Computer Science Journal of Moldova, vol.**26**, no.2(77), 2018

Efficiency and Penalty Factors on Monoids of Strings

Mitrofan Choban, Ivan Budanaev

Abstract

In information theory, linguistics and computer science, metrics for measuring similarity between two given strings (sequences) are important. In this article we introduce efficiency, measure of similarity and penalty for given parallel decompositions of two strings. Relations between these characteristics are established. In this way, we continue the research from [3], [4].

Keywords: invariant distance, measure of similarity, Levenshtein distance, Hamming distance, Graev method, penalty.

1 Introduction

Let G be a semigroup and d be a metric on G. The metric d is called:

- *left* (respectively, *right*) *invariant* if $d(xa, xb) \leq d(a, b)$ (respectively, $d(ax, bx) \leq d(a, b)$) for all $x, a, b \in G$;
- *invariant* if it is both left and right invariant;
- *strong invariant* if $d(xa, xb) = d(ax, bx) = d(a, b)$ for all $x, a, b \in G$;
- *stable* if $d(xy, uv) \leq d(x, u) + d(y, v)$ for all $x, y, u, v \in G$.

Example 1.1. *Let G be the additive semigroup of non-negative real numbers and G^+ be the subsemigroup of positive real numbers. We put $d(x, x) = 0$ for each $x \in G$, $d(x, y) = 1$ for any distinct numbers $x, y \in G^+$ and $d(0, x) = d(x, 0) = 2$ for any $x \in G^+$. Then d is an invariant metric on G. Since $1 = d(2, 5) = d(0 + 2, 3 + 2) < d(0, 3) = 2$, the metric d is not strong invariant.*

The following assertion is well known.

Proposition 1. *Let d be an invariant metric on a group G. Then the metric d is strong invariant and $d(x^{-1}, y^{-1}) = d(x, y)$ for all $x, y \in G$.*

A monoid is a semigroup with an identity element. Fix a non-empty set A. The set A is called an alphabet. We put $\bar{A} = A \cup \{\varepsilon\}$. Let $L^*(A)$ be the set of all finite strings $a_1a_2 \dots a_n$ with $a_1, a_2, \dots, a_n \in \bar{A}$. Let ε be the empty string. Consider the strings $a_1a_2 \dots a_n$ for which $a_i = \varepsilon$ for some $i \leq n$. If $a_i \neq \varepsilon$, for any $i \leq n$ or $n = 1$ and $a_1 = \varepsilon$, the string $a_1a_2 \dots a_n$ is called an *irreducible string* or *canonical string*. The set $Supp(a_1a_2 \dots a_n) = \{a_1, a_2, \dots, a_n\} \cap A$ is the support of the string $a_1a_2 \dots a_n$ and $l(a_1a_2 \dots a_n) = |\{i \leq n : a_i \neq \varepsilon\}|$ is the length of the string $a_1a_2 \dots a_n$. For two strings $a_1 \dots a_n$ and $b_1 \dots b_m$, their product (concatenation) is $a_1 \dots a_n b_1 \dots b_m$. If $n \geq 2$, $i < n$ and $a_i = \varepsilon$, then the strings $a_1 \dots a_n$ and $a_1 \dots a_{i-1}a_{i+1} \dots a_n$ are considered equivalent. In this case any string is equivalent to one unique canonical string. We identify the equivalent strings. The set $L(A)$ of all canonical strings is the family of all classes of equivalent strings. In this case $L^*(A)$ is a semigroup and $L(A)$ becomes a monoid with identity ε. The set $L(A)$ is not a subsemigroup of $L^*(A)$. Only the set $L(A) \setminus \{\varepsilon\}$ is a subsemigroup of the semigroup $L^*(A)$.

Let $Supp(a, b) = Supp(a) \cup Supp(b) \cup \{\varepsilon\}$, and $Supp(a, a) = Supp(a) \cup \{\varepsilon\}$. It is well known that any subset $L \subset L(A)$ is an abstract language over the alphabet A.

Let a, b be two strings. For any two representations $a = a_1a_2 \dots a_n$ and $b = b_1b_2 \dots b_m$ we put

$$\begin{aligned} d_H(a_1a_2 \dots a_n, b_1b_2 \dots b_m) =& |\{i : a_i \neq b_i, i \leq min\{n, m\}\}| \\ &+ |\{i : n < i \leq m, b_i \neq e_i\}| \\ &+ |\{j : m < j \leq n, a_j \neq e_i\}|. \end{aligned}$$

The function d_H is called the Hamming distance on the space of strings [3], [4], [7].

Now we put:

$$d_G(a, b) = inf\{d_H(a, b) : a = a_1a_2 \dots a_n, b = b_1b_2 \dots b_n\}.$$

The function d_G is called the Graev – Markov distance on the space of strings [6], [9].

The V. I. Levenshtein's distance $d_L(a,b)$ between two strings $a = a_1a_2\dots a_n$ and $b = b_1b_2\dots b_m$ is defined as the minimum number of insertions, deletions, and substitutions required to transform one string into the other [4], [8].

We put $A^{-1} = \{a^{-1} : a \in A\}$, $\varepsilon^{-1} = \varepsilon$, $(a^{-1})^{-1} = a$ for any $a \in A$ and consider that $A^{-1} \cap \bar{A} = \emptyset$. Denote $\check{A} = A \cup A^{-1} \cup \{\varepsilon\}$. Let $\check{L}(A) = L(\check{A})$ be the set of all strings over the set $\check{A}$. The strings over the set $\check{A}$ are called *words*. A word $a = a_1a_2\dots a_n \in \check{L}(A)$ is called an irreducible string if either $n = 1$ and $a_1 \in \check{A}$, or $n \geq 2$, $a_i \neq \varepsilon$ for any $i \leq n$ and $a_j^{-1} \neq a_{j+1}$ for each $j < n$.

Let $a = a_1a_2\dots a_n \in \check{L}(A)$ and $n \geq 2$. Then:

- if $i \leq n$ and $a_i = \varepsilon$, then the words $a_1\dots a_n$ and $a_1\dots a_{i-1}a_{i+1}\dots a_n$ are considered equivalent;

- if $i < n$ and $a_i^{-1} = a_{i+1}$, then the words $a_1a_2\dots a_n$ and $a_1\dots a_{i-1}\varepsilon a_{i+2}\dots a_n$ are considered equivalent.

In this case, any word $a_1a_2\dots a_n \in \check{L}(A)$ is equivalent to one unique irreducible word from $\check{L}(A)$. We identify equivalent words. Classes of equivalence form free group $F(A)$ over A with unity ε. We have that $L(A)$ is a subsemigroup of the group $F(A)$.

Let $a = a_1a_2\dots a_n \in F(A)$ be an irreducible word. The representation $a = x_1x_2\dots x_m \in L^*(A)$ is called an *almost irreducible* representation of a if there exist $1 \leq i_1 < i_2 < ... < i_n \leq m$ such that $a_j = x_{i_j}$ for any $j \leq n$ and $x_i = \varepsilon$ for each $i \in \{1,2,...,m\} \setminus \{i_1, i_2, ..., i_n\}$. If $a = a_1a_2\dots a_n \in L^*(A)$ is a representation of the string a, then $a_1a_2\dots a_n$ is an almost irreducible word.

If $a = a_1a_2\dots a_n$, then $a^s = a_na_{n-1}\dots a_1$ and $a^{-1} = a_n^{-1}a_{n-1}^{-1}\dots a_1^{-1}$. The word a^s is the symmetric word of a and a^{-1} is the inverse word of a. If a and b are equivalent words, then the words a^{-1} and b^{-1} are equivalent, as well as the words a^s and b^s.

Hence the mappings $\cdot^s, \cdot^{-1} : F(A) \longrightarrow F(A)$ are the group automorphisms. Obviously that $L(A)^s = L(A)$.

Let $a, b \in A$ and $a \neq b$, then we put $d_H(a,b) = d_H(a^{-1},b^{-1}) = d_H(a,\varepsilon) = d_H(\varepsilon,a) = d_H(a^{-1},\varepsilon) = d_H(\varepsilon,a^{-1}) = 1$. If $a \in A$ and $b \in$

A^{-1}, then $d_H(a,b) = d_H(b,a) = 2$. For any $x \in \check{A}$ we put $d_H(x,x) = 0$. Thus d_H is a metric on $\check{A}$. For any two words $a_1a_2\dots a_n, b_1b_2\dots b_m \in \check{L}(A)$ we put:

$$\begin{aligned} d_H(a_1a_2\dots a_n, b_1b_2\dots b_m) =& \Sigma\{d_H(a_i,b_i) : i \leq min\{n,m\}\} \\ &+|\{i : n < i \leq m, b_i \neq e_i\}| \\ &+|\{j : m < j \leq n, a_j \neq e_i\}|. \end{aligned}$$

For $a, b \in F(A)$ we put:

$$\check{d}(a,b) = inf\{d_H(a,b) : a = a_1\dots a_n \in \check{L}(A), b = b_1\dots b_n \in \check{L}(A)\}.$$

Remark 1.1. *The function $\check{d}$ is called the Graev – Markov distance on the free group [6]. The method of extensions of distances for free groups, used by us, was proposed by A. A. Markov [9] and M. I. Graev [6]. For metrics on free universal algebras it was extended in [2], for quasimetrics on free groups and varieties of groups it was examined in [5], [12].*

M. I. Graev [6] has proved the following assertions:

G1. $\check{d}$ is an invariant metric on $F(A)$ and $\check{d}(a,b) = d_H(a,b)$ for all $a, b \in A^*$.

G2. If ρ is an invariant metric on $F(A)$ and $\rho(x,y) \leq d_H(x,y)$ for any x, y inA^*, then $\rho(x,y) \leq \check{d}(x,y)$ for any $x, y \in F(A)$.

G3. For any two words $a, b \in F(A)$ there exist $m \geq 1$ and two almost irreducible representations $a = x_1x_2\dots x_m$ and $b = y_1y_2\dots y_m$ such that $\check{d}(a,b) = d_H(x_1x_2\dots x_m, y_1y_2\dots y_m)$.

Theorem 1.1. *The distance d_G on a monoid $L(A)$ has the following properties:*

1. d_G is a strong invariant metric on $L(A)$ and $d_G(x,y) = d_G(zx,zy) = d_G(xz,yz)$ for all $x, y, z \in L(A)$.

2. $d_G(a,b) = d_G(a^s, b^s)$ for all $a, b \in L(A)$.

3. If ρ is an invariant metric on $L(A)$ and $\rho(x,y) \leq d_G(x,y)$ for all $x, y \in \bar{A}$, then $\rho(a,b) \leq d_G(a,b)$ for all $a, b \in L(A)$.

4. For any $a, b \in L(A)$ there exist $n \in \mathbb{N}$, $x_1, x_2, \dots, x_n \in Supp(a,a)$ and $y_1, y_2, \dots, y_n \in Supp(b,b)$ such that $a = x_1x_2\dots x_n$,

$b = y_1y_2 \dots y_n$ *such that* $n \leq l(a) + l(b)$ *and* $d_G(a,b) = |\{i : i \leq n, a_i \neq b_i\}| = d_H(x_1x_2 \dots x_n, y_1y_2 \dots y_n)$.

5. $d_G(a,b) = d_L(a,b) = \check{d}(a,b) \leq d_H(a,b)$ *for all* $a, b \in L(A)$.

Proof. Fix $a, b \in L(A)$. Let $a = a_1a_2 \dots a_n$, $b = b_1b_2 \dots b_n$. If $n > l(a) + l(b)$, then there exists $i \leq n$ such that $a_i = b_i = \varepsilon$, $a = a_1a_2 \dots a_{i-1}a_{i+1} \dots a_n$, $b = b_1b_2 \dots b_{i-1}b_{i+1}, \dots b_n$ and $d_H(a_1a_2...a_n, b_1b_2...b_n) = d_H(a_1...a_{i-1}a_{i+1}...a_n,\ b_1...b_{i-1}b_{i+1}...b_n)$. Hence $d_G(a,b) = inf\{d_H(a_1a_2 \dots a_n,\ b_1b_2 \dots b_n) : a = a_1a_2 \dots a_n$, $b = b_1b_2 \dots b_n,\ n \leq l(a) + l(b)\}$. Since we have finite pairs of parallel representations $a = a_1a_2 \dots a_m$, $b = b_1b_2 \dots b_m$ of length $m \leq l(a)+l(b)$, there exist $n \in \mathbb{N}$, $x_1, x_2, \dots, x_n \in Supp(a,a)$ and $y_1, y_2, \dots, y_n \in Supp(b,b)$ such that $a = x_1x_2 \dots x_n$, $b = y_1y_2 \dots y_n$ with $n \leq l(a)+l(b)$ and $d_G(a,b) = |\{i : i \leq n, a_i \neq b_i\}| = d_H(x_1x_2 \dots x_n, y_1y_2 \dots y_n)$. Thus, *Assertion 4* is proved. *Assertion 2* is obvious.

Fix $a, b \in L(A)$ and $c \in A$. It is clear that $d_G(ca, cb) \leq d_G(a,b)$. Assume that $d_G(ca, cb) < d_G(a,b)$. Then there exist representations $ca = x_1x_2 \dots x_n$ and $cb = y_1y_2 \dots y_n$ such that $n \leq l(a) + l(b) + 2$ and $d_G(ca, cb) = d_H(x_1x_2 \dots x_n, y_1y_2 \dots y_n)$, where $A \cap \{x_i, y_i\} \neq \emptyset$ for each $i \leq n$. If $x_1 = y_1$, then $x_1 = y_1 = c$. In this case $a = x_2 \dots x_n$, $b = y_2 \dots y_n$ and $d_G(a,b) \leq d_H(x_2 \dots x_n, y_2 \dots y_n) = d_H(x_1x_2 \dots x_n, y_1y_2 \dots y_n) = d_H(ca, cb) < d_H(a,b)$, a contradiction. Hence $x_1 \neq y_1$. In this case we have two possibilities: $x_1 = c$, $y_1 = \varepsilon$ or $x_1 = \varepsilon$, $y_1 = c$. We can assume that $x_1 = c$ and $y_1 = \varepsilon$. Let $1 < j$, $y_j = c$ and $y_i = \varepsilon$ for each $i < j$. We put $u_1 = v_i = \varepsilon$ for each $i \leq j$, $u_i = x_i$ for each $i \geq 2$ and $v_k = y_k$ for each $k > j$. Then $b = u_1u_2 \dots u_n$, $b = v_1v_2 \dots v_n$, $0 = d_H(u_1, v_1) < d_H(x_1, y_1) = 1$, $d_H(x_j, y_j) \leq 1$, $d_H(u_j, v_j) \leq 1$ and $d_H(u_i, v_i) = d_H(x_i, y_i)$ for $i \in \{2, 3, ..., j-1, j+1, ..., n\}$. Hence $d_G(a,b) \leq d_H(u_1u_2...u_n, v_1v_2...v_n) \leq d_H(x_1x_2...x_n, y_1y_2...y_n) = d_G(ca, cb) < d(a,b)$, a contradiction. Hence $d_G(ca, cb) = d(a,b)$. From *Assertion 2* it follows that $d_G(ac, bc) = d_G(a,b)$. *Assertion 1* is proved.

We put $d(x,x) = 0$ and $d(x,y) = 1$ for any distinct strings $x, y \in L(A)$. Let $ID(A)$ denote the family of all invariant metrics ρ on $L(A)$ with the property: $\rho(x,y) \leq d(x,y)$ for all $x, y \in \bar{(}A)$. Since $d \in ID(A)$, the set $ID(A)$ is non-empty. Now we put $d^*(a,b) = sup\{\rho(a,b) : \rho \in$

$ID(A)\}$. One can easily observe that $d^* \in ID(A)$, $d(a,b) \leq d^*(a,b)$ for any $a, b \in L(A)$ and $d(x,y) = d^*(x,y) = 1$ for all distinct $x, y \in \bar{(}A)$.

Property 1. If $\rho \in ID(A)$, then

$$\begin{aligned}\rho(x_1x_2\dots x_n, y_1y_2\dots y_n) \leq & |\{i \leq 1 : x_i \neq y_i\}| \\ = & d_H(x_1x_2\dots x_n, y_1y_2\dots y_n)\end{aligned}$$

for any two strings $(x_1x_2\dots x_n, y_1y_2\dots y_n) \in L(A)$.

This property follows from the conditions of invariance of metric d.

Property 2. $d_G = d^* = d_L$.

Since d_G and d^* are invariant distances on $L(A)$ and they are constructed with the conditions of extremity

$$\begin{gathered}d^*(a,b) = sup\{\rho(a,b) : \rho \in ID(A)\}, \\ d_G(a,b) = inf\{d_H(a,b) : a = a_1a_2\dots a_n \ , \ b = b_1b_2\dots b_n\},\end{gathered}$$

we have $d_G = d^*$. In [3], [4] it was proved that $d^* = d_L$. The equality $d_G(a,b) = \check{d}(a,b)$ for all $a, b \in L(A)$ follows from the Graev's assertion $G3$ in the above Remark. This completes the proof of the theorem. □

Example 1.2. *The metrics d, $d_G = d_L = d^*$ are strong invariant on $L(A)$. On $L(A)$ there exists a metric $d_r \in ID(A)$ which is invariant, but not strong invariant. Fix a real number r for which $2^{-1} \leq r < 1$. We put $d_r(x,x) = 0$ for each $x \in L(A)$, $d(x,y) = r$ for any distinct strings $x, y \in L(A) \setminus \{\varepsilon\}$ and $d(0,x) = d(x,0) = 1$ for any $x \in L(A) \setminus \{\varepsilon\}$. Then d is an invariant metric on G. Fix $a \in A$. Since $r = d(a,aa) = d(\varepsilon \cdot a, a \cdot a) < d(\varepsilon, a) = 1$, the metric d_r is not strong invariant.*

Remark 1.2. *For the metric d_H we have $d_H(a,b) \leq max\{l(a, l(b)\}$ for any strings $a, b \in L(A)$. The Hamming distance d_H is left invariant: $d_H(xa, xb) = d(a,b)$ for all strings $x, a, b \in L(A)$. Assume now that $x, y, z \in A$, $a = xyzxyz$, $b = yzxy$ and $c = xyz$. Then $d_G(a,b) = 2$ and $6 = d_H(a,b) < d_H(ac,bc) = 9$. Therefore, metric d_H is not right invariant.*

2 Efficiency and Penalty of Two Strings

The longest common substring and pattern matching in two or more strings is a well known class of problems. For any two strings $a, b \in L(A)$ we find the decompositions of the form $a = v_1u_1v_2u_2 \dots v_ku_kv_{k+1}$ and $b = w_1u_1w_2u_2 \dots w_ku_kw_{k+1}$, which can be represented as $a = a_1a_2 \dots a_n$, $b = b_1b_2 \dots b_n$ with the following properties:

- some a_i and b_j may be empty strings, i.e. $a_i = \varepsilon$, $b_j = \varepsilon$;
- if $a_i = \varepsilon$, then $b_i \neq \varepsilon$, and if $b_j = \varepsilon$, then $a_j \neq \varepsilon$;
- if $u_1 = \varepsilon$, then $a = v_1$ and $b = w_1$;
- if $u_1 \neq \varepsilon$, then there exists a sequence $1 \leq i_1 \leq j_1 < i_2 \leq j_2 < \dots < i_k \leq j_k \leq n$ such that:

$u_1 = a_{i_1} \dots a_{j_1} = b_{i_1} \dots b_{j_1}$, $u_2 = a_{i_2} \dots a_{j_2} = b_{i_2} \dots b_{j_2}$, $u_k = a_{i_k} \dots a_{j_k} = b_{i_k} \dots b_{j_k}$;

- if $v_1 = w_1 = \varepsilon$, then $i_1 = 1$;
- if $v_{k+1} = w_{k+1} = \varepsilon$, then $j_k = n$;
- if $k \geq 2$, then for any $i \in \{2, \dots, k\}$ we have $v_i \neq \varepsilon$ or $w_i \neq \varepsilon$.

In this case $l(u_1) + l(u_2) + \dots + l(u_k) = |\{i : a_i = b_i\}|$.

The above decompositions forms are called *parallel decompositions* of strings a and b [3], [4]. For any parallel decompositions $a = v_1u_1 \dots v_ku_kv_{k+1}$ and $b = w_1u_1 \dots w_ku_kw_{k+1}$ the number

$$E(v_1u_1 \dots v_ku_kv_{k+1}, w_1u_1 \dots w_ku_kw_{k+1}) = \sum_{i \leq k+1} \{\max\{l(v_i), l(w_i)\}\} = d_H(x_1x_2 \dots x_n, y_1y_2 \dots y_n)$$

is called the efficiency of the given parallel decompositions. The number $E(a, b)$ is equal to the minimum of efficiency values of all parallel decompositions of the strings a, b and is called the *common efficiency of the strings a,b*. It is obvious that $E(a, b)$ is well determined and $E(a, b) = d_G(a, b)$. We say that the parallel decompositions $a = v_1u_1v_2u_2 \dots v_ku_kv_{k+1}$ and $b = w_1u_1w_2u_2 \dots w_ku_kw_{k+1}$ are optimal if the following equality holds:

$$E(v_1u_1v_2u_2 \dots v_ku_kv_{k+1}, w_1u_1w_2u_2 \dots w_ku_kw_{k+1}) = E(a, b).$$

This type of decompositions are associated with the problem of approximate string matching [10]. If the decompositions $a = v_1u_1 \dots v_ku_kv_{k+1}$ and $b = w_1u_1 \dots w_ku_kw_{k+1}$ are optimal and $k \geq 2$, then we may consider that $u_i \neq \varepsilon$ for any $i \leq k$.

