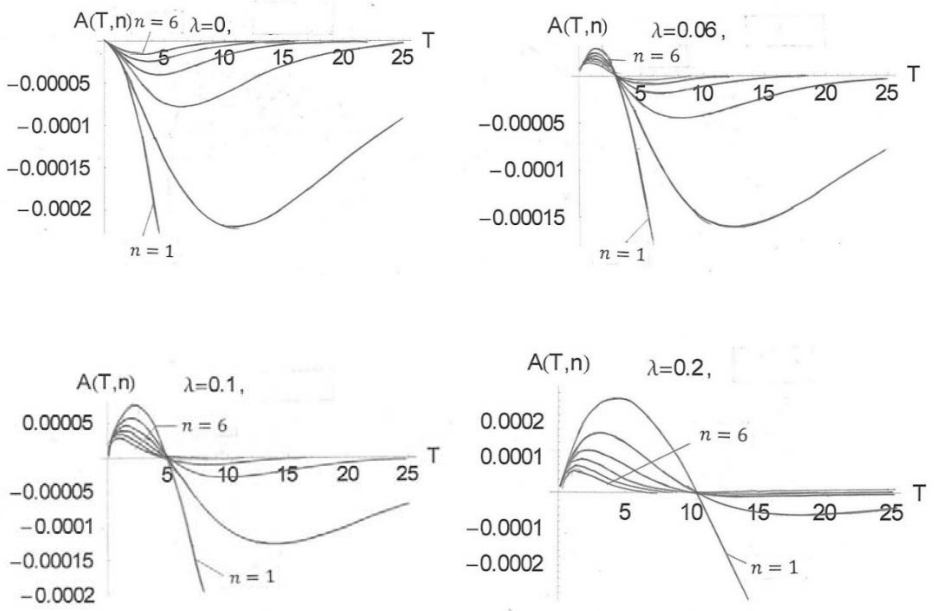


Рис. 3. Теоретические кривые A(n), построенные по формуле (6) для нескольких значений периода T при постоянном амплитуде τ_0 *(*λ *– степень асимметрии цикла).*

На Рис. 4 представлены теоретические кривые A(T), построенные по формуле (6) для нескольких циклов n при постоянном амплитуде τ_0.



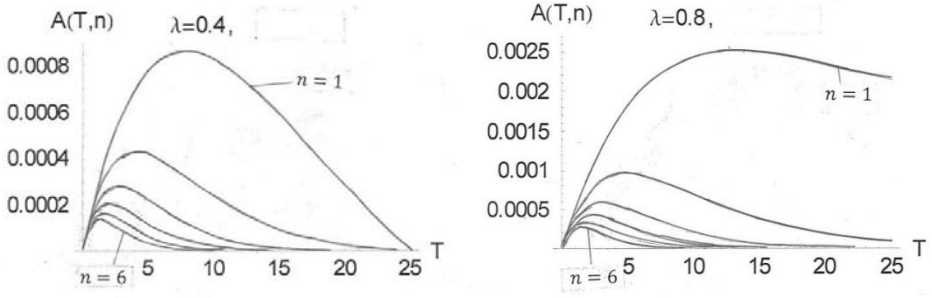


Рис. 4. Теоретические кривые A(T), построенные по формуле (6) для нескольких циклов n при постоянном амплитуде τ_0 (λ – степень асимметрии цикла).

Как видно из Рис. 3 и Рис. 4, остаточные деформации после каждого цикла в процессе образования замкнутой петли гистерезиса очень зависят от периода нагружения, а при наличии асимметрии нагружения может меняться знак остаточной деформации.

Очевидно, что при условии нулевой остаточной деформации из (6) можно получить соотношение между периодом T и степенью асимметрий λ. С помощью вышеуказанного полученного соотношения можно построить изолинии остаточной деформации для разных грунтов (Рис. 5).

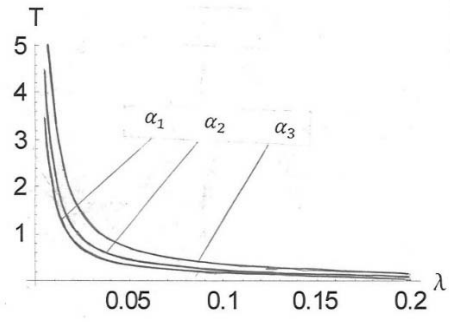


Рис. 5. Изолинии нулевой остаточной деформации при знакопеременном периодическом нагружении ($\alpha_1, \alpha_2, \alpha_3$ параметры ползучести разных грунтов, определяемые из опыта).