Any parallel decompositions $a = a_1a_2 \dots a_n = v_1u_1v_2u_2 \dots v_ku_kv_{k+1}$ and $b = b_1b_2 \dots b_n = w_1u_1w_2u_2 \dots w_ku_kw_{k+1}$ generate a common subsequence $u_1u_2 \dots u_k$. The number $m(a_1a_2 \dots a_n, b_1b_2 \dots b_n) = l(u_1) + l(u_2) + \dots + l(u_k)$ is the *measure of similarity* of the decompositions [1], [11]. There exist parallel decompositions $a = v_1u_1v_2u_2 \dots v_ku_kv_{k+1}$ and $b = w_1u_1w_2u_2 \dots w_ku_kw_{k+1}$ for which the measure of similarity is maximal. The maximum value of the measure of similarity of all decompositions is denoted by $m^*(a, b)$. The maximum value of the measure of similarity of all optimal decompositions is denoted by $m^\omega(a, b)$. We can note that $m^\omega(a, b) \leq m^*(a, b)$. For any two parallel decompositions $a = a_1a_2 \dots a_n$ and $b = b_1b_2 \dots b_n$ as in [4], we define the *penalty factors* as

$$p_r(a_1a_2 \dots a_n, b_1b_2 \dots b_n) = |\{i \leq n : a_i = \varepsilon\}|,$$

$$p_l(a_1a_2 \dots a_n, b_1b_2 \dots b_n) = |\{j \leq n : b_j = \varepsilon\}|,$$

$$p(a_1a_2 \dots a_n, b_1b_2 \dots b_n) = |\{i \leq n : a_i = \varepsilon\}| + |\{j \leq n : b_j = \varepsilon\}|$$

$$= p_r(a_1a_2 \dots a_n, b_1b_2 \dots b_n) + p_l(a_1a_2 \dots a_n, b_1b_2 \dots b_n)$$

and

$$\begin{aligned} &M_r(a_1a_2 \dots a_n, b_1b_2 \dots b_n) \\ &\quad = m(a_1a_2 \dots a_n, b_1b_2 \dots b_n) - p_r(a_1a_2 \dots a_n, b_1b_2 \dots b_n) \\ &M_l(a_1a_2 \dots a_n, b_1b_2 \dots b_n) \\ &\quad = m(a_1a_2 \dots a_n, b_1b_2 \dots b_n) - p_l(a_1a_2 \dots a_n, b_1b_2 \dots b_n) \\ &M(a_1a_2 \dots a_n, b_1b_2 \dots b_n) \\ &\quad = m(a_1a_2 \dots a_n, b_1b_2 \dots b_n) - p(a_1a_2 \dots a_n, b_1b_2 \dots b_n) \end{aligned}$$

as the *measures of proper similarity.*

The number $d_H(a_1a_2\dots a_n, b_1b_2\dots b_n) = |\{i \leq n : a_i \neq b_i\}|$ is the Hamming distance between decompositions and it is another type of penalty: we have that $p(a_1\dots a_n, b_1\dots b_n) \leq d_H(a_1\dots a_n, b_1\dots b_n)$.

The assertions from the following theorem establish the main results.

Theorem 2.1. *Let a and b be two non-empty strings, $a = a_1a_2\dots a_n$ and $b = b_1b_2\dots b_n$ be the initial optimal decompositions, and $a = a'_1a'_2\dots a'_q$ and $b = b'_1b'_2\dots b'_q$ be the second decompositions, which are arbitrary. Denote by*

$$\begin{aligned}
m &= m(a_1a_2\dots a_n, b_1b_2\dots b_n), & m' &= m(a'_1a'_2\dots a'_n, b'_1b'_2\dots b'_q),\\
p &= p(a_1a_2\dots a_n, b_1b_2\dots b_n), & p' &= p(a'_1a'_2\dots a'_n, b'_1b'_2\dots b'_q),\\
p_l &= p_l(a_1a_2\dots a_n, b_1b_2\dots b_n), & p'_l &= p_l(a'_1a'_2\dots a'_n, b'_1b'_2\dots b'_q),\\
p_r &= p_r(a_1a_2\dots a_n, b_1b_2\dots b_n), & p'_r &= p_r(a'_1a'_2\dots a'_n, b'_1b'_2\dots b'_q),\\
r &= d_H(a_1a_2\dots a_n, b_1b_2\dots b_n), & r' &= d_H(a'_1a'_2\dots a'_n, b'_1b'_2\dots b'_q),
\end{aligned}$$

$$\begin{aligned}
M &= m - p, & M' &= m' - p',\\
M_l &= m - p_l, & M'_l &= m' - p'_l,\\
M_r &= m - p_r, & M'_r &= m' - p'_r.
\end{aligned}$$

The following assertions are true:

1. $p' - p = 2(m' - m) + 2(r' - r)$.

2. *If the second decompositions are non optimal, then $M_l > M'_l$ and $M_r > M'_r$.*

3. *If the second decompositions are optimal, then $M_l = M'_l$ and $M_r = M'_r$ and the measures M_l and M_r are constant on the set of optimal parallel decompositions.*

4. *If $m' \geq m$ and the second decompositions are non optimal, then $p' > p$, $p_l\prime > p_l$, $p'_r > p_r$ and $M > M'$.*

5. *If $m' = m$ and the second decompositions are optimal, then $p' = p$, $p_l\prime = p_l$, $p'_r = p_r$ and $M' = M$.*

6. *If $m' \leq m$ and the second decompositions are non optimal, then $m' - r' < m - r$.*

The proof of Theorem 2.1 follows from the next lemmas.

Lemma 1.

$$p_r(a'_1a'_2 \dots a'_q, b'_1b'_2 \dots b'_q) = q - l(a),$$
$$p_l(a'_1a'_2 \dots a'_q, b'_1b'_2 \dots b'_q) = q - l(b),$$
$$p(a'_1a'_2 \dots a'_q, b'_1b'_2 \dots b'_q) = 2q - l(a) - l(b).$$

Proof. Follows immediately from the definitions of penalty factors and parallel decompositions. □

Lemma 2. $p' - p = 2(m' - m) + 2(r' - r)$.

Proof. From Lemma 1 it follows that $p' - p = (2q - l(a) - l(b)) - (2n - l(a) - l(b)) = 2(q - n)$. Since $q = m' + r'$ and $n = m + r$, the proof is complete. □

Lemma 3. $p'_l - p_l = p'_r - p_r = (m' - m) + (r' - r)$.

Proof. We can assume that $l(a) \leq l(b)$. Then $p_l = (l(b) - l(a)) + l_r$ and $p_l - p_r = l(b) - l(a)$. Hence $p_l - p_r = p'_l - p'_r$ and $p'_l - p_l = p'_r - p_r$. The equality $p' - p = (p'_r - p_r) + (p'_l - p_l)$ and Lemma 2 complete the proof. □

Lemma 4. *Assume that $m' > m$. Then:*

1. $M > M'$, $M_l \geq M'_l$ and $M_r \geq M'_r$.

2. $M_l > M'_l$ and $M_r > M'_r$ provided that the second decompositions are non optimal.

3. $M_l = M'_l$ and $M_r = M'_r$ provided that the second decompositions are optimal.

Proof. Since the initial decompositions are optimal, we have $r' \geq r$. Moreover, we have $r' = r$ if and only if the second decompositions are optimal as well. By virtue of definitions, we have $n = m + r$ and $q = m' + r'$. Therefore $n < q$. From Lemma 2, it follows that $p' - p = 2(m' - m) + 2(r' - r)$ and $p < p'$. Thus $p' - p > m' - m$ and $M = m - p > m' - p' = M'$.

Also, from Lemma 3, it follows that $p'_l - p_l = p'_r - p_r = (m' - m) + (r' - r)$. Hence, $M_l = m - p_l = (m' - p'_l) + (r' - r) = M'_l + (r' - r)$ and $M_r = m - p_r = (m' - p'_r) + (r' - r) = M'_r + (r' - r)$. Since $r' \geq r$ and $r' = r$ if and only if the second decompositions are optimal, the proof is complete. □

Corollary 2.1. *The measures M_l and M_r are constant on the set of optimal parallel decompositions.*

Lemma 5. *Let $m' = m$. Then:*

1. $M \geq M'$, $M_l \geq M'_l$ and $M_r \geq M'_r$.

2. $M_l > M'_l$ and $M_r > M'_r$ provided that the second decompositions are non optimal.

3. $M_l = M'_l$ and $M_r = M'_r$ provided that the second decompositions are optimal.

Proof. We have that $n = m + r$ and $q = m' + r'$. Since $r \leq r'$, we have that $n \leq q$.

Assume that $M < M'$. Then $m - p < m' - p'$, $p' = 2q - l(a) - l(b)$ and $p = 2n - l(a) - l(b)$. Hence $m - 2n + l(a) + l(b) < m - 2q + l(a) + l(b)$, or $-2n < -2q$ and $n > q$, a contradiction.

From Lemma 3 it follows that $p'_l - p_l = p'_r - p_r = r' - r$. Hence $p'_l \geq p_l$ and $p'_r \geq p_r$. If the second decompositions are non optimal, then $p'_l > p_l$ and $p'_r > p_r$. Assertions are proved. □

Lemma 6. *Assume that $m' < m$ and the second decompositions are non optimal. Then $M_l > M'_l$ and $M_r > M'_r$.*

Proof. Since the initial decompositions are optimal, we have $r' > r$. By virtue of Lemma 3, we have $p'_l - p_l = p'_r - p_r = (m' - m) + (r' - r)$. Hence, $M_l = m - p_l = (m' - p'_l) + (r' - r) = M'_l + (r' - r)$ and

$M_r = m - p_r = (m' - p'_r) + (r' - r) = M'_r + (r' - r)$. Since $r' - r > 0$, the proof is complete. □

Remark 2.1. *From Assertions 1 and 3 of Theorem 2.1 it follows that on the class of all optimal decompositions of given two strings:*

- *the maximal measure of proper similarity is attained on the optimal parallel decomposition with minimal penalties (minimal measure of similarity);*
- *the minimal measure of proper similarity is attained on the optimal parallel decomposition with maximal penalties (maximal measure of similarity).*

3 Computing algorithms

The algorithm of computing the Levenshtein distance for the case of a discrete metric was presented in [4]. Below we show a well known algorithm (see Algorithm 1, Annex 1) that permits to calculate the Graev-Markov-Levenshtein distance between two irreducible strings for any metric.

For any two non-empty strings there exist parallel decompositions with maximal measure of similarity and optimal decompositions on which measure of similarity is minimal. Pseudo-code of such algorithm is presented in Algorithm 2 (see Algorithm 2, Annex 2).

Algorithm 2 makes calls to functions *LevenshteinDistance* and *BuildOPD*. The first function computes distance function and builds the memoization matrix. The function *BuildOPD* uses the memoization matrix to generate optimal parallel decompositions. The pseudocode for these functions was presented in [4].

4 Conclusions

For any two non-empty strings there exist parallel decompositions with maximal measure of similarity and optimal decompositions on which measure of similarity is minimal. The following example shows that there exist some exotic non optimal parallel decompositions $a =$

$a'_1a'_2\dots a'_q$ and $b = b'_1b'_2\dots b'_q$, such that for optimal decompositions $a = a_1a_2\dots a_n$ and $b = b_1b_2\dots b_n$ we have $m' < m$, $p' < p$ and $M' > M$.

Example 4.1. *Let $a = ABCDEF$ and $b = CDEFED$ be trivial non optimal decompositions of strings a, b, and $a = ABCDEF\varepsilon\varepsilon$ and $b = \varepsilon\varepsilon CDEFED$ be their optimal decompositions. Then $m' = 1$, $r' = 5$, $p' = p'_l = p'_r = 0$ and $m = 4$, $r = 4$, $p = 4$, $p_l = p_r = 2$. In this example we have that $M'_l = M'_r = M' = m' - p' = 1 - 0 = 1 > 0 = 4 - 4 = m - p = M$, $m' - r' = -4 < 0 = m - r$, $M_l = 4 - 2 = 2 > 1 = M'_l$, $M_r = 4 - 2 = 2 > 1 = M'_r$.*

Example 4.2. *Let $a = AAAACCC$ and $b = CCCBBBB$ be trivial optimal decompositions of strings a, b, and $a = AAAACCC\varepsilon\varepsilon\varepsilon\varepsilon$ and $b = \varepsilon\varepsilon\varepsilon\varepsilon CCCBBBB$ be their non-optimal decompositions. Then $m' = 3$, $r' = 8$, $p' = 8$ and $m = 0$, $r = 7$, $p = p_l = p_r = 0$. In this example we have that $-5 = m' - r' > m - r = -7$ and $-5 = m' - p' < m - p = 0$.*

The above examples show that Theorem 2.1 cannot be improved in the case of $m' < m$.

Decompositions with minimal penalty and maximal proper similarity are of significant interest. Moreover, if we consider the problem of text editing and correction, the optimal decompositions are more favorable. Therefore, optimal decompositions are the best parallel decompositions and we may solve the string matching problems only on class of optimal decompositions.

To summarize the results, we established that optimal decompositions:

- describe the proper similarity of two strings;
- permit to obtain long common sub-sequences;
- permit to calculate the distance between strings;
- permit to appreciate changeability of information over time.

References

[1] V. B. Barahnin, V. A. Nehaeva and A. M. Fedotov, "Similarity Determination for Textual Documents Clusterization," *Vestnik*

Novosibirskogo Gos. Un-ta, Ser. Informactionnye tehnologii, vol. 6, no. 1, pp. 3–9, 2008. (in Russian).

[2] M. M. Choban, "The theory of stable metrics," *Math. Balkanica*, vol. 2, pp. 357–373, 1988.

[3] M. M. Choban and I. A. Budanaev, "Distances on Monoids of Strings and Their Applications," in *Proceedings of the Conference on Mathematical Foundations of Informatics MFOI2016*, (Chisinau, Republic of Moldova), 2016, pp. 144–159.

[4] M. M. Choban and I. A. Budanaev, "About Applications of Distances on Monoids of Strings," *Computer Science Journal of Moldova*, vol. 24, no. 3, pp. 335–356, 2016.

[5] M. M. Choban and L. L. Chiriac, "On free groups in classes of groups with topologies," *Bul. Acad. Ştiinţe Repub. Moldova, Matematica*, no. 2-3, pp. 61–79, 2013.

[6] M. I. Graev, "Free topological groups," *Trans. Moscow Math. Soc.*, vol. 8, pp. 303–364, 1962 (Russian original: *Izvestia Akad. Nauk SSSR*, vol. 12, 279–323, 1948).

[7] R. W. Hamming, "Error Detecting and Error Correcting Codes," *Bell System Technical Journal*, vol. 29, no 2, pp. 147–160, 1952.

[8] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *DAN SSSR*, vol. 163, no 4, pp. 845–848, 1965 (in Russian) (English translation: *Soviet Physics – Doklady*, vol. 10, no. 8, pp. 707–710, 1966).

[9] A. A. Markov, "On free topological groups," *Trans. Moscow Math. Soc.*, vol. 8, pp. 195–272, 1962.

[10] G. Navarro, "A guided tour to approximate string matching," *ACM Computing Surveys*, vol. 33, no 1, pp. 31–88, 2001.

[11] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of Molecular Biology*, vol. 48, no 3, pp. 443–453, 1970.

[12] S. Romaguera, M. Sanchis and M. Tkachenko, "Free paratopological groups," *Topology Proceed.*, vol. 27, no 2, pp. 613–640, 2003.

Mitrofan Choban, Ivan Budanaev, Received May 5, 2018

Mitrofan Choban
Tiraspol State University, Republic of Moldova
str. Iablochkin 5, Chisinau, Moldova
Phone: +373 69109553
E–mail: mmchoban@gmail.com

Ivan Budanaev
Doctoral School of Mathematics and Information Science
Institute of Mathematics and Computer Sciences
Tiraspol State University, Republic of Moldova
str. Academiei, 3/2, MD-2028, Chisinau, Moldova
E–mail: ivan.budanaev@gmail.com

ANNEX 1

Algorithm 1: Metric:
Given $x, y \in F(A)$ compute $\check{d}(x, y)$, for the case of metric.

Data: $x = x_1x_2 \dots x_n$, $y = y_1y_2 \dots y_m$, metric function $\check{d}$ on $\check{A}$.
Parameters: costs of insertion and removal operations - $cost_{insert}$ and $cost_{remove}$ respectively.
Result: $d_L(x, y)$, and matrix D.

```
// initialize distance matrix
for i ← 1 to m do D[i,0]=i;
for j ← 1 to n do D[0,j]=j;
// initialize loop variables
i := 1, j := 1;
for j ← 1 to n do
    for i ← 1 to m do
        if dist(x_i, y_j) = 0 then
            d[i, j] := d[i-1, j-1];
        else
            // Dynamic Programming recursive function
            d[i,j] := min(d[i-1, j] + cost_remove, min(d[i, j-1] +
              cost_insert, d[i-1, j-1] + dist(x_i, y_i)));
return D[m, n], D;
```

ANNEX 2

Algorithm 2: Maximal Measure of Similarity:
Finds maximum value of measure of similarity of $x, y \in L(\bar{A})$.

```
   /* Helper functions to compute similarity and
      penalty factors                                    */
 1 Function similarity (a,b)
 2 |   n = max(length(a), length(b))
 3 |   sim = 0
 4 |   for i ← 1 to n do
 5 |   |   if (a[i] == b[i]) sim = sim + 1
 6 |   return sim;
 7 Function penalty (a,b)
 8 |   n = max(length(a), length(b))
 9 |   pen = 0
10 |   for i ← 1 to n do
11 |   |   if (a[i] == ε) pen = pen + 1
12 |   |   if (b[i] == ε) pen = pen + 1
13 |   return pen;
   /* Main Algorithm Body                               */
   Data: x = x1x2 ... xn, y = y1y2 ... ym.
   Result: Maximal measure of similarity of x and y.
   // Calling Metric or QuasiMetric Functions
14 d, D := LevenshteinDistance(x, y);
   // Calling BackTracking function BuildOPD
15 S = BuildOPD(n,m,x,y,a,b,D);
16 max_sim = 0
17 for ((a,b): S) do
18 |   sim = similarity(a,b) - penalty(a,b)
19 |   max_sim = max_sim < sim ? sim : max_sim
20 return max_sim
```

Computer Science Journal of Moldova, vol.**26**, no.2(77), 2018

Strong stability measures for multicriteria quadratic integer programming problem of finding extremum solutions

Vladimir Emelichev, Yury Nikulin

Abstract

We consider a wide class of quadratic optimization problems with integer and Boolean variables. In this paper, the lower and upper bounds on the strong stability radius of the set of extremum solutions are obtained in the situation where solution space and criterion space are endowed with various Hölder's norms.

Keywords: Multicriteria problem, extremum solutions, strong stability radius, Hölder's norms.

1 Problem formulation and basic definitions

Let $A = [a_{ijk}]$ be a $n \times n \times m$-matrix with corresponding cuts $A_k \in \mathbf{R}^{n\times n},\ k \in N_m = \{1, 2, \ldots, m\},\ m \geq 1$. Let also $X \subseteq \mathbf{Z}^n, 2 \leq |X| \leq \infty$, be a set of feasible solutions (integer vectors) $x = (x_1, x_2, \ldots, x_n)^T$, $n \geq 2$.

We define a vector criterion

$$f(x, A) = \big(f_1(x, A_1), f_2(x, A_2), \ldots, f_m(x, A_m)\big) \to \min_{x\in X},$$

with partial criteria being quadratic functions

$$f_k(x, A_k) = x^T A_k x,\ k \in N_m.$$

In decision making theory, along with the well-known Pareto optimality principle (see e.g. [1]), various choice functions are considered [2]–[5]. In this paper, under m-criteria quadratic problem $Z_m(A)$

we understand the problem of finding the set of extremum solutions defined in traditional way (see e.g. [2]–[4]):

$$C_m(A) = \{x \in X : \ \exists s \in N_m \ \ \forall x' \in X \ \ \big(g_s(x, x', A_s) \leq 0\big)\},$$

where

$$g_s(x, x', A_s) = f_s(x, A_s) - f_s(x', A_s) = (x - x')^T A_s (x - x').$$

Thus, the choice of extremum solutions can be interpreted as finding best solutions for each of m criteria, and then combining them into one set. In other words, the set of extremum solutions contains all the individual minimizers of each objective. Obviously, $C_1(A),\ A \in \mathbf{R}^{n\times n}$ is the set of optimal solutions for scalar problem $Z_1(A)$ with $A \in \mathbf{R}^{n\times n}$.

Taking into account that X is finite, the following formulae below are true:

$$C_m(A) = S_m(A)\backslash(P_m(A)\backslash L_m(A)) = L_m(A) \cup (S_m(A)\backslash P_m(A)),$$

$$C_m(A) \cap P_m(A) = L_m(A),$$

$$L_m(A) \subseteq P_m(A) \subseteq S_m(A),$$

$$L_m(A) \subseteq C_m(A) \subseteq S_m(A),$$

where $P_m(A)$ denotes the Pareto set [6], $S_m(A)$ denotes the Slater set [7], and $L_m(A)$ denotes the lexicographic set (see e.g. [1],[8]). Below we define all the three sets in a traditional way (see e.g. [9]–[11]):

$$P_m(A) = \Big\{x \in X : \ X(x, A) = \emptyset\Big\},$$

$$S_m(A) = \Big\{x \in X : \ \nexists x^0 \in X \ \ \forall k \in N_m \ \ \big(g_k(x, x^0, A_k) > 0\big)\Big\},$$

$$L_m(A) = \bigcup_{\pi \in \Pi_m} L(A, \pi),$$

$$L(A, \pi) = \Big\{x \in X : \ \forall x' \in X \ \ \big(g(x, x', A) \leq_\pi 0_{(m)}\big)\Big\},$$

$$X(x, A) = \Big\{x' \in X : \ g(x, x', A) \geq 0_{(m)} \ \&\ g(x, x', A) \neq 0_{(m)}\Big\},$$

$$g(x,x',A)=\big(g_1(x,x',A_1),g_2(x,x',A_2),\dots,g_m(x,x',A_m)\big),$$

$$0_{(m)}=(0,0,\dots,0)\in\mathbf{R}^m.$$

Here Π_m is the set of all $m!$ permutations of numbers $1,2,...,m$; $\pi=(\pi_1,\pi_2,...,\pi_m)\in\Pi_m$; and the binary relation of lexicographic order between two vectors $y=(y_1,y_2,\dots,y_m)\in\mathbf{R}^m$ and $y'=(y'_1,y'_2,\dots,y'_m)\in\mathbf{R}^m$ is defined as follows

$$y\le_\pi y'\Longleftrightarrow$$

$$\Big(y=y'\Big)\vee\Big(\exists k\in N_m\ \ \forall i\in N_{k-1}\ \ \big(y_{\pi_k}<y'_{\pi_k}\ \&\ y_{\pi_i}=y'_{\pi_i}\big)\Big),$$

where $N_0=\emptyset$. Obviously all the sets, $P_m(A),S_m(A),L_m(A)$ and $C_m(A)$, are non-empty for any matrix $A=[a_{ijk}]\in\mathbf{R}^{n\times n\times m}$ due to the finite number of alternatives in X.

We will perturb the elements of matrix $A\in\mathbf{R}^{n\times n\times m}$ by adding elements of the perturbing matrix $A'\in\mathbf{R}^{n\times n\times m}$. Thus the perturbed problem $Z_m(A+A')$ of finding extremum solutions has the following form

$$f(x,A+A')\to\min_{x\in X}.$$

In the solution space $\mathbf{R}^n$, we define an arbitrary Hölder's norm $l_p,\ p\in[1,\infty]$, i.e. under norm of vector $a=(a_1,a_2,\dots,a_n)^T\in\mathbf{R}^n$ we understand the number

$$\|a\|_p=\begin{cases}\Big(\sum\limits_{j\in N_n}|a_j|^p\Big)^{1/p} & \text{if}\ \ 1\le p<\infty,\\[2ex] \max\{|a_j|:\ j\in N_n\} & \text{if}\ \ p=\infty.\end{cases}$$

Thus, for any matrix $A_k\in\mathbf{R}^{n\times n}$, the norm of the matrix is defined as a norm of vector composed of all the matrix elements.

In the criterion space $\mathbf{R}^m$, we define another Hölder's norm $l_q,\ q\in[1,\infty]$, i.e. under norm of matrix $A\in\mathbf{R}^{n\times n\times m}$ we understand the number

$$\|A\|_{pq}=\|(\|A_1\|_p,\|A_2\|_p,\dots,\|A_m\|_p)\|_q,$$

It is easy to see that

$$\|A_k\|_p \le \|A\|_{pq},\ k \in N_m. \tag{1}$$

Let ζ be either p or q. It is well-known that l_ζ norm, defined in $\mathbf{R}^n$, induces conjugated l_{ζ^*} norm in $(\mathbf{R}^n)^*$. For ζ and ζ^*, the following relations hold

$$\frac{1}{\zeta} + \frac{1}{\zeta^*} = 1, \qquad 1 < \zeta < \infty.$$

In addition, if $\zeta = 1$, then $\zeta^* = \infty$. Obviously, if $\zeta^* = 1$, then $\zeta = \infty$. Also notice that ζ and ζ^* belong to the same range $[1, \infty]$. We also set $\frac{1}{\zeta} = 0$ if $\zeta = \infty$.

For any two vectors a and b of the same dimension, the following Hölder's inequalities are well-known (see e.g. [12])

$$|a^T b| \le \|a\|_\zeta \|b\|_{\zeta^*}. \tag{2}$$

To any vector $x = (x_1, x_2, \ldots, x_n)^T \in \mathbf{Z}^n$, we assign a vector $\tilde{x}$ composed of all the possible products $x_i x_j$, i.e.

$$\tilde{x} = (x_1x_1, x_1x_2, \ldots, x_nx_{n-1}, x_nx_n)^T \in \mathbf{Z}^{n^2}.$$

Taking into account Hölder's inequalities (2), we can see that for any $x, x' \in \mathbf{Z}^n$ and $k \in N_m$ the following inequalities hold

$$|f_k(x, A_k)| = |x^T A_k x| = |A_k x x^T| \le \|A_k\|_p \|\tilde{x}\|_{p^*},$$

$$|g_k(x, x', A_k)| \le \|A_k\|_p \|\tilde{x} - \tilde{x'}\|_{p^*}. \tag{3}$$

It is easy to see that for any vector $a = (a_1, a_2, \ldots, a_n)^T \in \mathbf{R}^n$ with condition $|a_j| = \alpha$, $j \in N_n$, and any matrix $A_k = [a_{ijk}] \in \mathbf{R}^{n \times n}$ with condition $|a_{ijk}| = \alpha$, $(i, j) \in N_n \times N_n$, the following inequalities are valid

$$\|a\|_p = \alpha n^{\frac{1}{p}}, \tag{4}$$

$$\|A_k\|_p = \alpha n^{\frac{2}{p}}. \tag{5}$$

Given $\varepsilon > 0$, let

$$\Omega_{pq}(\varepsilon) = \left\{ A' \in \mathbf{R}^{n \times n \times m} : \ \ \|A'\|_{pq} < \varepsilon \right\}$$

be the set of perturbing matrices A' with cuts $A'_k \in \mathbf{R}^{n\times n},\ k \in N_m$, and $\|A'\|_{pq}$ is the norm of $A' = [a'_{ijk}] \in \mathbf{R}^{n\times n\times m}$. Denote

$$\Xi_{pq} = \Big\{\varepsilon > 0 :\ \ \forall\ A' \in \Omega_{pq}(\varepsilon)\ \ \big(C_m(A + A') \cap C_m(A) \neq \emptyset\big)\Big\}.$$

Following [10] and [13], the number

$$\rho_m(p,q) = \begin{cases} \sup\ \Xi_{pq} & \text{if}\ \ \Xi_{pq} \neq \emptyset, \\ 0 & \text{if}\ \ \Xi_{pq} = \emptyset \end{cases}$$

is called the *strong stability* (in terminology of [14] and [15] T_1-stability) radius of problem $Z_m(A)$, $m \in \mathbf{N}$, with Hölder's norms l_p and l_q in the spaces $\mathbf{R}^n$ and $\mathbf{R}^m$ respectively. Thus, the strong stability radius of problem $Z_m(A)$ defines the extreme level of independent perturbations of the elements of matrix A in the space $\mathbf{R}^{n\times n\times m}$ not leading to the situation where new extremum solutions appear.

2 Main result

Given $p, q \in [1,\infty]$, for problem $Z_m(A)$, $m \in \mathbf{N}$, we set

$$\phi_m(p) = \min_{x\notin C_m(A)}\ \min_{k\in N_m}\ \max_{x'\in X\setminus\{x\}}\ \frac{g_k(x,x',A_k)}{\|\tilde{x} - \tilde{x'}\|_{p^*}},$$

$$\psi_m(p) = \max_{x'\in C_m(A)}\ \max_{k\in N_m}\ \min_{x\notin C_m(A)}\ \frac{g_k(x,x',A_k)}{\|\tilde{x} - \tilde{x'}\|_{p^*}},$$

$$\gamma_m(p,q) = n^{\frac{2}{p}} m^{\frac{1}{q}} \min_{x\notin C_m(A)}\ \max_{k\in N_m}\ \max_{x'\in C_m(A)}\ \frac{g_k(x,x',A_k)}{\|\tilde{x} - \tilde{x'}\|_1}.$$

It is evident that if $C_m(A) = X$, the inequality

$$C_m(A + A') \cap C_m(A) \neq \emptyset$$

holds for any perturbing matrix $A' \in \Omega_{pq}(\varepsilon)$ with $\varepsilon > 0$. So, the stability radius is infinite when $C_m(A) = X$. The problem $Z_m(A)$ that satisfies $C_m(A) \neq X$ is called *non-trivial*.

Theorem 1. *Given $p, q \in [1, \infty]$ and $m \in \mathbf{N}$, for the strong stability radius $\rho_m(p, q)$ of non-trivial problem $Z_m(A)$, the following lower bound is valid*

$$\rho_m(p, q) \geq \max\{\phi_m(p), \psi_m(p)\} > 0.$$

In addition,

$$\gamma_m(p, q) \geq \rho_m(p, q) \geq \max\{\phi_m(p), \psi_m(p)\} > 0 \tag{6}$$

if $Z^m(A)$ is a problem with Boolean variables, i.e. if $X \subseteq \mathbf{E}^n$.

Proof. Since the formula

$$\forall x \notin C_m(A) \quad \forall k \in N_m \ \ \exists x^0 \in X \ \ \big(g_k(x, x^0, A_k) > 0\big),$$

is true, the inequality $\phi_m(p) > 0$ tells us that the lower bound on the strong stability radius as well as the strong stability radius itself are always positive.

First, we prove that $\rho_m(p, q) \geq \phi_m(p)$. Let $A' \in \Omega_{pq}(\phi_m(p))$ be a perturbing matrix with cuts $A'_k \in \mathbf{R}^{n \times n}$, $k \in N_m$. Then according to the definition of the number $\phi_m(p)$, for any index $k \in N_m$ and any solution $x \notin C_m(A)$ there exists a solution $x^0 \in X \backslash \{x\}$ such that

$$\frac{g_k(x, x^0, A_k)}{\|\tilde{x} - \tilde{x^0}\|_{p^*}} \geq \phi_m(p) > \|A'\|_{pq} \geq \|A'_k\|_p,$$

due to (1). Using (3) we conclude that for any $k \in N_m$ there exists $x^0 \neq x$ such that

$$g_k(x, x^0, A_k + A'_k) = g_k(x, x^0, A_k) + g_k(x, x^0, A'_k) \geq$$

$$g_k(x, x^0, A_k) - \|A'_k\|_p \|\tilde{x} - \tilde{x^0}\|_{p^*} > 0,$$

i.e. $x \notin C_m(A + A')$. Thus, any solution that is not extremum in the problem $Z_m(A)$, so stays in the problem $Z_m(A+A')$. Then we conclude that for any perturbing matrix $A' \in \Omega_{pq}(\phi_m(p))$ the inclusion holds $\emptyset \neq C_m(A + A') \subseteq C_m(A)$. It implies that $C_m(A + A') \cap C_m(A) \neq \emptyset$ for any $A' \in \Omega_{pq}(\phi_m(p))$, and hence $\rho_m(p, q) \geq \phi_m(p)$.

Further, we prove that $\rho_m(p,q) \geq \psi_m(p)$. Since the formula

$$\exists x' \in C_m(A) \quad \exists k \in N_m \quad \forall x \notin C_m(A) \quad (C_k(x - x') > 0)$$

is true, the inequality $\psi_m(p) > 0$ is also evident.

Let $A' \in \Omega_{pq}(\psi_m(p))$ be a perturbing matrix with cuts $A'_k \in \mathbf{R}^{n\times n}$, $k \in N_m$. Then according to the definition of the number $\psi_m(p)$, there exist index $s \in N_m$ and solution $x^0 \in C_m(A)$ such that for any solution $x \notin C_m(A)$ we have

$$\frac{g_s(x, x^0, A_s)}{\|\tilde{x} - \tilde{x^0}\|_{p^*}} \geq \psi_m(p) > \|A'\|_{pq} \geq \|A'_s\|_p,$$

due to (1). Using (3), we conclude that for any $x \notin C_m(A)$ and any $A' \in \Omega_{pq}(\psi_m(p))$ the following inequalities hold

$$g_s(x, x^0, A_s + A'_s) = g_s(x, x^0, A_s) + g_s(x, x^0, A'_s) \geq$$

$$g_s(x, x^0, A_k) - \|A'_s\|_p \|\tilde{x} - \tilde{x^0}\|_{p^*} > 0.$$

Therefore,

$$\Big(X \backslash C_m(A)\Big) \cap C_s(x^0, A_s + A'_s) = \emptyset,$$

where

$$C_s(x^0, A_s + A'_s) = \{x \in X : \ g_s(x^0, x, A_s + A'_s) > 0\}.$$

Thus, any solution that is not extremum in the problem $Z_m(A)$ so stays in the problem $Z_m(A + A')$. Then we conclude that for any perturbing matrix $A' \in \Omega_{pq}(\psi_m(p))$ the following inequality holds

$$C_m(A + A') \cap C_m(A) \neq \emptyset,$$

and hence $\rho_m(p,q) \geq \psi_m(p)$.

Further we will consider the problem $Z_m(A)$ with Boolean variables ($X \subseteq \mathbf{E}^n$). And we demonstrate that $\gamma_m(p,q) \geq \rho_m(p,q)$. According to the definition of number $\gamma_m(p,q)$, there exists a Boolean solution $x^0 =$

$(x_1^0, x_2^0, ..., x_n^0) \notin C_m(A) \subseteq \mathbf{E}^n$ such that for any extremum solution $x \in C_m(A)$ and any index $k \in N_m$ we get

$$\gamma_m(p,q)\|\tilde{x^0} - \tilde{x}\|_1 \geq n^{\frac{2}{p}} m^{\frac{1}{q}} g_k(x^0, x, A_k). \tag{7}$$

Setting $\varepsilon > \gamma_m(p,q)$, we define the elements a_{ijk}^0 of any cut A_k^0, $k \in N_m$, of the perturbing matrix A^0 according to the formula

$$a_{ijk}^0 = \begin{cases} -\delta & \text{if } \ x_i^0 x_j^0 = 1, \\ \delta & \text{if } \ x_i^0 x_j^0 = 0, \end{cases}$$

where

$$\gamma_m(p,q) < \delta n^{\frac{2}{p}} m^{\frac{1}{q}} < \varepsilon. \tag{8}$$

Then according to (4) and (5), we get

$$\|A_s^0\|_p = \delta n^{\frac{2}{p}},$$

$$\|A^0\|_{pq} = \delta n^{\frac{2}{p}} m^{\frac{1}{q}},$$

$$A^0 \in \Omega_{pq}(\varepsilon).$$

In addition, due to the construction of matrix A_k^0, for any solution $x \neq x^0$ we have

$$g_k(x^0, x, A_k^0) = (x^0 - x)^T A_k^0 (x^0 - x) =$$

$$\sum_{i \in N_n} \sum_{j \in N_n} a_{ijk}^0 (x_i^0 x_j^0 - x_i x_j) = -\delta \|\tilde{x^0} - \tilde{x}\|_1. \tag{9}$$

Using (7), (8) and (9), we continue

$$g_k(x^0, x, A_k + A_k^0) = g_k(x^0, x, A_k) + g_k(x^0, x, A_k^0) \leq$$

$$\Big(\gamma_m(p,q)(n^{\frac{2}{p}} m^{\frac{1}{q}})^{-1} - \delta\Big)\|\tilde{x^0} - \tilde{x}\|_1 < 0.$$

Thus, $x \notin C_m(A + A^0)$ when $x \in C_m(A)$. Summarizing, for any $\varepsilon > \gamma_m(p, q)$, we can guarantee the existence of the perturbing matrix $A^0 \in \Omega_{pq}(\varepsilon)$ such that

$$C_m(A + A^0) \cap C_m(A) = \emptyset,$$

i.e. $\rho_m(p.q) < \varepsilon$ for any number $\varepsilon > \gamma_m(p, q)$. So, inequality (6) holds. □

From the Theorem we get the following result.

Corollary 1 If $Z_m(A)$, $A \in \mathbf{R}^{m\times n}$, is a non-trivial problem with Boolean variables, i.e. if $C_m(A) \neq X \subseteq \mathbf{E}^n$, then for any $m \in \mathbf{N}$.

$$0 < \max\{\phi, \psi\} \leq \rho_m(\infty, \infty) \leq \gamma,$$

where

$$\phi = \min_{x\notin C_m(A)} \min_{k\in N_m} \max_{x'\in X\setminus\{x\}} \frac{g_k(x, x', A_k)}{\|\tilde{x} - \tilde{x'}\|_1},$$

$$\psi = \max_{x'\in C_m(A)} \max_{k\in N_m} \min_{x\notin C_m(A)} \frac{g_k(x, x', A_k)}{\|\tilde{x} - \tilde{x'}\|_1},$$

$$\gamma = \min_{x\notin C_m(A)} \max_{k\in N_m} \max_{x'\in C_m(A)} \frac{g_k(x, x', A_k)}{\|\tilde{x} - \tilde{x'}\|_1}.$$

Corollary 2 If $Z_1(A)$, $A \in \mathbf{R}^n$, is a scalar non-trivial problem with Boolean variables ($X \subseteq \mathbf{E}^n$), then the following formula holds

$$\rho_1(\infty, q) = \min_{x\notin C_1(A)} \max_{x'\in X\setminus\{x\}} \frac{g(x, x', A)}{\|\tilde{x} - \tilde{x'}\|_1}.$$

Finally, we notice that Corollary 2 proves the attainability of $\phi_m(\rho)$ and $\gamma_m(p, q)$ when $m = 1$ and $p = \infty$.

References

[1] V. Podinovski and V. Nogin, *Pareto-optimal solutions of multicriteria problems*. Moscow: FizMatLit, 2007.

[2] D. Molodtsov, *Stability of optimality principle*. Moscow: Nauka, 1987.

[3] L. Sholomov, *Logical methods for the investigation of discrete choice models*. Moscow: Nauka, 1989.

[4] D. Yudin, *Computational methods in decision making*. Moscow: Nauka, 1989.

[5] M. Aizerman and F. Alekserov, *Choice of alternatives: theoretical foundations*. Moscow: Nauka, 1990.

[6] V. Pareto, *Manuel d'ecoonomie politique*. Paris: Qiard, 1906.

[7] M. Slater, "Lagrange multipliers revisited: A contribution to nonlinear programming," in *Cowles Commission Discussion Paper 80*, 1950.

[8] R. Steuer, *Multiple criteria optimization: theory, computation and application*. New York: John Wiley, 1986.

[9] V. Emelichev, K. Kuzmin and V. Mychkov, "Estimates of stability radius of multicriteria Boolean problem with Hölder metrics in parameter spaces," *Bulletin of the Academy of Sciences of Moldova. Mathematics*, vol. 78, no. 2, pp. 74–81, 2015.

[10] V. Emelichev, E. Girlich, Y. Nikulin and D. Podkopaev, "Stability and regularization of vector problems of integer linear programming," *Optimization*, vol. 51, no. 4, pp. 645–676, 2002.

[11] V. Emelichev, Y. Nikulin and V. Korotkov, "Stability analysis of efficient portfolios in a discrete variant of multicriteria investment problem with Savage's risk criteria," *Computer Science Journal of Moldova*, vol. 75, no. 3, pp. 303–328, 2017.

[12] G. Hardy, J. Littlewood and G. Polya, *Inequalities (2nd ed.)*. Cambridge: Cambridge University Press, 1952.

[13] V. Emelichev and K. Kuzmin, "On a type of stability of a multicriteria integer linear programming problem in the case of a monotone norm," *Journal of Computer and Systems Sciences International*, vol. 46, no. 5, pp. 714–720, 2007.

[14] V. Emelichev and K. Kuzmin, "On the T1-stability radius of a multicriteria linear Boolean problem with Hölder norms in parameter spaces," *Taurida Journal of Computer Science Theory and Mathematics*, vol. 30, no. 1, pp. 49–64, 2016.

[15] I. Sergienko and V. Shilo, *Discrete Optimization Problems. Problems, methods, research*. Kiev: Naukova dumka, 2003.

Vladimir Emelichev, Yury Nikulin,

Received May 2, 2018

Vladimir Emelichev,
Belarusian State University
Nezavisimosti 4,
220030 Minsk, Belarus
E–mail: vemelichev@gmail.com

Yury Nikulin (corresponding author)
University of Turku
Vesilinnantie 5,
20014 Turku, Finland
E–mail: yury.nikulin@utu.fi

Computer Science Journal of Moldova, vol.**26**, no.2(77), 2018

Invertible Graphs of Finite Groups

T. Chalapathi, R.V.M.S.S. Kiran Kumar

Abstract

We investigate some properties of invertible graphs of finite groups, which are newly defined in this paper. The main results have been proved using finite group classification. For each finite group, the size, the girth, the diameter, the clique number and the chromatic number have been studied. These studies show that the invertible graphs are weakly perfect. Specifically, formulas for enumerating a total number of edges in the invertible graph of the Symmetric group and Dihedral group have been derived. Further, the relations between isomorphic, non-isomorphic groups and their invertible graphs are presented.

Keywords: Self-inverse elements, Mutual inverse elements, Weakly-perfect, Isomorphic graphs, Invertible graph.

1 Introduction

Abstract algebra is largely concerned with the study of abstract sets and endowed with one or more binary operations. In this paper, we consider one of the basic algebraic structures known as group. The concept of finite group plays a fundamental role in the theory of group-theoretic graphs. The aim of this paper is to discuss some of the interconnections which exist between graphs and groups. Many authors in graph theory specify so many specific graph-theoretic properties, and results have analogs for algebraic systems such as semi groups, groups, rings, fields, etc. Our main purpose inthis paper is to describe some interactions between finite graphs, and finite groups have been exploited to give new results about group-theoretic graphs. The theory of group-theoretic graphs has provided an interesting and powerful structural

abstract approach to the study of the symmetries and non-symmetries of various configurations in the modern design theory and communication science. In recent years, a theory of group-theoretic graphs has found many applications in engineering and applied science, and many articles have been published on group-theoretic graphs such as [1]-[5].

In the present investigation, we write about a group theoretic graph, namely, invertible graph $IG(G)$ of a basic algebraic structure G, a finite group. However, finite group is a core in this paper. Our algebraic approach here is realized on group theoretic graphs with group elements and their corresponding binary operation. Although it is not quite elementary, it is an important aspect in dealing with the inter relation between simple graphs and finite groups.

For a finite group G, we denote by $S(G)$ the set of all self inverse elements, and by $M(G)$ – the set of mutual inverse elements of G. In this paper, we prove that there are some relations between G, $S(G)$, $M(G)$ and $IG(G)$. We classify the finite groups whose invertible graph is one of connected, complete but not bipartite graphs. Also we prove that $IG(G)$ is never Eulerian. For any given finite group G, we estimate the degree, the size, the girth, the diameter, the clique number and the chromatic number. We also discuss isomorphic theorems with some applications and structure of invertible graphs of finite abelian, non-abelian and cyclic groups.

2 Definition and Notations

Now we recall some basic definitions and notations of group theory from [6]. Let G be a finite group with identity e. Then the number of elements in G is the order of G and is denoted by $|G|$. If $a \in G$, then the order of a is $|a| = | < a > |$, where $< a >= \{a^n : n = 0, \pm 1, \pm 2, ...\}$ is a cyclic subgroup of G generated by a. If $G =< a >$, then G must be a cyclic group.

Usually, Z_n is the group of integers over addition modulo n, $U_n = Z_n^*$ is the group of multiplicative inverse elements of modulo n, $Z_p^* = Z_p - \{0\}$ is the multiplicative group of integers modulo p, S_n is the symmetric group of degree n, D_n is the dihedral group of order $2n$, Q_8

is the quaternion group. Further, we have $G_{2p} = \{2, 4, 6, ..., 2(p-1)\}$, a group of order $p-1$ with respect to multiplication modulo $2p$, a prime $p > 2$.

Theorem 1. *(Lagrange's Theorem, [6]) If H is a subgroup of the finite group G, then the order of H divides the order of G.*

Theorem 2. [6] *If $a \in G$ and G is finite, then $|a|$ divides $|G|$.*

We use [7] and [8] for the standard terminology of simple and algebraic graphs, respectively. Let X be a finite simple graph. We denote the vertex set and the edge set of X is $V(X)$ and $E(X)$, respectively. If $a \in V(X)$, then the degree of a denoted by $deg(a)$. If a and b are two adjacent vertices of X then we write $a-b$. A graph X in which any two distinct vertices are adjacent is said to be complete. If any two vertices a and b in X are connected by a path $a = a_0 - a_1 - ... - a_n = b$ then X is called connected graph. A path is a cycle if $a = b$. The length of a path or a cycle is the number of distinct edges in it. A cycle of length n is denoted by C_n.