ЛИТЕРАТУРА

1. *Баркан Д.Д.* Виброметод в строительстве, М. Госстройиздат, 1959, 313с.
2. *Xianzhang Ling, Peng Li, Feng Zhang, Yingying Zhao, Yan Li, Lingshi An* Permanent Deformation Characteristics of Coarse Grained Subgrade Soils under Train-Induced Repeated Load, Advances in Materials Science and Engineering, vol. 2017, Article ID 6241479, 15 pages, 2017. https://doi.org/10.1155/2017/6241479
3. *Arefi M.J., Cubrinovski M., Bradley B.A.* (2012). A model for nonlinear total stress analysis with consistent stiffness and damping variation. Lisbon, Portugal: 15th World Congress on Earthquake Engineering (15WCEE), 24–28 Sep. 2012.
4. *Nogami Y., Murono Y. and Morikawa H.* (2012), Nonlinear hysteresis model taking into account s-shaped hysteresis loop and its standard parameters, Proceedings of 15th WCEE, Lisbon, Portugal, September.
5. *Zhang Y., Knoph K. Aamodt, Amir M.* Kaynia Hysteretic damping model for laterally loaded piles, Marine Structures,Volume 76,2021,102896, ISSN 0951-8339: https://doi.org/10.1016/j.marstruc.2020.102896.
6. *Красников Н.Д.* Динамические свойства грунтов и методы их определения, Ленинград: Изд-во «Стройиздат», 1979, 238с.
7. *Сабылин М.И., Мойсейчик Е.К.* Исследование уникальных свойств грунтов ультразвуковым методом // Изв. ВУЗ-ов: Строительство и архитектура. Ер.: Изд-во «Айастан», 1992, 296, № 9. СС. 135–139.
8. *Сорокин Е.С.* Методы экспериментального определения внутреннего трения в твердых телах // в кн. «Вопросы прикладной механики», 1964, вып. 193, СС. 5–42.
9. *Ишихара К.* Поведение грунтов при землетресениях (пер с англ.), СПб, 2006, 384с.
10. *Месчян С.Р.* Реологические процессы в глинистых грунтах (с учетом особых воздействий), Ер.: «Айастан», 1992, 395с., ISBN: 5-540-00826-х
11. *Silver, Marshallh and Seed M.* Boltom Deformation Characteristics of Sands Under Cyclic Loading, Journal of the Soil Mechanics and Foundatious Division, ASCE Vol 97, no SN 8 ProcPoper 8334 August 1971. PP. 1081–1098.
12. *Петросян Т.Л.* Процесс образования замкнутой петли гистерезиса материалов // сб. Научных трудов международной конференции Актуальные проблемы механики сплошний среды, 21–26 сентябрь 2015г. Цахкадзор, Армения. СС. 334–338.

13. *Петросян Т.Л., Симонян А.М.* Исследование гистерезиса при малоцикловой ползучести // Изв. НАН Армении, Механика, 2007, Т. 60, № 2. СС. 114–121.
14. *Гарофало Ф.* Законы ползучести и длительной прочности металлов. М. 1968, 304с.

HYSTERESIS UNDER SIGN-ALTERNATE PERIODIC LOADING OF SOILS

T. Petrosyan, S. Khachatryan

Institute of Mechanics of NAS RA
National University of Architecture and Construction of Armenia

ABSTRACT

The process of formation of a closed hysteresis loop of soils under conditions of alternating periodic voltage changes was studied in this work. The study was carried out on the basis of the hereditary theory of creep, with an exponential kernel. An equation for the hysteresis loop and an expression for calculating the residual strain depending on the cycle number were obtained. Isolines of zero residual strain under alternating periodic loading were obtained.

Keywords: hysteresis, dissipation, creep, periodic loading.

СВЕДЕНИЯ ОБ АВТОРАХ

Галстян Т. аспирант третьего года обучения по направленности «Математическое моделирование,численные методы и комплексы программ» РАУ, научный сотрудник YerevaNN Research Lab

Драмбян А. магистрант второго курса по направлению подготовки «Прикладная математика и информатика РАУ

Киракосян Л. магистрант второго курса по направлению подготовки «Прикладная математика и информатика» (ОП «Интеллектуальные системы и робототехника») РАУ, научный сотрудник лаборатории CAST

Маргарян Ж.Г. к.ф.-м.н., доцент, доцент кафедры дискретной математики и теоретической информатики ЕГУ

Мелконян В. магистрант второго курса по направлению подготовки «Прикладная математика и информатика» (ОП «Интеллектуальные системы и робототехника») РАУ, научный сотрудник лаборатории CAST

Микаелян Г. магистрант второго курса факультета информатики и прикладной математики ЕГУ, программный инженер компании «Кодсигнал»

Оганесян О. магистрант второго курса по направлению подготовки «Прикладная математика и информатика» (ОП «Интеллектуальные системы и робототехника») РАУ, научный сотрудник лаборатории CAST

Петросян Т. магистрант второго курса по направлению подготовки «Прикладная математика и информатика РАУ

Саакян В. магистрант второго курса по направлению подготовки «Прикладная математика и информатика» (ОП «Интеллектуальные системы и робототехника») РАУ, научный сотрудник лаборатории CAST

Саакян О. магистрант второго курса факультета информатики и прикладной математики ЕГУ

Ханданян Н.А. магистрант второго курса факультета математики и механики ЕГУ

Петросян Т.Л. к. тех. н., научный сотрудник Институт механики НАН РА

Хачатрян С.Н. магистрант Национального университета архитектуры и строительства Армении

Главный редактор РНИ *– М.Э. Авакян*
Корректор *– М.Р. Асланян*
Компьютерная верстка *– А.Г. Антонян*

Editor-in-Chief of the RNI *– M.E. Avakyan*
Proofreader *– M.R. Aslanyan*
Computer layout *– A.G. Antonyan*

Адрес Редакции научных изданий
Российско-Армянского университета:
0051, г. Ереван, ул. Овсепа Эмина, 123
тел./факс: (+374 12) 77-57-75 (внутр. 8130)
e-mail maria.avakian@rau.am

Address of the Editorial Board of Scientific Publications
of the Russian-Armenian University*:*
0051, Yerevan, st. Hovsep Emin, 123
tel./fax: (+374 12) 77-57-75 (ext. 8130)
e-mail: maria.avakian@rau.am

Заказ № 1
Подписано к печати 22.12.2022г.
Формат 70x100$^{1}/_{16}$. Бумага офсетная № 1.
Объем 5.5 усл. п.л. Тираж 100 экз.

Order No. 1
Signed for publication on December 22, 2022
Format 70x100$^{1}/_{16}$. Offset paper No. 1.
Volume 5.5 conv. p.l. Circulation 100 copies