3 Self and Mutual Inverse Elements of a Group

We introduce in this section the concepts of self and mutual inverse elements of a finite group with a few examples. The results of this section, though simple, are used throughout the paper.

Definition 1. *Let $(G, *)$ be a finite group with the identity e. Then an element $a \in G$ is called a self inverse element of G if $a = a^{-1}$, where a^{-1} is the inverse of a in G. The set of self inverse elements of G is $S(G)$ and its cardinality is $|S(G)|$.*

Next, an element $a \in G$ is called a mutual inverse element of G if there exists $b \in G$ such that $a * b = b * a = e$. The set of mutual inverse elements of G is denoted by $M(G)$. In particular, $M(G) = \{a \in G : a \neq a^{-1}\}$.

In the preceding definition, we have temporarily reverted to the $*$ notation for group operations to remind you that in a specific group, the operation might be addition, multiplication, or something else.

For any finite group G, $S(G)$ is a subgroup of G and $M(G)$ is not a subgroup of G. If $|G| > 2$ and G is a finite cyclic group, then $S(G) \neq G$. We are now ready to state and prove several results about $S(G)$ of G. The proof of the first theorem is implicit in our discussion of a finite cyclic group.

Theorem 3. *Let G be a finite cyclic group. Then*

$$|S(G)| = \begin{cases} 1, & \text{if } |G| \text{ is odd} \\ 2, & \text{if } |G| \text{ is even} \end{cases}.$$

Proof. For each finite cyclic group G, we have $G = S(G) \cup M(G)$ and $S(G) \cap M(G) = \phi$. Now consider two cases on $|G|$.
Case 1. If $|G|$ is odd, then we have to prove that $|S(G)| = 1$. Suppose $|S(G)| \geq 2$. Assume that $|S(G)| = 2$. Therefore, $S(G) = \{e, a : a^2 = e\}$. This implies that $|a| = 2$. By the Theorem 2, $|a||G|$, which is a contradiction to the fact that $|G|$ is odd. Hence $|S(G)| = 1$.
Case 2. If $|G|$ is even, then we shall show that $|S(G)| = 2$. Without loss of generality we may assume that $|S(G)| = 3$. This implies that every non-identity of $S(G)$ has order 2. That is, $a \neq b$ in $S(G)$ such that $a = a^{-1}$ and $b = b^{-1} \Rightarrow (ab) = (ab)^{-1}$, since G is abelian. It is clear that $ab \in S(G)$, and $S(G) = \{e, a, b, ab\}$ which is a contradiction to our assumption that $|S(G)| = 3$. Hence $|S(G)| = 2$. □

We next observe one of the most important results of $S(G)$. That is, if G is not a cyclic group of even order, then $|S(G)| \geq 2$. The following examples illustrate this point.

Example 1. *Since $e = e^{-1}$, $a = a^{-1}$, $b = b^{-1}$ and $c = c^{-1}$ in the Klein group $V_4 = \{e, a, b, c\}$, therefore, $|S(V_4)| = 4$.*

Example 2. $|S(S_3)| = 4$, $|S(D_3)| = 4$, $|S(Q_8)| = 2$, $|S(Z_2 \times Z_2)| = 4$.

According to the above examples, the following consequences specify the orders of $S(G)$ and $M(G)$ in a given finite group G.

Corollary 1. *Let G be a finite group of even order. Then $|S(G)|$ and $|M(G)|$ are both even.*

Proof. It is obvious, since a finite group G can be written as disjoint union of $S(G)$ and $M(G)$. □

Corollary 2. *Let G be a finite group of odd order. Then $|S(G)| = 1$ and $|M(G)| = |G| - 1$.*

Proof. It is obviously true because $|G|$ is odd if and only if $S(G) = \{e\}$. □

Example 3. $|S(Z_3 \times Z_3)| = 1$, $|S(Z_3 \times Z_5)| = 1$.

Remark 1. [9] *If there is a one-to-one mapping $a \leftrightarrow a^{'}$ of the elements of a group G onto those of a group $G^{'}$, and if $a \leftrightarrow a^{'}$ and $b \leftrightarrow b^{'}$ imply $ab \leftrightarrow a^{'}b^{'}$, then we say that G and $G^{'}$ are isomorphic and write $G \cong G^{'}$. If we put $a^{'} = f(a)$ and $b^{'} = f(b)$ for $a, b \in G$, then $f : G \to G^{'}$ is a bijection satisfying $f(ab) = a^{'}b^{'} = f(a)f(b)$.*

Lemma 1. *Let G and $G^{'}$ be any two finite groups. If $G \cong G^{'}$, then $S(G) \cong S(G^{'})$. But converse is not true.*

Proof. Suppose $G \cong G^{'}$. Then, by the Remark 1, there exists a group isomorphism f from G onto $G^{'}$ with the relation $f(a) = a^{'}$, for every $a \in G$ and $a^{'} \in G^{'}$. Now define a map $\varphi : S(G) \to S(G^{'})$ by the relation $\varphi(s) = s^{'}$ for every s in $S(G)$. Let $s, t \in S(G)$. If $\varphi(s) = \varphi(t)$, then $s^{'} = t^{'} \Rightarrow f(s) = f(t) \Rightarrow s = t$, since f is one-to-one. By the way φ was constructed, we see that φ is onto. The only condition that remains to be checked is that φ is operation preserving. To do this, let s and t belong to $S(G)$. Then obviously $s, t \in G$. Therefore $\varphi(st) = f(st) = s^{'}t^{'} = f(s)f(t) = \varphi(s)\varphi(t)$. Hence $S(G) \cong S(G^{'})$. But the converse of this result is not true. For example, $S(U_6) = \{1, 5\}$ and $S(U_{10}) = \{1, 9\}$. It is clear that $S(U_6) \cong S(U_{10})$, but U_6 is not isomorphic to U_{10}. □

4 Properties of Invertible Graphs

This section introduces invertible graph of a finite group and a study of its basic properties such as degree, size, connectedness and completeness. Further, we obtain a formula for finding the clique number,

the chromatic number and hence prove that invertible graph is weakly perfect.

We begin with the notion and definition of the invertible graph of a finite group.

Definition 2. *An undirected simple graph $IG(G)$ is called invertible graph of a finite group G whose vertex set is G and two distinct vertices a and b in G are adjacent in $IG(G)$ if and only if either $a \neq b^{-1}$, or, $b \neq a^{-1}$, where a^{-1} is the inverse of the element a in G.*

Before exploring the results and concepts of invertible graphs, instead of $a * b$, we shall write ab. The preceding definition can be visualized as shown in Figure 1. If $f_1(x) = x, f_2(x) = 1 - x, f_3(x) = \frac{1}{x}, f_4(x) = 1 - \frac{1}{x}, f_5(x) = \frac{1}{1-x}, f_6(x) = \frac{x}{1-x}$ are functions from $R - \{0, 1\}$ to $R - \{0, 1\}$, then the set $G = \{f_1, f_2, f_3, f_4, f_5, f_6\}$ is a non-abelian group under composition of functions. Here $S(G) = \{f_1, f_2, f_3, f_6\}$.

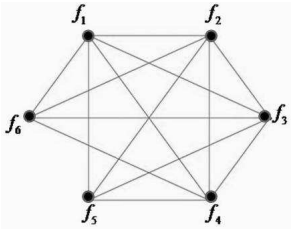


Figure 1. The Graph $IG(G)$.

Theorem 4. *For any finite group G, the invertible graph of G is a connected graph.*

Proof. It is obvious since e is the identity element of G, $ae \neq e$ for every $a \neq e$ in G, so that the vertex e is adjacent with remaining all the vertices of $IG(G)$. Hence $IG(G)$ is a connected graph. □

Theorem 5. *Let a be an element of a finite group G. Then*

$$deg(a) = \begin{cases} |G| - 1, & \text{if } a \in S(G) \\ |G| - 2, & \text{if } a \notin S(G) \end{cases} .$$

Proof. If $a \in S(G)$, then there exists $b \neq a$ in G such that $ab = a^{-1}b \neq e$. This implies that the vertex a is adjacent to all other vertices of $IG(G)$ if and only if $a \in S(G)$, therefore it is easy to derive that the degree of vertex a is $|G| - 1$.

If $a \notin S(G)$, then a has mutual inverse, say $b \neq a^{-1}$ in G such that $ab = e = ba$. It is clear that the vertex a is adjacent to all vertices of $IG(G)$ except b. However, if $a \notin S(G)$, then the vertex a is not adjacent to exactly one vertex of the graph $IG(G)$. Hence $deg(a) = |G| - 2$. □

Theorem 6. [8] *A connected graph is Eulerian if and only if the degree of each vertex is even.*

Corollary 3. *The invertible graph $IG(G)$ is never Eulerian.*

Proof. By the Theorem 5, it is clear that the degree of each vertex in $IG(G)$ is either $|G| - 1$ or $|G| - 2$. If $|G|$ is even, then $|G| - 1$ is odd. On the other hand, if $|G|$ is odd, then $|G| - 2$ is also odd. Hence in both cases, we found that the degree of each vertex in $IG(G)$ cannot be even. Thus, by the Theorem 6, the result follows. □

In view of the Theorem 5, the following Remark is obvious.

Remark 2. *Let $|G| > 3$ and $S(G) \neq G$. Then the graph $IG(G)$ is never a regular, a cycle, a star and a triangle free graph.*

Theorem 7. [8] *The total number of edges of the simple graph of order n is $\binom{n}{2}$.*

By combining Theorems 5 and 7, we can easily count the number of edges (size) in an invertible graph of a given finite group. For convenience, we introduce the following theorem.

Theorem 8. *For any finite group G, the size of invertible graph $IG(G)$ is* $\frac{1}{2}(|S(G)|(|G| - 1) + |M(G)|(|G| - 2))$.

By using Theorem 8, we derive a formula for enumerating the total number of edges in $IG(S_n)$ and $IG(D_n)$, respectively.

Theorem 9. *The size of invertible graph of the symmetric group S_n, $n > 1$, is $\frac{1}{2}((n!)^2 - 2n! + s(n))$, where $s(n)$ is the number of self inverse elements in S_n.*

Proof. Let $s(n)$ be the number of elements in S_n satisfying the relation $a = a^{-1}$, for every $a \in S_n$. Then $s(n)$ satisfies the recurrence relation, see [10], $s(n+2) = s(n+1) + (n+1)s(n)$, where $s(1) = 1$, $s(2) = 2$. In view of the Theorem 8, the size of the graph $IG(S_n)$ is $|E(IG(S_n))| = \frac{1}{2}(s(n)(n! - 1) + (n! - s(n))(n! - 2)) = \frac{1}{2}((n!)^2 - 2n! + s(n))$. □

Theorem 10. *The size of invertible graph of the Dihedral group of order $2n$ is*

$$|E(IG(D_n))| = \begin{cases} \frac{1}{2}(4n^2 - 3n), & \text{if } n \text{ is even} \\ \frac{1}{2}(4n^2 - 3n + 1), & \text{if } n \text{ is odd} \end{cases}.$$

Proof. Let $D_n = \{1, a, a^2, ..., a^{n-1}, b, ab, a^2b, ..., a^{n-1}b : a^n = 1, b^2 = 1, bab^{-1} = a^{-1}\}$. In D_n, the n elements $b, ab, a^2b, .., a^{n-1}b$ always have order 2. If n is even, then $a^{n/2}$ also has order 2. Therefore, the total number of elements in D_n, n is even, satisfying the relation $x = x^{-1}$ is $n + 2$. Similarly, if n is odd, there are $n + 1$ self inverse elements in D_n. However,

$$|S(D_n)| = \begin{cases} n+2, & \text{if } n \text{ is even} \\ n+1, & \text{if } n \text{ is odd} \end{cases} and |M(D_n)| = \begin{cases} n-2, & \text{if } n \text{ is even} \\ n-1, & \text{if } n \text{ is odd} \end{cases}.$$

By the Theorem 8, we have

$$|E(IG(D_n))| = \begin{cases} \frac{1}{2}(4n^2 - 3n), & \text{if } n \text{ is even} \\ \frac{1}{2}(4n^2 - 3n + 1), & \text{if } n \text{ is odd} \end{cases}.$$

□

Corollary 4. *The size of an invertible graph of a finite cyclic group G is*

$$|E(IG(G))| = \begin{cases} \frac{1}{2}(|G|-1)^2, & \text{if } n \text{ is even} \\ \frac{1}{2}(|G|^2-2|G|+2), & \text{if } n \text{ is odd} \end{cases}.$$

Proof. Let G be a finite cyclic group. Then there are two cases for $|G|$.
Case 1. Let $|G|$ be odd. Then $|G|-1$ is even. In view of Theorem 8, the total number of non-adjacent edges in $IG(G)$ is $\frac{1}{2}(|G|-1)$. But the maximum number of edges in a simple graph of order $|G|$ is $\binom{|G|}{2}$. So in this case the total number of edges in $IG(G)$ is $\binom{|G|}{2}$ - $\frac{1}{2}(|G|-1) = \frac{1}{2}(|G|-1)^2$.
Case 2. Let $|G|$ be even. Then, in view of Theorem 5, there are exactly $\frac{|G|}{2}-1$ pairs of distinct vertices that satisfy the relation $ab=e$ in G. Therefore, the total number of non-adjacent pairs in $IG(G)$ is $\frac{|G|}{2}-1$. So in this case the total number of edges in $IG(G)$ is $\binom{|G|}{2}$ -$(\frac{|G|}{2}-1) = \frac{1}{2}(|G|^2-2|G|+2)$. □

By using Theorem 8, the following short table illustrates the way we can easily determine the size of an invertible graph of some finite groups.

Group G	Z_p^*	G_{2p}	$U(2^k)$	S_3
Size of $IG(G)$	$\frac{1}{2}(p^2-4p+5)$	$\frac{1}{2}(p^2-4p+5)$	$2^{k-1}(2^k-2)+2$	14

Theorem 11. *Let $S(G) \neq G$. Then $IG(G)$ is never a complete graph.*

Proof. Suppose on the contrary that, $IG(G)$ is a complete graph. Then by the Theorem 7, the size of $IG(G)$ is $\binom{|G|}{2} = \frac{|G|}{2}(|G|-1)$, but in view of Theorem 8, we arrived at a contradiction to the completeness of $IG(G)$. □

Our next theorem shows how the bi-implication of $S(G) = G$ and completeness of $IG(G)$ are intertwining.

Theorem 12. *The invertible graph $IG(G)$ is complete if and only if $S(G) = G$.*

Proof. **Necessity.** Suppose that $IG(G)$ is a complete graph of a finite group G. Then any two vertices a and b in G are adjacent in $IG(G)$. Consequently $ab \neq e$, for every $a, b \in G$. This implies that $a \neq b^{-1}$ and $b \neq a^{-1}$. Therefore $a = a^{-1}$ and $b = b^{-1}$. That is, $a, b \in S(G)$. This shows that $G \subseteq S(G)$, also since $S(G) \subseteq G$. Hence $S(G) = G$.
Sufficiency. Let $S(G) = G$. Suppose $IG(G)$ is not a complete graph. Then there exist distinct vertices a and b in G such that $ab = e$ and $ba = e$. This implies that $a^{-1} = b$ and $b^{-1} = a$. It is clear that $a, b \notin S(G)$. Therefore, $S(G) \neq G$, which is a contradiction to our hypothesis, and hence $IG(G)$ is complete. □

We are now ready to prove a number of useful consequences of Theorem 12.

Corollary 5. *The graph $IG(G)$ is complete if and only if G is isomorphic to one of the groups, $Z_2 \times Z_2$, U_4, U_6, U_8, U_{12} and V_4.*

Proof. It is true from the fact that the Klein four-group V_4 is isomorphic to $Z_2 \times Z_2$, U_8, U_{12}. Also the group Z_2 is isomorphic to U_4, U_6. □

Corollary 6. *The invertible graph of G is complete if and only if $|G| = 2$.*

Proof. We have, $|G| = 2 \Leftrightarrow G = \{e, a : a^2 = e\} \Leftrightarrow G = S(G)$. □

Before going to further properties of invertible graph, let us consider the following example for the description of the result in the Theorem 12.

Example 4. *The invertible graph of the group $(P(X), \Delta)$ is complete. Let $X = \{a, b, c\}$ and let $A = \{a\}$, $B = \{b\}$, $C = \{c\}$ so that $\overline{A} = \{b, c\}$, $\overline{B} = \{a, c\}$ and $\overline{C} = \{a, b\}$. Then $P(X) = \{\phi, A, B, C, \overline{A}, \overline{B}, \overline{C}, X\}$ is*

an abelian group with respect to the symmetric difference Δ of sets and $S(P(X)) = P(X)$ but $P(X)$ is not a Klein four-group. The Figure 2 shows the complete invertible graph of the group $(P(X), \Delta)$.

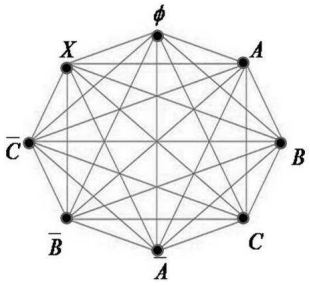


Figure 2. The invertible graph of $(P(X), \Delta)$.

Theorem 13. [8] *A simple graph is bipartite if and only if it does not have any odd cycle.*

Theorem 14. *If $|G|$ is composite, then $IG(G)$ is not a bipartite graph.*

Proof. Assume that $|G|$ is composite. Suppose $IG(G)$ is a bipartite graph. Then there exists a bipartition $(S(G), M(G))$. Without loss of generality we assume that $e \notin M(G)$. Since $|G|$ is composite, so there exists at least one self-inverse element $s \neq e$ in $S(G)$. If $m \in M(G)$ such that $m^{-1} \neq m$, then clearly $es \neq e$, $sm \neq e$ and $me \neq e$. Therefore, the triads (s, e, m) of the graph $IG(G)$ form a triangle. This violates the condition in the Theorem 13 for a bipartite graph. Hence $IG(G)$ is not a bipartite graph. □

Theorem 15. *Let $|G| > 3$. Then the girth of invertible graph $IG(G)$ is 3.*

Proof. We know that the girth of a simple graph is the length of a smallest cycle. Here there exist two cases.
Case 1. Suppose $S(G) = G$ and $|G| > 3$. Then, by the Theorem 12, $IG(G)$ is complete. Therefore $IG(G)$ has a smallest cycle of length 3. Hence, $gir(IG(G)) = 3$.

Case 2. Suppose $S(G) \neq G$ and $|G| > 3$. The vertex e in $IG(G)$ is adjacent to all other vertices. For this reason we can choose two vertices $a \neq e$ and $b \neq e$ in $IG(G)$ such that $a^{-1} = a$ and $b^{-1} \neq b$. Then $ab \neq ab^{-1} \Rightarrow ab \neq a^{-1}b^{-1} \Rightarrow ab \neq (ba)^{-1} \Rightarrow ab \neq e$, since $S(G) \neq G$. This implies that $ea \neq e, ab \neq e$ and $be \neq e$, so the graph $IG(G)$ always has a three cycle $e - a - b - e$, which is the smallest. Hence, $gir(IG(G)) = 3$. □

For distinct vertices x and y of a simple graph X, the diameter of X is $diam(X) = max\{d(x, y) : x, y \in V(X)\}$, where $d(x, y)$ is the length of the shortest path from x to y in X.

Theorem 16. *If $|G| > 1$, then the diameter of invertible graph is either 1 or 2.*

Proof. Let G be a finite group with $|G| > 1$. Then we consider the following two cases on $S(G)$.
Case 1. Suppose $S(G) = G$. In view of Theorem 12, $IG(G)$ is complete, hence $diam(IG(G)) = 1$.
Case 2. Suppose $S(G) \neq G$. Then, $IG(G)$ is never a complete graph. Let us assume that a and b are any two vertices in $IG(G)$. However, if the vertex $a \neq e$ is adjacent to vertex $b \neq e$, then trivially $d(a, b) = 1$. Otherwise, if a is not adjacent to b in $IG(G)$, then clearly $d(a, b) > 1$, where $a \neq e$ and $b \neq e$, but in the graph $IG(G)$ there always exists a path $a-e-b$ of the shortest length 2. It follows that $diam(IG(G)) = 2$.

From Case 1 and Case 2 we conclude that the diameter of invertible graph is either 1 or 2. □

A clique in a simple graph X is a complete subgraph. A clique Y in X is called maximal if no vertex set outside of Y is adjacent to all members of Y. The size of the largest clique in X is called the clique number $\omega(X)$. Simply, $\omega(X)$ is the maximum number of pair wise adjacent vertices. For any simple graph X, $1 \leq \omega(X) \leq |V(X)|$.

Theorem 17. *Let G be a finite group. Then the clique number of $IG(G)$ is $\omega(IG(G)) = \frac{1}{2}(|G| + |S(G)|)$.*

Proof. If $s \in S(G)$, then clearly the vertex s is adjacent to all other vertices of $IG(G)$ since $sa \neq e$ for every a in G. Therefore, the pair of non-adjacent vertices in $IG(G)$ is of degree $\frac{1}{2}|M(G)|$, and hence the total number of mutually adjacent vertices in $IG(G)$ is $|G|-\frac{1}{2}|M(G)| = \frac{1}{2}(|G|+|S(G)|)$, which is $\omega(IG(G))$. □

Theorem 18. *Let G be a finite cyclic group. Then the clique number of $IG(G)$ is $\omega(IG(G)) = \begin{cases} \frac{1}{2}(|G|+1), & \text{if } |G| \text{ is odd} \\ \frac{1}{2}(|G|+2), & \text{if } |G| \text{ is even} \end{cases}$.*

Proof. We know that the order of invertible graph $IG(G)$ of a finite cyclic group G is $|G|$. We need the following two cases on $|G|$.

Case 1. $|G|$ is odd. Trivially any two distinct vertices a and b are non adjacent if and only if $ab = e$. It follows that any vertex a is non-adjacent with exactly one vertex b, and hence total number of such vertices in $IG(G)$ is $|G| - 1$. It is clear that, the pair of non-adjacent vertices in $IG(G)$ is of degree $\frac{1}{2}(|G| - 1)$, and hence the total number of mutually adjacent vertices in $IG(G)$ is $|G|-\frac{1}{2}(|G|-1) = \frac{1}{2}(|G|+1)$, which is the size of a maximum clique.

Case 2. $|G|$ is even. So in this case the pair of non-adjacent vertices in $IG(G)$ is of degree $\frac{1}{2}(|G|-2)$. Hence total number of mutually adjacent vertices in $IG(G)$ is $|G| - \frac{1}{2}(|G| - 2) = \frac{1}{2}(|G| + 2)$, which is the size of a maximum clique. This completes the proof of the theorem. □

Example 5. *The following table shows the values of $\omega(IG(G))$ for some non-cyclic groups G.*

Non-cyclic group	V_4	Q_8	S_3	D_3	D_4
$\omega(IG(G))$	4	5	5	5	7

Definition 3. *A simple graph X is n-colorable if there exists a colouring of X which uses n colours. The minimum number of colors required*

to color a graph X is called the chromatic number and is denoted by $\chi(X)$. Note that $\omega(X) \leq \chi(X) \leq |V(X)|$. If $\chi(X) = \omega(X)$, then the graph X is called weakly-perfect.

Definition 4. *In a simple graph X, the set of pair-wise non-adjacent vertices is called an independent set of vertices.*

Theorem 19. *Let G be a finite group. Then the chromatic number of $IG(G)$ is $\chi(IG(G)) = \frac{1}{2}(|G| + |S(G)|)$.*

Proof. **Case 1.** Suppose that $S(G) \neq G$. In this case $a \in M(G)$ if and only if a is not adjacent with exactly one vertex in $IG(G)$. Therefore, the maximum independent set of $IG(G)$ is of size 2, moreover, total number of such independent sets in $IG(G)$ is $\frac{1}{2}|M(G)|$. For all these vertices we need $\frac{1}{2}|M(G)|$ colors, since each independent set in $IG(G)$ is uniquely colorable. But, vertices in $S(G)$ are adjacent with all the vertices in $M(G)$, and thus we require for $S(G)$ more colors distinct from these colors. Hence, the minimum number of colors required to colour the invertible graph is

$$\chi(IG(G)) = \frac{1}{2}|M(G)| + |S(G)| = \frac{1}{2}(|G| + |S(G)|),$$

since $G = S(G) \cup M(G)$ and $S(G) \cap M(G) = \phi$.
Case 2. Suppose $S(G) = G$. Then, trivially, $|M(G)| = 0$. But, by the Theorem 12 the graph $IG(G)$ is complete, therefore the required result is obviously true. That is, $\chi(IG(G)) = \frac{1}{2}(|G| + |S(G)|)$. □

Theorem 20. *Let G be a finite cyclic group. Then the chromatic number of $IG(G)$ is*

$$\chi(IG(G)) = \begin{cases} \frac{1}{2}(|G|+1), & \text{if } |G| \text{ is odd} \\ \frac{1}{2}(|G|+2), & \text{if } |G| \text{ is even} \end{cases}.$$

Proof. **Case 1.** Suppose that $|G| = 2$. Then obviously, $IG(G) \cong K_2$, and hence $\chi(IG(G)) = 2$.

Case 2. Suppose that $|G| > 2$. Then $IG(G)$ is never a complete graph since $S(G) \neq G$ for any finite cyclic group G with $|G| > 2$. Now we shall show the required result with the help of the following two subcases.

Subcase 1. Suppose that $|G|$ is even. Then, $|S(G)| = 2$, since G is cyclic. Therefore the order of the independent set in the graph $IG(G)$ is $\frac{1}{2}(|G| - 2)$. In fact each independent set is uniquely colorable, it means that for all these vertices we need $\frac{1}{2}(|G| - 2)$ colors. However two vertices in $S(G)$ are adjacent with remaining all vertices in $IG(G)$, thus minimum number of colors to color the invertible graph is $\frac{1}{2}(|G| - 2) + 2 = \frac{1}{2}|G| + 1$.

Subcase 2. Suppose that $|G|$ is odd. Then, $|S(G)| = 1$, since G is cyclic. Therefore the order of independent set in $IG(G)$ is $\frac{1}{2}(|G| - 1)$. Here one vertex in $S(G)$ is adjacent with all other vertices of $IG(G)$, thus we require one more color from these colors. Hence,

$$\chi(IG(G)) = \frac{1}{2}(|G| - 1) + 1 = \frac{1}{2}(|G| + 1).$$

□

By combining Theorems 17 and 19, we can easily prove that the invertible graph of any finite group is weakly perfect.

Theorem 21. *For any finite cyclic group G, the graph $IG(G)$ is weakly perfect.*

Proof. It follows directly from Theorems 17 and 19, since $\omega(IG(G)) = \chi(IG(G))$. □

5 Isomorphic properties of IG(G)

In this section, we examine isomorphic properties of invertible graphs of finite groups in detail and determine their important characteristics. We begin with a few examples.

Example 6. *The isomorphic groups and their invertible graphs are traced in Figure 3.*

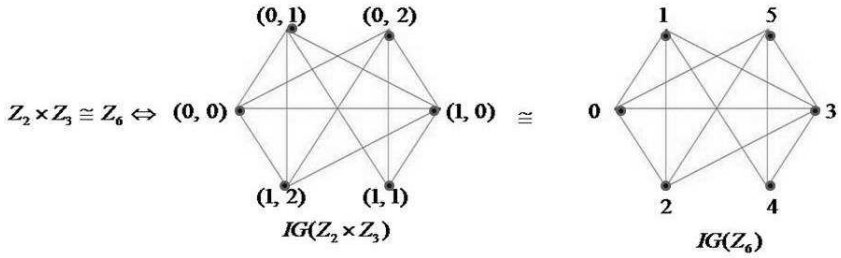


Figure 3. $Z_2 \times Z_3 \cong Z_6 \Leftrightarrow IG(Z_2 \times Z_3) \cong IG(Z_6)$.

Example 7. *Figure 4 shows that groups are not isomorphic and their invertible graphs are also not isomorphic.*

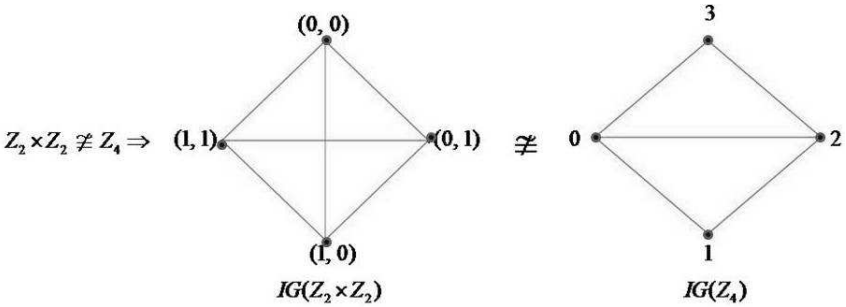


Figure 4. $Z_2 \times Z_2 \ncong Z_4 \Leftrightarrow IG(Z_2 \times Z_2) \ncong IG(Z_4)$.

Example 8. *Figure 5 shows that groups are not isomorphic but their invertible graphs are isomorphic. Consider the cyclic group* $G = \{I, A, B, C, D, E, F, G, H\}$ *with respect to addition modulo* 9 *and* $G' = \{I', A', B', C', D', E', F', G', H'\}$ *is an abelian but not cyclic group with respect to addition modulo 3, where* $A = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}, B = \begin{bmatrix} 2 & -2 \\ 2 & -2 \end{bmatrix}, C = \begin{bmatrix} 3 & -3 \\ 3 & -3 \end{bmatrix}, D = \begin{bmatrix} 4 & -4 \\ 4 & -4 \end{bmatrix}, E = \begin{bmatrix} 5 & -5 \\ 5 & -5 \end{bmatrix}, F = \begin{bmatrix} 6 & -6 \\ 6 & -6 \end{bmatrix},$

$G = \begin{bmatrix} 7 & -7 \\ 7 & -7 \end{bmatrix}, H = \begin{bmatrix} 8 & -8 \\ 8 & -8 \end{bmatrix}, I' = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, A' = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}, \varphi : G \to G', C' = \begin{bmatrix} x+1 & -(x+1) \\ x+1 & -(x+1) \end{bmatrix}, D' = \begin{bmatrix} 2x+1 & -(2x+1) \\ 2x+1 & -(2x+1) \end{bmatrix}, E' = \begin{bmatrix} x+2 & -(x+2) \\ x+2 & -(x+2) \end{bmatrix}, F' = \begin{bmatrix} 2x+2 & -(2x+2) \\ 2x+2 & -(2x+2) \end{bmatrix}, G' = \begin{bmatrix} 2x & -2x \\ 2x & -2x \end{bmatrix}, H' = \begin{bmatrix} 2 & -2 \\ 2 & -2 \end{bmatrix}$ *and* x *is an indeterminate over* Z_3. *Under the mapping* $\varphi : G \to G'$ *such that* $\varphi(I) = I'$, $\varphi(A) = A'$, $\varphi(B) = B'$, $\varphi(C) = C'$, $\varphi(D) = D'$, $\varphi(E) = E'$, $\varphi(F) = F'$, $\varphi(G) = G'$, $\varphi(H) = H'$. *Hence, the fact that* G *is not isomorphic to* G' *implies that* $IG(G) \cong IG(G')$.

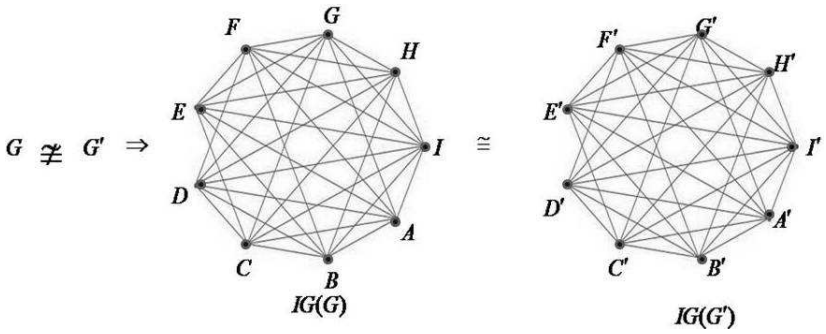


Figure 5. $G \ncong G' \Rightarrow IG(G) \cong IG(G')$.

As the above examples suggest, the invertible graphs of isomorphic groups are isomorphic but converse need not to be true. So, the next theorem completely characterises all isomorphic invariable graphs.

Theorem 22. *Let* G *and* G' *be finite groups. If* $G \cong G'$, *then* $IG(G) \cong IG(G')$. *But the converse is not true.*

Proof. Suppose that $G \cong G'$. Then there is a group isomorphism f from G onto G' such that $f(a) = a'$, for every element a in G and a' in G'. Now, define a map φ from $IG(G)$ to $IG(G')$ by the relation $\varphi(a) = f(a)$, for every vertex $a \in G$. By Remark 1, φ is a bijection. Now let us prove that φ preserves adjacency. For this let $ab \neq e$, then

$f(ab) \neq f(e)$. That implies $f(a)f(b) \neq f(e)$. That is, $\varphi(a)\varphi(b) \neq e^{'}$. So the vertex $\varphi(a)$ is adjacent to the vertex $\varphi(b)$ in $IG(G^{'})$. Similarly, if a is not adjacent to b in $IG(G)$, then $\varphi(a)$ is also not-adjacent to $\varphi(b)$ in $IG(G^{'})$. This shows that $IG(G) \cong IG(G^{'})$. The converse of this statement is false, as the Example 8 shows. That is, if $IG(G) \cong IG(G^{'})$, it does not necessarily follow that $G \cong G^{'}$. □

Let G be a finite group. Then an isomorphism from G onto G is called a group automorphism and set of all automorphisms of G is denoted by $Auto(G)$. Further, an isomorphism from a simple graph X to itself is called graph automorphism of X, and the set of all graph automorphisms forms a group under the operation of composition. This group is also denoted by $Auto(X)$ and is called automorphism group of a graph X.

The following result is an analogous result between $Auto(G)$ and $Auto(IG(G))$.

Theorem 23. *If G is a finite group, then $Auto(G) \subseteq Auto(IG(G))$. But the converse is not true.*

Proof. Let $\psi \in Auto(G)$. Then $\psi : G \to G^{'}$ is a group isomorphism from G onto itself. We shall now show that $\psi \in Auto(IG(G))$. Suppose vertices a and b in G are adjacent in $IG(G)$. Then, either $ab \neq e$, or, $ba \neq e$. $\Rightarrow \psi(ab) \neq e$, or, $\psi(ba) \neq e$. $\Rightarrow \psi(a)\psi(b) \neq e$, or, $\psi(b)\psi(a) \neq e$. $\Rightarrow$ The vertex $\psi(a)$ is adjacent to the vertex $\psi(b)$ in $IG(G)$.

This shows that ψ is a graph isomorphism from $IG(G)$ onto itself. It is clear that $\psi \in Auto(IG(G))$. Hence, $Auto(G) \subseteq Auto(IG(G))$. But, the converse of this result is not true. For this we consider the group $Z_5 = \{0, 1, 2, 3, 4\}$ with respect to addition modulo 5. Define a map $\psi : Z_5 \to Z_5$ by $\psi(0) = 0$, $\psi(1) = 2$, $\psi(2) = 3$ and $\psi(3) = 4$. It is clear that $Auto(Z_5) \subseteq IG(Z_5)$. But $\psi(1 \oplus_5 2) = \psi(Z_3) = 4$ and $\psi(1) \oplus_5 \psi(2) = 2 \oplus_5 3 = 0$. Therefore, $\psi(1 \oplus_5 2) \neq \psi(1) \oplus_5 \psi(2)$ so that ψ is not a homomorphism of group Z_5. □

Acknowledgements

The authors wish to express their cordial thanks to Prof. L. Nagamuni Reddy and Prof. S. Vijaya Kumar Varma for valuable suggestions to improve the presentation of this paper.

References

[1] Xuan Long Ma, Hua Quan Wei, and Guo Zhong, "The cyclic graph of a finite group," *Algebra*, vol. 2013, pp. 1–7, 2013, Art.ID. 107265. Available: http://dx.doi.org/10.1155/2013/107265.

[2] T. Chalapathi and R.V.M.S.S. Kiran Kumar, "The structural properties of even free graphs," *Int. J. Comput. Applications*, vol. 121, no. 2, pp. 19–21, July 2015. Available: http://doi.org/10.5120/21512-4476.

[3] Peter J. Cameron and Shamik Ghosh, "The power graph of a finite group," *Discrete Mathematics*, vol. 311, no. 6, pp. 1220–1222, July 2011. Available: https://doi.org/10.1016/j.disc.2010.02.011.

[4] M. Sattanathan and R. Kala, "An introduction to order prime graph," *Int. J. Contemp. Math. Sciences*, vol. 4, no. 10, pp. 467–474, Jan. 2009.

[5] Timothy C. Burness and Elisa Covato, "On the prime graph of simple groups," *Bulle. of Aus. Math. Socie.*, vol. 91, no. 2, pp. 227–240, Apr. 2015. Available: https://doi.org/10.1017/S0004972714000707.

[6] Jonathan K. Hodge, Steven Schlicker, and Ted Sundstrom, *Abstract Algebra: An Inquiry Based Approach* (Textbooks in Mathematics), CRC Press, 2014, 595 p. ISBN-13: 978-1466567061, ISBN-10: 1466567066.

[7] Norman Biggs, *Algebraic graph theory* (Cambridge Mathematical Library), 2nd ed., Cambridge, USA: Cambridge Univ. Press, 1993, 216 p. ISBN-10: 0521458978, ISBN-13: 978-0521458979.

[8] Vitaly I. Voloshin, *Introduction to graph theory*, New York: Nova Sci. Publishers. Inc, 2009, 144 p. ISBN-10: 1606923749, ISBN-13: 978-1606923740.

[9] *Encyclopedic Dictionary of Mathematics: The Mathematical Society of Japan*, 2nd ed., vol. 1, Kiyosi Ito, Ed., The MIT Press, 2000, 2168 p. ISBN-10: 0262590204, ISBN-13: 978-0262590204.

[10] Vivek Sahai and Vikas Bist, *Algebra*, 3rd ed., Oxford, United Kingdom: Alpha Science International Ltd, 2008, 347 p. ISBN-10: 1842654063, ISBN-13: 9781842654064.

T. Chalapathi, R. V. M. S. S. Kiran Kumar Received April 27, 2018

T. Chalapathi

Department of Mathematics, Sree Vidyanikethan Eng. College,
Tirupati-517502,
Andhra Pradesh, India
Phone: +91 9542865332
E–mail: chalapathi.tekuri@gmail.com

R. V. M. S. S. Kiran Kumar,

Department of Mathematics, S.V. University,
Tirupati-517502,
Andhra Pradesh, India
Phone: +91 9493523133
E–mail: kksaisiva@gmail.com

Computer Science Journal of Moldova, vol.**26**, no.2(77), 2018

Degree subtraction eigenvalues and energy of graphs

H. S. Ramane, K. C. Nandeesh, G. A. Gudodagi, B. Zhou

Abstract

The degree subtraction matrix $DS(G)$ of a graph G is introduced, whose (j,k)-th entry is $d_G(v_j) - d_G(v_k)$, where $d_G(v_j)$ is the degree of a vertex v_j in G. If G is a non-regular graph, then $DS(G)$ has exactly two nonzero eigenvalues which are purely imaginary. Eigenvalues of the degree subtraction matrices of a graph and of its complement are the same. The degree subtraction energy of G is defined as the sum of absolute values of eigenvalues of $DS(G)$ and we express it in terms of the first Zagreb index.

Keywords: Degree of a vertex, degree subtraction matrix, eigenvalues, energy, first Zagreb index.

1 Introduction

In the study of spectral graph theory, we use the spectrum of certain matrices associated with the graph, such as the adjacency matrix, Laplacian matrix and other related matrices. Some useful information about the graph can be obtained from the spectrum of these various matrices [4], [5].

The ordinary energy of a graph G is defined as the sum of the absolute values of the eigenvalues of its adjacency matrix [10]. It is closely related with the total π-electron energy of molecules [13]. This motivates the researchers to introduce different matrices associated with the graph and study the various energies. Several graph energies, such as, Laplacian energy [15], distance energy [16], Randić energy [8], [17], skew

energy [1], [20], incidence energy [8], degree sum energy [21], distance-based energies [9], [19], [22] etc. have been introduced to study the properties of graphs.

In this paper we introduce the degree subtraction matrix of a graph and study the eigenvalues and energy, related to this matrix.

Let G be a simple graph without loops and multiple edges on n vertices and m edges. Let $V(G) = \{v_1, v_2, \ldots, v_n\}$ be the vertex set and $E(G)$ be the edge set of G. The edge between the vertices u and v is denoted by uv. The *degree* of a vertex v_j in G is the number of edges incident to it and is denoted by $d_j = d_G(v_j)$. If the degrees of all vertices of a graph are the same, then the graph is called a *regular graph*. The *degree subtraction matrix* (DS-matrix) of a graph G is a square matrix of order n, defined as $DS(G) = [d_{jk}]$, where

$$d_{jk} = \begin{cases} d_G(v_j) - d_G(v_k) & \text{if } \ j \neq k \\ 0 & \text{if } \ j = k. \end{cases}$$

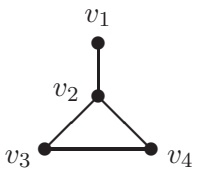
Then *DS-polynomial* of a graph G is the characteristic polynomial of degree subtraction matrix of G and is donoted by $\phi(G : \eta)$. That is $\phi(G : \eta) = \det(\eta I_n - DS(G))$, where I_n is an identity matrix of order n. The roots of the equation $\phi(G : \eta) = 0$ are called the *DS-eigenvalues* of G and they are labeled as $\eta_1, \eta_2, \ldots, \eta_n$. Since $DS(G)$ is a skew symmetric matrix, its eigenvalues are purely imaginary or zero. Two graphs are said to be *DS-cospectral* if they have the same DS-eigenvalues. The *DS-energy* of a graph G, denoted by $E_{DS}(G)$ is difined as

$$E_{DS}(G) = \sum_{j=1}^{n} |\eta_j|. \tag{1}$$

The Eq. (1) is in full analogy with the *ordinary graph energy* defined as [10]

$$E_{\pi}(G) = \sum_{j=1}^{n} |\lambda_j|,$$

where $\lambda_1, \lambda_2, \ldots, \lambda_n$ are the eigenvalues of the adjacency matrix of G. Details on graph energies can be found in the books [12], [18] and the references cited therein.



$$DS(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 0 & -2 & -1 & -1 \\ 2 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \end{array}$$

Figure 1. Graph and its DS-matrix

The DS-polynomial of a graph given in Fig. 1 is $\phi(G : \eta) = \eta^4 + 8\eta^2$ and the DS-eigenvalues are $\mathbf{i}2\sqrt{2}$, 0, 0, $-\mathbf{i}2\sqrt{2}$, where $\mathbf{i} = \sqrt{-1}$. Therefore, $E_{DS}(G) = 4\sqrt{2}$.

The *first Zagreb index* is defined as [14]:

$$M_1 = M_1(G) = \sum_{u \in V(G)} [d_G(u)]^2 = \sum_{uv \in E(G)} [d_G(u) + d_G(v)] .$$

The first Zagreb index is one of the most studied degree-based topological index. For details, see the recent surveys [2], [3] and the references cited therein.

2 DS-eigenvalues

We need the following Lemma.

Lemma 1. [5] *If Q is a nonsingular square matrix, then*

$$\begin{vmatrix} M & N \\ P & Q \end{vmatrix} = |Q||M - NQ^{-1}P|.$$

Theorem 1. *Let G be a graph having n vertices, m edges and first Zagreb index $M_1(G)$. Then the DS-polynomial of G is*

$$\phi(G : \eta) = \eta^n + (nM_1(G) - 4m^2)\eta^{n-2}. \tag{2}$$

Proof. Let $v_1, v_2, \ldots, v_n$ be the vertices of G and let $d_G(v_j) = d_j$ be the degree of a vertex v_j in G, $j = 1, 2, \ldots, n$. Then the characteristic polynomial of DS-matrix of G is

$$\begin{aligned}\phi(G:\eta) &= \det(\eta I - DS(G)) \\ &= \begin{vmatrix} \eta & -d_1+d_2 & -d_1+d_3 & \cdots & -d_1+d_n \\ -d_2+d_1 & \eta & -d_2+d_3 & \cdots & -d_2+d_n \\ -d_3+d_1 & -d_3+d_2 & \eta & \cdots & -d_3+d_n \\ \vdots & & & \vdots & \\ -d_n+d_1 & -d_n+d_2 & -d_n+d_3 & \cdots & \eta \end{vmatrix}. \quad (3)\end{aligned}$$

Subtract the first row from the rows $2, 3, \ldots, n$ of (3) to obtain (4).

$$\begin{vmatrix} \eta & -d_1+d_2 & -d_1+d_3 & \cdots & -d_1+d_n \\ -d_2+d_1-\eta & \eta+d_1-d_2 & -d_2+d_1 & \cdots & -d_2+d_1 \\ -d_3+d_1-\eta & -d_3+d_1 & \eta+d_1-d_3 & \cdots & -d_3+d_1 \\ \vdots & & & \vdots & \\ -d_n+d_1-\eta & -d_n+d_1 & -d_n+d_1 & \cdots & \eta+d_1-d_n \end{vmatrix}. \quad (4)$$

Subtract the first column from columns $2, 3, \ldots, n$ of (4) to obtain (5).

$$\begin{vmatrix} \eta & -d_1+d_2-\eta & -d_1+d_3-\eta & \cdots & -d_1+d_n-\eta \\ -d_2+d_1-\eta & 2\eta & \eta & \cdots & \eta \\ -d_3+d_1-\eta & \eta & 2\eta & \cdots & \eta \\ \vdots & & & \vdots & \\ -d_n+d_1-\eta & \eta & \eta & \cdots & 2\eta \end{vmatrix}. \quad (5)$$

Subtract the second column from columns $3, 4, \ldots, n$ in (5) to obtain (6).

$$\begin{vmatrix} \eta & -d_1+d_2-\eta & d_3-d_2 & \cdots & d_n-d_2 \\ -d_2+d_1-\eta & 2\eta & -\eta & \cdots & -\eta \\ -d_3+d_1-\eta & \eta & \eta & \cdots & 0 \\ \vdots & & & \vdots & \\ -d_n+d_1-\eta & \eta & 0 & \cdots & \eta \end{vmatrix}. \quad (6)$$

Add rows $3, 4, \ldots, n$ to the second row in (6) to obtain (7).

$$\begin{vmatrix} \eta & -d_1+d_2-\eta & d_3-d_2 & \cdots & d_n-d_2 \\ -2m+nd_1-(n-1)\eta & n\eta & 0 & \cdots & 0 \\ -d_3+d_1-\eta & \eta & \eta & \cdots & 0 \\ \vdots & & & \vdots & \\ -d_n+d_1-\eta & \eta & 0 & \cdots & \eta \end{vmatrix} \quad (7)$$

$$= \begin{vmatrix} M & N \\ P & Q \end{vmatrix},$$

where

$$M = \begin{bmatrix} \eta & -d_1+d_2-\eta \\ -2m+nd_1-(n-1)\eta & n\eta \end{bmatrix}_{2\times 2},$$

$$N = \begin{bmatrix} d_3-d_2 & d_4-d_2 & \cdots & d_n-d_2 \\ 0 & 0 & \cdots & 0 \end{bmatrix}_{2\times(n-2)},$$

$$P = \begin{bmatrix} -d_3+d_1-\eta & \eta \\ -d_4+d_1-\eta & \eta \\ \vdots & \vdots \\ -d_n+d_1-\eta & \eta \end{bmatrix}_{(n-2)\times 2} \quad \text{and}$$

$$Q = \begin{bmatrix} \eta & 0 & \cdots & 0 \\ 0 & \eta & \cdots & 0 \\ \vdots & & \vdots & \\ 0 & 0 & \cdots & \eta \end{bmatrix}_{(n-2)\times(n-2)}.$$

By Lemma 1, and taking into account that $\sum_{j=1}^{n} d_j = 2m$ and noting that

$$X = -M_1(G) + 2m(d_1+d_2) - nd_1d_2 - 2m\eta + d_1\eta + d_2\eta + (n-2)d_2\eta$$

and $Y = (2m-d_1-d_2)\eta - (n-2)d_2\eta$, the Eq. (7) reduces to

$$\begin{aligned}
& \phi(G:\eta) \\
= \; & \eta^{n-2}\left|\begin{bmatrix} \eta & -d_1+d_2-\eta \\ -2m+nd_1-(n-1)\eta & n\eta \end{bmatrix} - N\frac{1}{\eta}I_{n-2}P\right| \\
= \; & \eta^{n-2}\left|\begin{bmatrix} \eta & -d_1+d_2-\eta \\ -2m+nd_1-(n-1)\eta & n\eta \end{bmatrix} - \frac{1}{\eta}\begin{bmatrix} X & Y \\ 0 & 0 \end{bmatrix}\right| \\
= \; & \eta^n + (nM_1(G)-4m^2)\eta^{n-2}.
\end{aligned}$$

□

Corollary 1. *Let G be a regular graph on n vertices. Then the DS-polynomial of G is*

$$\phi(G:\eta) = \eta^n.$$

By Theorem 1 we have the following result.

Theorem 2. *Let G be a graph having n vertices, m edges and first Zagreb index $M_1(G)$. Then the DS-eigenvalues of G are 0 $(n-2$ times$)$ and $\pm\mathbf{i}\sqrt{nM_1(G)-4m^2}$, where $\mathbf{i}=\sqrt{-1}$.*

By Theorem 2 we observe that, if G is a non-regular graph, then it has exactly two non-zero DS-eigenvalues.

In the following, $\overline{G}$ denotes the complement graph of G.

Theorem 3. *If η_j, $j=1,2,\ldots,n$ are the DS-eigenvalues of G, then $-\eta_j$, $j=1,2,\ldots,n$ are the DS-eigenvalues of $\overline{G}$.*

Proof. For any vertex u of a graph G of order n, $d_{\overline{G}}(u) = n-1-d_G(u)$. This implies that $DS(\overline{G}) = -DS(G)$. Consequently, if the DS-eigenvalues of a graph G are η_j, $j=1,2,\ldots,n$, then the DS-eigenvalues of $\overline{G}$ are $-\eta_j$, $j=1,2,\ldots,n$. □

Theorem 4. *For any graph G, $\phi(\overline{G}:\eta) = \phi(G:\eta)$.*

Proof. Case 1: If G is regular, then $\overline{G}$ is also regular. Therefore by the Corollary 1, $\phi(\overline{G} : \eta) = \phi(G : \eta)$.
Case 2: If G is non-regular, then $\overline{G}$ also non-regular. Therefore by Theorem 2, both G and $\overline{G}$ have exactly two non-zero DS-eigenvalues. Further by Theorem 3, if η_j, $j = 1, 2, \ldots, n$ are the DS-eigenvalues of G, then $-\eta_j$, $j = 1, 2, \ldots, n$ are the DS-eigenvalues of $\overline{G}$. Also the sum of all DS-eigenvalues is zero. This implies that $\phi(\overline{G} : \eta) = \phi(G : \eta)$. □

Theorem 5. *Let G be a graph on n vertices and m edges. Let δ and Δ be the minimum and maximum vertex degrees of G respectively. Then for $j = 1, 2, \ldots, n$,*

$$n\delta - 2m \leq \eta_j \leq n\Delta - 2m.$$

Proof. If the vertices of G are labeled as $v_1, v_2, \ldots, v_n$, then the sum of the elements of j-th row in DS-matrix is $nd_j - 2m$, where $d_j = d_G(v_j)$. It is well known that the eigenvalues of any matrix lie between the minimum row sum and maximum row sum. Hence

$$\min\{nd_j - 2m\} \leq \eta_j \leq \max\{nd_j - 2m\}.$$

This implies

$$n\delta - 2m \leq \eta_j \leq n\Delta - 2m.$$

□

By Theorem 4, G and $\overline{G}$ are DS-cospectral graphs. By Theorem 2, if G_1 and G_2 are two different graphs having the same number of vertices and the same number of edges, and if $M_1(G_1) = M_1(G_2)$, then G_1 and G_2 are DS-cospectral.

3 DS-Energy

From Theorem 2 and by the definition of DS-energy via Eq. (1), we have the following theorem.

Theorem 6. *Let G be a graph having n vertices, m edges and first Zagreb index $M_1(G)$. Then*

$$E_{DS}(G) = 2\sqrt{nM_1(G) - 4m^2}. \tag{8}$$

By Theorem 3, we have the following corollary.

Corollary 2. *For any graph G, $E_{DS}(G) = E_{DS}(\overline{G})$.*

If G is a regular graph, then by the Corollary 1, $E_{DS}(G) = 0$.

For fixed n and m, the Eq. (8) depends on the nature of $M_1(G)$.

Theorem 7. *Let G_1 and G_2 be two different graphs having equal number of vertices and equal number of edges.*
(i) *If $M_1(G_1) > M_1(G_2)$, then $E_{DS}(G_1) > E_{DS}(G_2)$.*
(ii) *If $M_1(G_1) = M_1(G_2)$, then $E_{DS}(G_1) = E_{DS}(G_2)$.*

Let $p = \lfloor 2m/n \rfloor$. Then the first Zagreb index $M_1(G)$ satisfies the inequalities:

$$2(2p+1)m - p(p+1)n \leq M_1(G) \leq m\left(\frac{2m}{n-1} + n - 2\right). \tag{9}$$

The right hand side of the Eq. (9) is due to de Caen [7], whereas the left hand side inequality is due to Das [6].

Using Eqs. (8) and (9) we have the following result.

Theorem 8. *Let G be a graph with n vertices and m edges. Let $p = \lfloor 2m/n \rfloor$. Then*

$$\begin{aligned} & 2\sqrt{2(2p+1)mn - p(p+1)n^2 - 4m^2} \\ \leq\ & E_{DS}(G) \\ \leq\ & 2\sqrt{mn\left(\frac{2m}{n-1} + n - 2\right) - 4m^2}. \end{aligned}$$

Let S_n be the star and P_n be the path on n vertices. Among all n-vertex trees, the star S_n has maximum value and the path P_n has minimum value of the first Zagreb index [11]. If T_n is an n-vertex tree, different from the star and path, then $M_1(S_n) > M_1(T_n) > M_1(P_n)$. Using this result and Eq. (8) we have the following result.

Theorem 9. *If T_n is an n-vertex tree, different from the star S_n and path P_n, then $E_{DS}(S_n) > E_{DS}(T_n) > E_{DS}(P_n)$.*

If u and v are the adjacent vertices of G, then $G - uv$ is the graph obtained from G by removing the edge uv. If u and v are non-adjacent vertices of G, then the graph $G + uv$ is obtained from G by adding an edge uv.

Theorem 10. *Let G be a graph having n vertices, m edges and first Zagreb index $M_1(G)$. Let u, v, w be three distinct vertices of G such that u is adjacent to v and u is not adjacent to w. Let $H = G - uv + uw$. Then*

$$E_{DS}(H) = 2\sqrt{nM_1(G) - 4m^2 + 2n[d_G(w) - d_G(v) + 1]}. \qquad (10)$$

Further,
(i) *if $d_G(w) - d_G(v) + 1 = 0$, then $E_{DS}(H) = E_{DS}(G)$*
(ii) *if $d_G(w) - d_G(v) + 1 > 0$, then $E_{DS}(H) > E_{DS}(G)$*
(iii) *if $d_G(w) - d_G(v) + 1 < 0$, then $E_{DS}(H) < E_{DS}(G)$.*

Proof. Let $d_1, d_2, \ldots, d_n$ be the degrees of the vertices of G. Without loss of generality, let $d_G(u) = d_1$, $d_G(v) = d_2$ and $d_G(w) = d_3$. Therefore, $d_H(u) = d_1$, $d_H(v) = d_2 - 1$ and $d_H(w) = d_3 + 1$. Hence by Theorem 6,

$$\begin{aligned} E_{DS}(H) &= 2\sqrt{nM_1(H) - 4m^2} \\ &= 2\sqrt{n\left[d_1^2 + (d_2 - 1)^2 + (d_3 + 1)^2 + \sum_{j=4}^{n}(d_G(v_j))^2\right] - 4m^2} \\ &= 2\sqrt{nM_1(G) - 4m^2 + 2n(d_3 - d_2 + 1)} \\ &= 2\sqrt{nM_1(G) - 4m^2 + 2n[d_G(w) - d_G(v) + 1]}. \end{aligned}$$

The results (i), (ii), and (iii) follow from Eqs. (8) and (10). □

Proof of the Theorem 11 is analogous to that of Theorem 10.

Theorem 11. *Let G be a graph having n vertices, m edges and first Zagreb index $M_1(G)$. Let u, v, w, x be the four distinct vertices of G such that u is adjacent to v and w is not adjacent to x. Let $H = G - uv + wx$. Then*

$$E_{DS}(H) = 2\sqrt{nM_1(G) - 4m^2 + 2n[d_G(w) + d_G(x) - d_G(u) - d_G(v) + 2]}. \tag{11}$$

Further,
(i) *if $d_G(w) + d_G(x) - d_G(u) - d_G(v) + 2 = 0$, then $E_{DS}(H) = E_{DS}(G)$*
(ii) *if $d_G(w) + d_G(x) - d_G(u) - d_G(v) + 2 > 0$, then $E_{DS}(H) > E_{DS}(G)$*
(iii) *if $d_G(w) + d_G(x) - d_G(u) - d_G(v) + 2 < 0$, then $E_{DS}(H) < E_{DS}(G)$.*

Theorem 12. *Let G be a graph having n vertices $v_1, v_2, \ldots, v_n$ and m edges and first Zagreb index $M_1(G)$. Let G' be the subgraph of G on $k \geq 1$ vertices $v_1, v_2, \ldots, v_k$, and m' edges. Let H be the graph obtained from G by removing the edges of G'. Then*

$$E_{DS}(H) = 2\sqrt{\begin{array}{l} nM_1(G) - 4m^2 - 2n\sum_{j=1}^{k}[d_G(v_j)d_{G'}(v_j)] \\ +nM_1(G') + 4m'(2m - m') \end{array}}. \tag{12}$$

Further,
(i) *if $-2n\sum_{j=1}^{k}[(d_G(v_j))(d_{G'}(v_j))] + nM_1(G') + 4m'(2m - m') = 0$, then $E_{DS}(H) = E_{DS}(G)$*
(ii) *if $-2n\sum_{j=1}^{k}[(d_G(v_j))(d_{G'}(v_j))] + nM_1(G') + 4m'(2m - m') > 0$, then $E_{DS}(H) > E_{DS}(G)$*
(iii) *if $-2n\sum_{j=1}^{k}[(d_G(v_j))(d_{G'}(v_j))] + nM_1(G') + 4m'(2m - m') < 0$, then $E_{DS}(H) < E_{DS}(G)$.*

Proof. Let $v_1, v_2, \ldots, v_k$ be the vertices of a subgraph G' of G, $k \geq 1$. Therefore $d_H(v_j) = d_G(v_j) - d_{G'}(v_j)$, for $j = 1, 2, \ldots, k$ and $d_H(v_j) = d_G(v_j)$, for $j = k+1, k+2, \ldots, n$. Also, if m' is the number of edges of G', then H has $m - m'$ edges. By Eq. (8)

$$\begin{aligned}
E_{DS}(H) &= 2\sqrt{nM_1(H) - 4(m-m')^2} \\
&= 2\sqrt{\begin{array}{l} n\left[\sum_{j=1}^{k}(d_G(v_j) - d_{G'}(v_j))^2 + \sum_{j=k+1}^{n}[d_G(v_j)]^2\right] \\ -4(m-m')^2 \end{array}} \\
&= 2\sqrt{\begin{array}{l} n\left[\sum_{j=1}^{n}[d_G(v_j)]^2 - 2\sum_{j=1}^{k}[d_G(v_j)d_{G'}(v_j)]\right. \\ \left. + \sum_{j=1}^{k}[d_{G'}(v_j)]^2\right] - 4(m^2 - 2mm' + m'^2) \end{array}} \\
&= 2\sqrt{\begin{array}{l} nM_1(G) - 4m^2 - 2n\sum_{j=1}^{k}[(d_G(v_j))(d_{G'}(v_j))] \\ +nM_1(G') + 4m'(2m - m') \end{array}}.
\end{aligned}$$

The results (i), (ii), and (iii) follow from the Eqs. (8) and (12). □

Corollary 3. *Let G be a graph having n vertices $v_1, v_2, \ldots, v_n$ and m edges and first Zagreb index $M_1(G)$. Let $e_1, e_2, \ldots, e_k$ be the k independent edges of G, $1 \le k \le \lfloor n/2 \rfloor$, where $e_j = v_{2j-1}v_{2j}$, $j = 1, 2, \ldots, k$. Let H be the graph obtained from G by removing its k independent edges e_j, $j = 1, 2, \ldots, k$. Then*

$$E_{DS}(H) = 2\sqrt{nM_1(G) - 4m^2 + 2n\left(k - \sum_{j=1}^{2k} d_G(v_j)\right) + 4k(2m-k)}.$$

Further,

(i) *if* $2n\left(k - \sum_{j=1}^{2k} d_G(v_j)\right) + 4k(2m-k) = 0$, *then* $E_{DS}(H) = E_{DS}(G)$

(ii) *if* $2n\left(k - \sum_{j=1}^{2k} d_G(v_j)\right) + 4k(2m-k) > 0$, *then* $E_{DS}(H) > E_{DS}(G)$

(iii) *if* $2n\left(k-\sum_{j=1}^{2k} d_G(v_j)\right)+4k(2m-k)<0$, *then* $E_{DS}(H)<E_{DS}(G)$
(iv) *if* $n=2k$, *then* $E_{DS}(H)=E_{DS}(G)$.

Proof. Follows from Theorem 12 by taking $G'=kK_2$, a k-matching. □

Corollary 4. *Let G be a graph having n vertices $v_1,v_2,\ldots,v_n$ and m edges and first Zagreb index $M_1(G)$. Let $V_k=\{v_1,v_2,\ldots,v_k\}$ be a k-element subset of the vertex set of the graph G, $k\geq 2$ such that every pair of vertices of V_k is adjacent in G. Let H be the graph obtained from G by deleting all the edges connecting pairs of vertices from V_k. Then*

$$E_{DS}(H)=2\sqrt{\begin{array}{l} nM_1(G)-4m^2+2(k-1)\left[2mk-n\sum_{j=1}^{k} d_G(v_j)\right] \\ +k(n-k)(k-1)^2 \end{array}}.$$

Further,
(i) *if* $2(k-1)\left[2mk-n\sum_{j=1}^{k} d_G(v_j)\right]+k(n-k)(k-1)^2=0$, *then* $E_{DS}(H)=E_{DS}(G)$
(ii) *if* $2(k-1)\left[2mk-n\sum_{j=1}^{k} d_G(v_j)\right]+k(n-k)(k-1)^2>0$, *then* $E_{DS}(H)>E_{DS}(G)$
(iii) *if* $2(k-1)\left[2mk-n\sum_{j=1}^{k} d_G(v_j)\right]+k(n-k)(k-1)^2<0$, *then* $E_{DS}(H)<E_{DS}(G)$
(iv) *if* $n=k$, *then* $E_{DS}(H)=E_{DS}(G)$.

Proof. Follows from Theorem 12 by taking $G'=K_k$, a complete graph on k vertices. □

Corollary 5. *Let G be a graph having n vertices $v_1,v_2,\ldots,v_n$ and m edges and first Zagreb index $M_1(G)$. Let C_k be the cycle of G, where*

the vertices $v_1, v_2, \ldots, v_k$ are on C_k, $k \geq 1$. Let H be the graph obtained from G by removing the edges of C_k. Then

$$E_{DS}(H) = 2\sqrt{nM_1(G) - 4m^2 + 4n\left[k - \sum_{j=1}^{k} d_G(v_j)\right] + 4k(2m-k)}.$$

Further,

(i) *if* $4n\left[k - \sum_{j=1}^{k} d_G(v_j)\right] + 4k(2m-k) = 0$, *then* $E_{DS}(H) = E_{DS}(G)$

(ii) *if* $4n\left[k - \sum_{j=1}^{k} d_G(v_j)\right] + 4k(2m-k) > 0$, *then* $E_{DS}(H) > E_{DS}(G)$

(iii) *if* $4n\left[k - \sum_{j=1}^{k} d_G(v_j)\right] + 4k(2m-k) < 0$, *then* $E_{DS}(H) < E_{DS}(G)$

(iv) *if* $k = n$, *then* $E_{DS}(H) = E_{DS}(G)$.

Proof. Follows from Theorem 12 by taking $G' = C_k$, a cycle on k vertices. □

Algorithm: Computation of DS-energy using adjacency matrix.

1. **Start**
2. **Declare:** **A**$[n][n]$, d$[n]$, r, s, n, m, $N = 0$, $S = 0$ as integers.
3. **Declare:** *Result* as floating point.
   Read n, **A**$[r][s]$
4. Compute the degree of each vertex
   for $r = 1$ to n increment by 1
   d$[r] \longleftarrow 0$
   for $s = 1$ to n increment by 1
   d$[r] \longleftarrow$ d$[r]$ + **A**$[r][s]$
   Display: Degree of vertex d$[r]$
   The square of degree of a vertex, d$[r]$ * d$[r]$

5. **Compute:** The sum of each row
   $S \longleftarrow S + \mathrm{d}[r]$.
   Sum of squares of each row sum as $N = N + \mathrm{d}[r] * \mathrm{d}[r]$
   and number of edges $m = S/2$.
6. **Display:** Sum of squares of each row sum N and number of edges m.
7. **Compute the Result**
   $Result = 2 * \mathrm{sqrt}(n * N - 4 * m * m)$
8. **Display:** the *Result*.
9. **Stop**

Terms:

n - Total number of vertices in a given graph.

m - Total number of edges in a given graph.

A - Adjacency Matrix.

d - Degree of a vertex.

In the above algorithm, the outer loop iterates r times and the inner loop iterates s times. Hence the statements inside the inner loop will be executed rs times. This means that the outer and inner loops are dependent on problem size n. Hence the time complexity of the algorithm is $O(n^2)$.

Acknowledgement: Authors are thankful to referees for giving useful suggestions. H. S. Ramane and K. C. Nandeesh are thankful to the University Grants Commission (UGC), New Delhi for support through research grant under UPE FAR-II grant No. F 14-3/2012 (NS/PE). G. A. Gudodagi is thankful to the Karnatak University, Dharwad for support through UGC-UPE (Non NET) scholarship No. KU/SCH/UGC-UPE/2014-15/901.

References

[1] C. Adiga, R. Balakrishnan, W. So, "The skew energy of a digraph," *Linear Algebra Appl.*, vol. 432, pp. 1825–1835, 2010.

[2] B. Borovicanin, K. C. Das, B. Furtula, I. Gutman, "Bounds for Zagreb indices," *MATCH Commun. Math. Comput. Chem.*, vol. 78, pp. 17–100, 2017.

[3] B. Borovicanin, K. C. Das, B. Furtula, I. Gutman, "Zagreb indices: Bounds and extremal graphs," in *Bounds in Chemical Graph Theory - Basics*, I. Gutman, B. Furtula, K. C. Das, E. Milovanovic, Eds. Kragujevac, Serbia: Univ. Kragujevac, 2017, pp. 67–153.

[4] A. E. Brouwer, W. H. Haemers, *Spectra of Graphs*, Berlin: Springer, 2012.

[5] D. M. Cvetković, M. Doob, H. Sachs, *Spectra of Graphs*, New York, USA: Academic Press, 1980.

[6] K. C. Das, "Sharp bounds for the sum of the squares of the degrees of a graph," *Kragujevac J. Math.*, vol. 25, pp. 31–49, 2003.

[7] D. de Caen, "An upper bound on the sum of squares of degrees in a graph," *Discrete Math.*, vol. 185, pp. 245–248, 1988.

[8] R. Gu, X. Li, J. Liu, "Note on three results on Randić energy and incidence energy," *MATCH Commun. Math. Comput. Chem.*, vol. 73, pp. 61–71, 2015.

[9] A. D. Güngör, A. S. Çevik, "On the Harary energy and Harary Estrada index of a graph," *MATCH Commun. Math. Comput. Chem.*, vol. 64, pp. 280–296, 2010.

[10] I. Gutman, "The energy of a graph," *Ber. Math. Stat. Sekt. Forschungsz. Graz*, vol. 103, pp. 1–22, 1978.

[11] I. Gutman, K. C. Das, "The first Zagreb indices 30 years after," *MATCH Commun. Math. Comput. Chem.*, vol. 50, pp. 83–92, 2004.

[12] I. Gutman, X. Li (Eds.), *Energies of Graphs - Theory and Applications*, Kragujevac, Serbia: Univ. Kragujevac, 2016.

[13] I. Gutman, O. E. Polansky, *Mathematical Concepts in Organic Chemistry*, Berlin: Springer-Verlag, 1986.

[14] I. Gutman, N. Trinajstić, "Graph theory and molecular orbitals, Total π-electron energy of alternant hydrocarbons," *Chem. Phys. Lett.*, vol. 17, pp. 535–538, 1972.

[15] I. Gutman, B. Zhou, "Laplacian energy of a graph," *Linear Algebra Appl.*, vol. 414, pp. 29–37, 2006.

[16] G. Indulal, I. Gutman, A. Vijayakumar, "On distance energy of graphs," *MATCH Commun. Math. Comput. Chem.*, vol. 60, pp. 461–472, 2008.

[17] J. Li, J. M. Guo, W. C. Shiu, "A note on Randić energy," *MATCH Commun. Math. Comput. Chem.*, vol. 74, pp. 389–398, 2015.

[18] X. Li, Y. Shi, I. Gutman, *Graph Energy*, USA, New York: Springer, 2012.

[19] H. S. Ramane, K. C. Nandeesh, "Complementary distance spectra and complementary distance energy of line graphs of regular graphs," *J. Indones. Math. Soc.*, vol. 22, pp. 27–35, 2016.

[20] H. S. Ramane, K. C. Nandeesh, I. Gutman, X. Li, "Skew equienergetic digraphs," *Trans. Comb.*, vol. 5, pp. 15–23, 2016.

[21] H. S. Ramane, D. S. Revankar, J. B. Patil, "Bounds for the degree sum eigenvalue and degree sum energy of a graph," *Int. J. Pure Appl. Math. Sci.*, vol. 6, pp. 161–167, 2013.

[22] H. S. Ramane, A. S. Yalnaik, "Reciprocal complementary distance spectra and reciprocal complementary distance energy of line graphs of regular graphs," *El. J. Graph Theory Appl.*, vol. 3, pp. 228–236, 2015.

H. S. Ramane, K. C. Nandeesh,
G. A. Gudodagi, B. Zhou

Received May 16, 2018

H. S. Ramane
Department of Mathematics,
Karnatak University,
Dharwad - 580003, India
E–mail: hsramane@yahoo.com

K. C. Nandeesh
Department of Mathematics,
Karnatak University,
Dharwad - 580003, India
E–mail: nandeeshkc@yahoo.com

G. A. Gudodagi
Department of Mathematics,
Karnatak University,
Dharwad - 580003, India
E–mail: gouri.gudodagi@gmail.com

B. Zhou
Department of Mathematics,
South China Normal University,
Guangzhou 510631, P. R. China
E–mail: zhoubo@scnu.edu.cn

Computer Science Journal of Moldova, vol.**26**, no.2(77), 2018

Prediction of the Survival of Kidney Transplantation with imbalanced Data Using Intelligent Algorithms

Zeinab Hassani Nasibeh Emami

Abstract

Kidney transplantation is one of the effective post-dialysis treatment methods for patients with chronic renal failure in the world. Most medical data are imbalanced and the output of algorithms is inefficient with imbalanced data. The aim of this study is to predict the two-year survival rate of kidney transplant patients and provide a more accurate model. We evaluate the data of renal transplant patients in Afzalipour Medical Education Center 2006-2010, Kerman, Iran. Survival prediction of kidney transplantation with MLP and RBF neural networks with two methods of sampling and investigating the factors affecting the survival of kidney transplant in renal transplant patients is considered by the binary particle optimization algorithm and nearest neighbor algorithm. Accuracy of the results can be increased by using the oversampling method in imbalanced medical data, and radial base network model is a suitable model for predicting the survival of kidney transplant patients.

Keywords: Neural network, imbalanced data, Binary particle optimization algorithm, nearest neighbor, Kidney transplantation.

1 Introduction

Kidney transplantation is an appropriate and effective strategy for patients with chronic renal failure. The end stage of the renal disease is associated with a significant reduction in the quality of life of patients

and early death. Treatment for chronic renal failure includes two types of dialysis and kidney transplantation. The kidney transplant brings good and preferred treatment and a more favorable life and reduces the risk of death for patients in the final stage of kidney failure [1],[2], and [3].

The history of kidney transplantation dates back to 1954 in Germany that the transplant was given by the living relative. However, the first renal transplant in Iran was carried out in 1967 in Shiraz. Iran has the highest number of kidney transplants. It is in the first place in the Middle East, and it is the fourth in the world [3].

Providing an appropriate transplant is effective for every patient. Therefore, evaluation of renal graft survival after transplantation is very important and several studies have been carried out on graft survival. Researchers in the field of prediction of kidney transplantation survival include statistical studies and artificial intelligence. In 2012, in the United States, a model of body mass index, race, gender, and age were identified as the factors influencing the survival of kidney transplantation using the Bayesian method [5]. In 2012, the support vector machine was used to determine the effective variables of age and level of creatinine, gender, and recipient weight [6]. Nematollahi et al. studied 717 patients of Nemazee Hospital of Shiraz during 2008–2012, and they Predicted Survival of Kidney for 5 years by multilayer perceptron of artificial neural networks (MLP), logistic regression (LR), Support Vector Machine (SVM). Also they identified that SVM and MLP models could efficiently be used for determining survival prediction in kidney transplant recipients [2]. In [4], a model is proposed for Predicticting Graft Survival in Kidney Transplant by decision tree, and Cox regression and Ensemble learners. They revealed that early acute rejection in the first year is associated with a substantially increased risk of graft failure. Machine learning methods may provide versatile and feasible tools for forecasting graft survival.

This article examines the survival of kidney transplantation after two years using the neural network. The ratio of successful data to the total data is 0.09 percent and results in a convergence of data in a neural network class. Therefore, in the pre-processing of data, we

prepare imbalanced data with the sampling method to provide balanced data for the input of the neural network. In the following, we obtain the effective factors in the survival of kidney transplant using binary particle optimization algorithm and the nearest neighbor algorithm.

2 Methods

In this section, technical aspects of machine learning and data mining Methods used for data analysis are described. The data of renal transplant patients were collected from the Afzalipour Educational Center in Kerman and were inconsistent. Therefore, the data are balanced by two sampling methods: under-sampling and oversampling, and then used in the neural network applying particle optimization algorithm and the nearest neighbor algorithm.

2.1 Dataset

The data of renal transplant patients of the Afzalipour Medical Center of Kerman city were collected during the years 2006–2010. Data were collected from 423 cases of kidney transplant patients of the Afzalipour Medical Center of Kerman city. 156 cases of kidney transplant patients with 14 attributes were selected after removing missing values or applying average method for missing values [7].

The dataset contains information on 156 renal transplantation patients with 14 features. Features include Recipients sex, Donors sex, Recipients age, Donors age, BMI, dialysis time before operation, blood group consistency, Recipient and donor of RH, Donors relationship, kidney transplantation history, diabetes of Recipient, donator type (Alive, corpse), Type of dialysis and result of the two-year graft survival. We consider weight and height with BMI. Table 1 shows the features of kidney transplant patients.

Table 1. Description of characteristics of kidney transplant patients

no	**Attributes**
1	Donor's age
2	Recipient's age
3	Donor's sex
4	Recipients sex
5	BMI
6	blood group consistency
7	Recipient and donor of RH
8	dialysis time before operation
9	Donor's relationship
10	kidney transplantation history
11	diabetes of Recipient
12	donator type (Alive, corpse)
13	Type of dialysis
14	result of the two-year graft survival

2.2 Neural network

An artificial neural network is a method of information processing that is inspired by neurological systems. An artificial neural network has the ability to model complex systems and relationships, and non-linear functions [8] and consists of a sum of neurons or the same processor elements. The relationship between the adjustable neurons depends on the conditions of the issue. The neurons of each layer are attached to subsequent layers with different weights.

Information is stored in the weights. The implementation of the neural network has three parts: sample provision, the training phase, and neural network testing. In this study, two types of the artificial neural network have been used: multi-layer perceptron network and radial base function network. We examine functioning of the neural network with input data after completing the training of the neural network and correcting its weight.

2.2.1 Multi-layer perceptron (MLP)

The most common neural network is reversible networks. Multi-layer perceptron network (MLP) is a model of reverse grids that maps the input data to the output data by adjusting the weight of the layers. MLP is a combination of at least three layers of neurons (input, intermediate, and output) that can contain one or more hidden layers [8],[9]. Neurons are arranged in several layers, so that each layer receives its inputs from the previous layer and sends its outputs to the next layer. The neural network is taught by the post-error algorithm [10].

The post-error algorithm is used to learn the weight of a multilayer network. In this method, we try to minimize the square of the error between the network outputs and the target function using the descending gradient. The function of this method is greatly improved if the weights and the number of neurons are selected correctly and optimally. The goal of learning in learning algorithms is to minimize the output error of the network with the optimum output. Learning algorithms include two kinds: observer learning and uncontrollable learning [11].

2.2.2 Radial Basis Function (RBF)

Radial Basis Function is reversible Network with three inputs, hidden and output layers. In the hidden layer, the active function of the radial base function is used. The main advantage of the RBF network is to minimize input data errors. The RBF network approximates any function using symmetric and local radial functions [10]. The active function in the hidden layer is Gaussian function and in the output layer is generally a sigmoid or a linear function [12], [13].

2.3 Balanced data

In imbalanced data, the number of samples in a class is much higher than in the other classes. A class which has a greater number of data is called the class of majority, and the class with less data – a minority class. In the imbalanced data, the main challenge is to identify the

correct lower sample class. Balanced data are used in classification algorithms. Therefore, the imbalanced data classification is not performed correctly, and categorization tends towards larger educational examples that increase the error in identifying a minority class. One of the methods for balancing the imbalanced data is the data level method. In this class of methods, the distribution of a imbalanced class is balanced by re-sampling in the data space [14].

Several approaches have been proposed for imbalanced data, such as sampling, data-level feature selection, and algorithm levels, such as cost-sensitive and single-class learning. Methods of pre-processing data are sampled by under-sampling and oversampling. In the under-sampling, the data are collected by deleting random samples of majority class, and in oversampling, it's collected by copying and adding random samples from the minority class. Oversampling is better than under-sampling [15].

2.4 Particles Swarm Optimization (PSO)

James Kennedy and Russell C. Eberhart are the main owners of the PSO idea [16]. Their first simulation was carried out in 1995. PSO is similar to evolutionary computation techniques like Genetic Algorithms, but does not incorporate any evolution operators[17]. In the PSO algorithm, there are a number of organisms that are referred to as particles and are spread in the space of the search function we intend to optimize. Each particle calculates the value of the objective function in the position of the space in which it is located. Then, using the current location information and the best location in the past, as well as the information of one or more best particles in the collection, it chooses a direction of movement. All particles are directed to move, and after completing the move, one step of the algorithm ends. These steps are repeated several times until the desired result is obtained. The PSO algorithm contains the following steps [18]:

1) Initialize a population of particles with random locations and velocities in dimension D in the search space

2) Calculate the merit function

3) Update the velocity of each particle and move to the next position based on the following relationships:

$$v_{id}(t+1) = w * v_{id}(t) + c_1 * r_1(x_i^{best} - x_{id}^t) + c_2 * r_2(x_g^{best} - x_{id}^t) \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t) \quad (2)$$

$x_{id}(t)$ and $v_{id}(t)$ respectively indicate the position and velocity of the particle in the dimension D, x_i^{best} – the best position of the particle up to the moment t, x_g^{best} – the best global position, w – the coefficient of inertia, c_1 and c_2 – the learning coefficients, and r_1 and r_2 are two random numbers.

4) Stop the algorithm if it reaches a specified stopping point, otherwise go to step 2.

2.4.1 Binary Particles Swarm Optimization (BPSO)

The binary particle optimization algorithm was proposed by James Kennedy and Russell Eberhart for solving binary problems [19]. In the binary version, the position of each particle in each dimension is limited to two values of zero and one. That is, every particle moves in a space that is limited to zero and one. The initial velocity of the particles is also in the interval [1 and 0]. The steps of the algorithm are as follows [20]:

1) Initialize a population of particles with random positions and velocities in dimension D in the search space

2) Calculate the merit function

3) Update the velocity of each particle based on the relationship 1. After calculating the velocity of each particle, it is necessary to check that the velocity of the particle is in the interval

$[v_{min}, v_{max}]$. If the particle velocity is extruded outside the range, we map the resulting velocity of the following interval using the following equation:

$$v_{id}(t+1) = max(min(v_{max}, v_{id}(t+1), v_{min}) \tag{3}$$

In order to update position of each particles, at first transform the velocity vector through as sigmoid limiting function:

$$S(v_{id}(t+1)) = 1/(1+e^{-v_{id}(t+1)}) \tag{4}$$

4) Update the position of each particle in accordance with equation 5.

$$x_{id}(t+1) = \begin{cases} 1 & rand < S(v_{id}(t+1)) \\ 0 & O.W \end{cases} \tag{5}$$

5) If the stop criterion is met, the algorithm stops; otherwise, it goes to step 2.

2.5 The nearest neighbor algorithm

In recognizing the pattern, the nearest neighbor algorithm is a method used for classification and regression. The input of the algorithm is the training samples, and the output is the class label. Examples of training are vectors in the D-dimensional space whose class labels are specified. To classify an unlabeled object, the distance to the tagged samples is calculated, and k of the nearest neighbor is identified and, based on the majority vote, the class label of the object is specified. The most common metric of distance or similarity is to calculate the distance between objects in Euclidean space [18], [21].

2.6 Model Evaluation Indicators

We need indicators that can be used to evaluate the function of models in comparison with the data set in order to evaluate the efficiency of neural network models. The best method for estimating the neural

network is the mean squared error (MSE) and the root mean square error (RMSE) [22] and [23].

$$MSE == \frac{1}{N}\sum_{n=1}^{N}(\widehat{y_n} - y_n)^2 \tag{6}$$

$$RMSE = \sqrt[2]{MSE} \tag{7}$$

$\widehat{y_n}, y_n$ respectively indicate predictions and original data of N instances. In the following, two other important indicators, sensitivity and specificity, are considered for a better understanding of network performance. Thus, sensitivity is the ability to test for the proper diagnosis of people requiring kidney transplantation, and the specificity is the ability to test ability to diagnose people who do not need any transplant [24].

2.7 Proposed Algorithm

In this study, the prediction and identification of the effective features in the survival of the two-year kidney transplantation have been investigated in four stages. The first stage consists of data collection. At this stage, the information about renal transplant patients of the Afzalipour Medical Center of Kerman was collected with 14 characteristics during the years 2006–2011. The second stage is pre-processing the data. We first correct incomplete data with mode and average values. We used mode for discrete values and average for continues values, however, if a record contains more than two attributes with a missing value, it is deleted. Effective results depend on the appropriate data in the model and some values of data attributes are in a different range. Hence, a normalization formula is used to integrate data. Moreover, we normalize the data to improve the accuracy and effectiveness of the results. The normalization formula is as follows:

$$N_i = \frac{x_i - \mu}{\sigma} \tag{8}$$

x_i shows value of data i^{th} and μ, σ respectively indicate average and variance of data.

Due to the collected data are imbalanced, the unbalancing data lead to the convergence of the classification to the majority class, which reduces the efficiency of the classification. Therefore, in the third stage, sampling techniques have been used to balance the data which involves two methods of oversampling and under-sampling. Then in the fourth stage, which is modelled, two structures of the neural network, MLP and RBF, have been used to predict the two-year survival of kidney transplantation. 25% of sample data were considered for testing and 75% of sampling data – for network training. In the neural network MLP, we have perceptron with 13 inputs in the input layer and 6 neurons of the hidden layer and the output layer. In the following, a binary particle optimization algorithm has been used to identify the effective factors in the survival of the kidney transplant. The fitness function of the binary particle optimization algorithm is the nearest neighbor's algorithm accuracy. Initialization parameters involve determining the number of population, Inertia weight, learning factors, and number of iterations that are in this study, their values are 20, 0.48, 2, 200, respectively. At the end, confusion matrix, RMSE and MSE are used to determine the algorithm performance. Figure 1 shows the proposed algorithm.

3 Results

In this study, we predict the survival of renal transplant and identify the variables that affect the survival of kidney transplantation. Algorithms from MLP and RBF neural networks are used to predict the survival of kidney transplant, a binary particle optimization algorithm and the nearest neighbor algorithm are used to identify the variables that affect the survival of the kidney. The primary data for patients with renal transplantation are imbalanced. In order to increase the accuracy, the results of the data are balanced in two methods: oversampling and under-sampling. Balanced data were studied for modeling by intelligent systems. The data of kidney transplant patients in the MATLAB software are examined in two sampling methods in neural network models MLP and RBF.

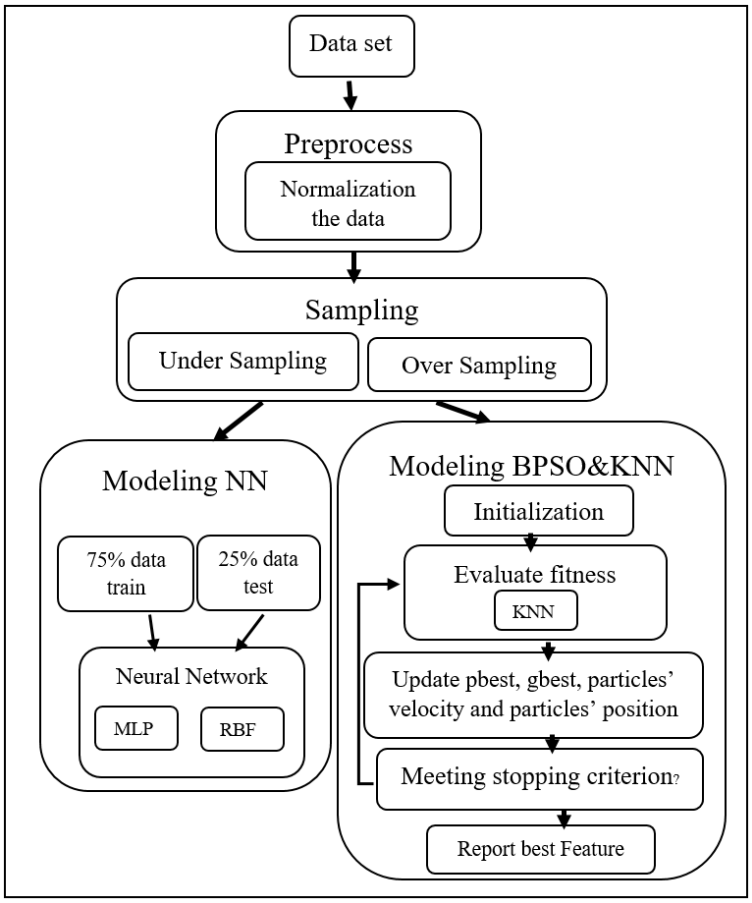


Figure 1. Proposed algorithm

In two sampling methods, we obtained the number of neurons and the appropriate neural network by trial and error method. 25% of sample data was considered for testing and 75% of sampling data for network training. In the neural network model, the laminated perceptron with 13 inputs in the input layer and 6 neurons of the hidden layer and the output layer were used to express the survival or lack of survival kidney transplantation. The results of the MLP and RBF neural networks using two sampling methods are shown in Tables 2 and 3.

In the MLP oversampling method we have accuracy with 93.57 percent, and in RBF – 98.16 percent, and in the MLP under-sampling method – with 81.48 percent and in RBF we have accuracy with 92 percent. The MLP and RBF neural networks oversampling methods were respectively 95.44% and 98.91%, and in the under-sampling MLP and RBF neural networks, the obtained precision was 83.47% and 92.85%, respectively.

In the oversampling, the sensitivity and specificity of the MLP network are respectively 93.6% and 91.4%, and for the RBF network they are respectively 98.2% and 97.4%. In the under-sampling method, the sensitivity and specificity of the MLP network are 88% and 62.5% respectively, and for the RBF network they are equal to 92% and 71.4% respectively. The oversampling method has the best output with precision, sensitivity, and specificity of the network, and RBF predicts the best performance compared to MLP.

In addition, the effective factors of kidney transplantation patients were identified using two algorithms: optimization of binary particles and nearest neighbor in two sampling methods. In the oversampling method, with factors such as *donator age, recipient age, gender, recipient mass index, consistency of the donator and recipient blood group, duration of preoperative dialysis, familial relative, history of kidney transplantation, presence of diabetes in the recipient*, the donator type was marked with 98.72% precision and 99.13% accuracy, and in the undersampling method, with factors such as *donator age, recipient age, recipient mass index, consistency of the donator and recipient blood group, duration of preoperative dialysis, history of kidney transplantation, presence of diabetes in the recipient*, the donator type was marked

with 89.66% precision and 96.87% accuracy. Tables 2 and 3 show the results of the proposed model.

Table 2. Results of suggested models for over sampling

	MSE	RMSE	Accuracy	Precision	Sensitivity	specificity
MLP	0.0456	0.214	93.57	95.44	93.6	91.4
RBF	0.0109	0.104	98.16	98.91	98.2	97.4
BPSO& KNN	0.0128	0.1132	99.13	98.72	100	96.68

Table 3. Results of suggested models for under sampling

	MSE	RMSE	Accuracy	Precision	Sensitivity	specificity
MLP	0.1653	0.406	81.48	83.47	88	62.5
RBF	0.0715	0.267	92	92.85	92	71.4
BPSO& KNN	0.1034	0.3215	96.87	89.66	100	89.63

4 Conclusion

Kidney transplantation is one of the most effective treatments for treating patients with advanced kidney disease. In this paper, we used two structures (multilayer perceptron neural network (MLP) and radial base networks (RBF)) in two methods (under-sampling and over-sampling) for unbalanced data. The sample size, the progress of medical treatment, the time and place studied are factors that influence the difference in the outcome. Several studies have been conducted to predict the survival of kidney transplantation in survival of 1, 3, 5 and 10 years. In the following, we compare the results of the proposed model with the best recent work to predict the survival of the kidney transplantation.

The best result of a 5-year survival prediction of a kidney transplant with a neural network structure was obtained in Egypt in 2008. It is

also shown in this study that the neural network has the best result in regression. The accuracy and sensitivity of the neural network are respectively 88 and 88.43 percent [25].

In this study, 156 data from renal transplant patients performed a 2-year survival prediction of renal transplantation with a data balancing technique. In the MLP oversampling method we had the accuracy equal to 93.57%, in RBF – 98.16%, and in the MLP under-sampling method we had the accuracy equal to 81.48% and in RBF – 92%. In the oversampling, the sensitivity and specificity of the MLP network are respectively equal to 93.6 and 91.4 percent, and for the RBF network they are respectively 98.2 and 97.4 percent. In the under-sampling, the sensitivity and specificity of the MLP network are respectively 88 and 62.5 percent, and for the RBF network they are equal to 92 and 71.4 percent respectively

The oversampling method has a more predictable model with higher precision and sensitivity compared the previous work. Hence, the predicting model with imbalanced medical data can be implemented with greater reliability and precision by balancing the data. The oversampling method has a better performance than the under-sampling method and the RBF network performs better than multi-layered MLP perceptron network in two sampling methods.

In 2012, the Bayesian network method was used to model 5144 kidney transplants with 48 attributes. BMI, race, recipient gender, and donator age were identified as the effective factors in the survival of the transplant [5]. In 2012, the vector machine identified the following effective attributes: age, level of creatinine, gender, and recipient weight [6]. In [7] there were studied neural networks, decision tree, support vector machine, and information fusion. The accuracy of neural networks, decision tree, and support vector machine were 94%, 92%, and 92%, respectively, and the accuracy of information fusion was 95.74%. Now we are considering those data. We have used their dataset and have obtained high accuracy using the oversampling method. In 2017, Nematollahi et al have examined the prediction of renal transplantation for 5 years. The results of the SVM, MLP, and logistic regression model have been estimated as 90.4%, 85.9%, 84.7%. They showed that SVM

and MLP models can efficiently be used to predict renal transplant recipient survival [2].

Identification of effective factors for transplantation in renal patients is very important, therefore, binary particles optimization algorithm and the nearest neighbor algorithm were used for obtaining effective factors in renal transplant patients in two sampling methods. In the oversampling method, precision equal to 98.72% and accuracy – to 99.13% were identified, and in the undersampling method the precision equal to 89.66% and accuracy – to 96.87% for the selected effective factors were identified. In general, donator age, recipient age, recipient mass index, blood group consistency, dialysis time before operation, kidney transplantation history and donator type were chosen as the most important factors. Table 4 shows the comparison between the best results of previous studies and the proposed method. According to the previous work, the proposed method has a better performance.

Table 4. Comparison of the proposed method with other existing studies

	Method	Best accuracy
2008 [25]	neural network, regression	88
2017 [2]	SVM, MLP-ANN, and logistic regression	90.04
Proposed Algorithm	MLP	93.57
	RBF	98.16
	BPSO&KNN	99.13

References

[1] C.V. Walraven, "Predicting potential survival benefit of renal transplantation in patients with chronic kidney disease," *CMAJ*, vol. 182, no. 7, pp. 666–672, 2010.

[2] M. Nematollahi, R. Akbari, S. Nikeghbalian, and C. Salehnasab, "Classification models to predict survival of kidney transplant recipients using two intelligent techniques of data mining and logistic regression," *International journal of organ transplantation medicine*, vol. 8, no. 2, pp. 119–122, 2017.

[3] A. Hashemian, B. Beiranvand, M. Rezaei, A. Bardideh, E. Zand-Karimi, "Comparison of artificial neural networks and cox regression models in prediction of kidney transplant survival," *International journal of advanced biological and biomedical research*, vol. 1, no. 10, pp. 1204–1212, 2013.(in Persian)

[4] k.D.Yoo et al, "A machine learning approach using survival statistics to predict graft survival in kidney transplant recipients: a multicenter cohort study," *Scientific Reports, Sci Rep* , vol. 7, no. 1, ID: 8904, 2017. DOI: 10.1038/s41598-017-08008-8. [Online]. Available: https://www.nature.com/articles/s41598-017-08008-8.

[5] T.S.Brown et al, "Bayesian modeling of pre transplant variables accurately predicts kidney graft survival," *American Journal of nephrology,AM J NEPHROL*, vol. 36, no. 6, pp. 561–569, 2012.

[6] J. Lasserre, S. Arnold, M. Vingron, P. Reinke, C. Hinrichs. "Predicting the outcome of renal transplantation," *Journal of the American Medical Informatics Association, J Am Med Inform Assoc*, vol. 19, no. 2, pp. 255–262, 2012.

[7] M. Mirzaei, M. Firooz Abadi, "The Impact of Data Mining on Prediction of Renal Transplantation Survival and Identifying the Effective Factors on the Transplanted Kidney," *Journal of Health and Biomedical Informatics Medical Informatics Research Center*, vol. 3, no. 1, pp. 1–9, 2016.

[8] U. Offor, S. Alabi, "Artificial neural network model for friction factor prediction," *Journal of materials science and chemical engineering, J. Chem. Eng. Mater. Sci*, no.4, pp. 77–83, 2016.

[9] S. Arabgol, H. Sang Ko, "Application of artificial neural network and genetic algorithm to healthcare waste prediction," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 3, no. 3, pp. 243–250, 2013.

[10] M. Rita Durao, Manuel T. Mendes, M. Joao Pereira, "Forecasting O3 levels in industrial area surroundings up to 24 h in advance, combining classification trees and MLP models," *Atmospheric pollution research, ATMOS POLLUT RES*, no. 7, no. 6, pp. 961–970, 2016.

[11] A. Ebrahimabadi, M. Azimipour, A. Bahreini, "Prediction of road headers performance using artificial neural network approaches (MLP and KOSFM)," *Journal of rock mechanics and geotechnical engineering*, no.7, pp. 573–583, 2015.

[12] A. Tatar, A. Barati-Harooni, H. Moslehi, S. Naseri, M. Bahadori, M. Lee, A. Bahadori, A. Najafi-Marghmaleki, "Prediction of water formation temperature in natural gas dehydrators using radial basis function (RBF) neural networks," *Natural Gas Industry B*, no.3, no. 2, pp. 173–180, 2016.

[13] S.S. Chiddarwar, N.R. Babu, "Comparison of RBF and MLP neural networks to solve inverse kinematic problem for 6 R serial robot by a fusion approach," *Engineering applications of artificial intelligence, Eng Appl Artif Intell*, vol. 23, no. 7, pp. 1083–1092, 2010.

[14] A. Ali, S.M. Shamsuddin, A.L. Ralescu, "Classification with class imbalance problem: a review," *Int. J. Advance Soft Compu. Appl*, vol. 7, no. 3, pp. 176–204, 2015.

[15] V. Garca, J.S. Sánchez, R.A. Mollined, "On the effectiveness of preprocessing methods when dealing with different levels of class imbalance," *Knowledge-based systems, KNOWL-BASED SYST*, no. 25, no.1, pp. 13–21, 2012.

[16] S. Sadiqbatcha, S. Jafarzadeh, Y. Ampatzidis, "Particle swarm optimization for solving a class of type-1 and type-2 fuzzy nonlinear

equations," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 8, no. 2, pp. 103–110, 2018.

[17] J. Kennedy, R.C. Eberhart, "Particle swarm optimization," in *IEEE International Conference on Neural Networks*, (Perth, WA, Australia, Australia), pp. 1942–1948, 1995.

[18] P. Moradi, M. Gholampour, "A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy," *Applied soft computing, Appl Soft Comput*, no. 43, pp. 117–130, 2016.

[19] J. Kennedy, R.C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, (Orlando, FL, USA), vol. 5, pp. 4104–4109, 1997.

[20] L.Y. Chuang, C.H. Yang, J.C. Li, "Chaotic maps based on binary particle swarm optimization for feature selection," *Applied Soft Computing, Appl Soft Comput*, no. 11, no. 1, pp. 239–248, 2011.

[21] X. Wu et al, "Top 10 algorithms in data mining," *Knowledge and Information Systems, KNOWL INF SYST* , vol. 14, no. 1, pp. 1–37, 2008.

[22] A. Muayad and A.N. Irtefaa, "Ridge regression using artificial neural network," *Indian journal of science and technology, Indian J Sci Technol*, vol. 9, no. 31, 2016.

[23] M. Kouchakzade, A. Bahmani, "Assessment of artificial neural networks revenue in reducing required parameters for estimation of reference evapotranspiration," *Journal of agricultural sciences, J Agric Sci*, vol. 11, no. 4, pp. 87–97, 2006.

[24] M. Yaseri, M.S. Yekaninejad, A. Pakpour, S. Rahmani, H. Rangin, A. Akaberi, "Self-Learning concepts of diagnostic tests by graphical approach: sensitivity," *Journal of north Khorasan University of medical sciences*, vol. 4, no. 2, pp. 275–282, 2012.

[25] A. Akl, A.M. Ismail, M. Ghoneim, "Prediction of graft survival of living-donor kidney transplantation: nomograms or artificial neural networks," *Transplantation*, vol. 86, no. 10, pp. 1401–1406, 2008.

Zeinab Hassani, Nasibeh Emami, Received June 18, 2018

Zeinab Hassani
Department of Computer Science, Kosar University of Bojnord, Bojnord, Iran
North Khorasan, Bojnord, Iran
Phone: 0985832411722
E–mail: Hassani@kub.ac.ir

Nasibeh Emami
Department of Computer Science, Kosar University of Bojnord, Bojnord, Iran
North Khorasan, Bojnord, Iran
Phone: 0985832411722
E–mail: nasibeh.emami@kub.ac.ir

Computer Science Journal of Moldova, vol.**26**, no.2(77), 2018

The application of Mathematica to research the restricted eight bodies problem

Elena Cebotaru

Abstract

Newton's restricted problem of eight bodies is investigated. In this paper an effective way of determining the stationary points of differential equations describing this problem is exposed. Analytic and numerical calculations are done with the system Mathematica.

Keywords: Newtonian problem, equation of motion, configuration, particular solution, equilibrium point.

1 Introduction

We consider the Newtonian problem of n bodies.This problem consists in studying the motion of n bodies in the Newtonian gravitational field. Its description is very simple [1], [2]. It is well-known that Newtonian many-body problem is not integrable in general.

The development of new computer technologies has provided the opportunity to otherwise approach the problem of n-bodies. Computer algebra system Mathematica is a very powerful tool for doing both symbolic and numeric calculations [4]. In many cases it turned out to be possible to construct not only approximate but exact solutions of differential equations of motion.

In this paper we study the existence of stationary solutions (the equilibrium positions) in the restricted eight bodies problem with incomplete symmetry, obtained with the help of symbolic calculus system Mathematica (SCS Mathematica).

2 Description of the configuration

Assume that in a non-inertial space P_0xyz there is the motion of eight bodies P_0, P_1, P_2, P_3, P_4, P_5, P_6, P with the masses m_0, m_1, m_2, m_3, m_4, m_5, m_6, μ, which attract each other in accordance with the law of universal attraction. We will investigate the planar dynamic pattern formed by a square, in the vertices of which the bodies P_1, P_2, P_3, P_4 are placed. The body P_0 is the center of the square and the bodies P_5, P_6 are placed on the diagonal P_1P_3 of the square at equal distances from P_0. We consider that $m_5 = m_6$ and the configuration rotates around the body P_0 with the constant angular velocity ω, which is determined from the model parameters. It will be studied the motion of the infinitely small mass μ (the so-called passive gravitational body with $\mu \approx 0$) in the gravitational field formed by the seven bodies P_0, P_1, P_2, P_3, P_4, P_5, P_6 that attract each other and attract the body P. As we study the planar configuration, we have $z_j = 0$, $j = 0, 1, ..., 7$. We can assume that $P_1(1, 1)$, $P_2(-1, 1)$, $P_3(-1, -1)$, $P_4(1, -1)$, $P_5(\alpha, \alpha)$, $P_6(-\alpha, -\alpha)$, $f = 1$, $m_0 = 1$, $m_5 = m_6$, then out of the differential equations of the motion the following conditions of existence of this configuration are obtained: $m_1 = m_3$, $m_2 = m_4 = f_1(m_1, \alpha)$, $m_5 = m_6 = f_2(m_1, \alpha)$, $\omega^2 = f_3(m_1, \alpha)$.

The differential equations of the motion of the point P in the gravitational field of the points P_0, P_1, P_2, P_3, P_4, P_5, P_6 in the rotating Cartesian coordinate system P_0xyz have the form [1]:

$$\begin{cases} \dfrac{d^2X}{dt^2} - 2\omega\dfrac{dY}{dt} = \omega^2 X - \dfrac{fm_0X}{r^3} + \dfrac{\partial R}{\partial X}, \\ \dfrac{d^2Y}{dt^2} + 2\omega\dfrac{dX}{dt} = \omega^2 Y - \dfrac{fm_0Y}{r^3} + \dfrac{\partial R}{\partial Y}, \\ \dfrac{d^2Z}{dt^2} = -\dfrac{fm_0Z}{r^3} + \dfrac{\partial R}{\partial Z}, \end{cases} \tag{1}$$

where

$$\begin{cases} R = f\sum\limits_{j=1}^{6} m_j \left(\dfrac{1}{\Delta_{kj}} - \dfrac{XX_j + YY_j + ZZ_j}{r_j^3} \right), \\ \Delta_j^2 = (X_j - X)^2 + (Y_j - Y)^2 + (Z_j - Z)^2, \\ r_j^2 = X_j^2 + Y_j^2 + Z_j^2,\ r^2 = X^2 + Y^2 + Z^2, \\ j = 1, 2, ..., 6. \end{cases} \tag{2}$$

According to the definition of the stationary solutions of the differential equations, the equilibrium positions (in case when they exist) are solutions of the functional system of equations:

$$\begin{cases} u = 0,\ v = 0,\ w = 0, \\ \omega^2 X + 2\omega v - \dfrac{fm_0 X}{r^3} + \dfrac{\partial R}{\partial X} = 0, \\ \omega^2 Y - 2\omega u - \dfrac{fm_0 Y}{r^3} + \dfrac{\partial R}{\partial Y} = -\dfrac{fm_0 Z}{r^3} + \dfrac{\partial R}{\partial Z} = 0, \end{cases} \tag{3}$$

or in the deployed form

$$\begin{cases} u = 0,\ v = 0,\ w = 0, \\ \omega^2 x + 2\omega v - \dfrac{fm_0 X}{r^3} - f\sum\limits_{j=1}^{6} m_j \left(\dfrac{X - X_j}{\Delta_j^3} + \dfrac{X_j}{r_j^3} \right) = 0, \\ \omega^2 y - 2\omega u - \dfrac{fm_0 Y}{r^3} - f\sum\limits_{j=1}^{6} m_j \left(\dfrac{Y - Y_j}{\Delta_j^3} + \dfrac{Y_j}{r_j^3} \right) = 0, \\ -\dfrac{fm_0 Z}{r^3} - f\sum\limits_{j=1}^{6} m_j \left(\dfrac{Z - Z_j}{\Delta_j^3} + \dfrac{Z_j}{r_j^3} \right) = 0, \end{cases} \tag{4}$$

$$\begin{cases} \Delta_j^2 = (X_j - X)^2 + (Y_j - Y)^2 + (Z_j - Z)^2, \\ r_j^2 = X_j^2 + Y_j^2 + Z_j^2,\ r^2 = X^2 + Y^2 + Z^2, \\ j = 1, 2, ..., 6. \end{cases} \tag{5}$$

For simplicity as above it has been taken $f = 1$, $m_0 = 1$. Replacing in relations (4) $(X_j, Y_j, Z_j = 0)$, $Z = 0$, $m_2 = m_4 = f_1(m_1, \alpha)$, $m_5 = m_6 = f_2(m_1, \alpha)$ and $\omega^2 = f_3(m_1, \alpha)$, determined above for admissible α and m_1, we obtain the following system:

$$
\left\{
\begin{array}{l}
u = 0,\ v = 0,\ w = 0, \\
f(x,y) = \omega^2 x + 2\omega v - \dfrac{x}{(x^2+y^2)^{\frac{3}{2}}} + \\
+m_1\left(\dfrac{-1-x}{\left((1+x)^2+(1+y)^2\right)^{3/2}} + \dfrac{1-x}{\left((1-x)^2+(1-y)^2\right)^{3/2}}\right) + \\
+m_4\left(\dfrac{1-x}{\left((1-x)^2+(1+y)^2\right)^{3/2}} + \dfrac{-1-x}{\left((1+x)^2+(1-y)^2\right)^{3/2}}\right) + \\
+m_6\left(\dfrac{-\alpha-x}{\left((\alpha+x)^2+(\alpha+y)^2\right)^{3/2}} + \dfrac{\alpha-x}{\left((\alpha-x)^2+(\alpha-y)^2\right)^{3/2}}\right) = 0, \\
g(x,y) = \omega^2 y - 2\omega u - \dfrac{y}{(x^2+y^2)^{\frac{3}{2}}} + \\
+m_1\left(\dfrac{-1-y}{\left((1+x)^2+(1+y)^2\right)^{3/2}} + \dfrac{1-y}{\left((1-x)^2+(1-y)^2\right)^{3/2}}\right) + \\
+m_4\left(\dfrac{1-y}{\left((1-x)^2+(1+y)^2\right)^{3/2}} + \dfrac{-1-y}{\left((1+x)^2+(1-y)^2\right)^{3/2}}\right) + \\
+m_6\left(\dfrac{-\alpha-y}{\left((\alpha+x)^2+(\alpha+y)^2\right)^{3/2}} + \dfrac{\alpha-y}{\left((\alpha-x)^2+(\alpha-y)^2\right)^{3/2}}\right) = 0.
\end{array}
\right. \tag{6}
$$

3 Determination of equilibrium points

The equations in the system (6) have a rather complicated structure. Its solving is quite cumbersome. If the solution of the system (6) is

determined, then the solution of the equilibrium position of differential equations describing the restricted problem of the eight bodies is obtained. Using the graphical package of Mathematica for different parameter values α and m_1, the graphs of the curves $f(x, y)$ and $g(x, y)$ described by the equations in the system (6) have been constructed. Obviously, the points of intersection of these curves in the plain P_0xy will be the equilibrium positions of the investigated system. We will name the points that are on the lines passing through the center of the configuration and any peak of the square as radial equilibrium position (we will note them in the future by N_i). We will name the other points as equilibrium bisectorial positions (we will note them in the future by S_i). For this we use the following algorithm:

Algorithm 1

- constructs the configuration and graphs of the curves f and g;
- calculates the coordinates of the equilibrium bisectorial points S_i, $i = 1, ...4$ and displays them on the computer screen;
- shows the position of the point S_1.

This algorithm in the SCS Mathematica can be realised in the following way:

Algorithm 1 **in SCS Mathematica**

$graph[n\_, a\_] :=$
$Module[\{m_1 = n,\ \alpha = a\},\ gf = f(x, y, m_1, \alpha); gg = g(x, y, m_1, \alpha);$
$cpx = ContourPlot[gf, \{x,\ -2.5,\ 2.5\},\ \{y,\ 2.5,\ 2.5\},\ Contours \to \{0\},$
$ContourShading \to False, PlotPoints \to 100, ContourStyle \to$
$\{Black\},\ Axes \to True, Frame \to False];$
$cpy = ContourPlot[g, \{x,\ -2.5,\ 2.5\},\ \{y,\ -2.5,\ 2.5\},\ Contours \to \{0\},$
$ContourShading \to False, PlotPoints \to 100, ContourStyle \to$
$\{Dashed\},\ Axes \to True, Frame \to False];$
$square =\ ListPlot[\{\{1, 1\}, \{1, -1\}, \{-1, -1\}, \{-1, 1\}\},$
$PlotStyle \to \{PointSize[0.02]\}];$
$points =\ ListPlot[\{\{\alpha,\ \alpha\},\ \{-\alpha,\ -\alpha\}\},$
$PlotStyle \to \{PointSize[0.02]\}];$

$M0 := Graphics[Text[''P_0'', \{-0.15, -0.15\}]];$
$M1 := Graphics[Text[''P_1'', \{0.85, 1.05\}]];$
$M2 := Graphics[Text[''P_2'', \{-0.85, 1.05\}]];$
$M3 := Graphics[Text[''P_3'', \{-0.85, -1.05\}]];$
$M4 := Graphics[Text[''P_4'', \{0.85, -1.05\}]];$
$M5 := Graphics[Text[''P_5'', \{\alpha - 0.1, \alpha - 0.1\}]];$
$M6 := Graphics[Text[''P_6'', \{-\alpha + 0.1, -\alpha + 0.1\}]];$
$f1 = FiindRoot[\{gf == 0, gg == 0\}, \{x, 1\}, \{y, 0\}];$
$S1 := Graphics[Text[''S_1'', \{1.55, -0.25\}]]; Print[''S_1'', f1];$
$f2 = FiindRoot[\{gf == 0, gg == 0\}, \{x, 0\}, \{y, -1\}]; Print[''S_2'', f2];$
$f3 = FiindRoot[\{gf == 0, gg == 0\}, \{x, -1\}, \{y, 0\}]; Print[''S_3'', f3];$
$f4 = FiindRoot[\{gf == 0, gg == 0\}, \{x, 0\}, \{y, 1\}]; Print[''S_4'', f4];$
$Show[points, square, cpx, cpy, p1, p2, M0, M1, M2, M3, M4, M5,$
$M6, S1,$
$PlotRange \to \{\{-2, 2\}, \{-2, 2\}\}, DisplayFunction \to$
$\$DisplayFunction,$
$AxesLabel \to \{x, y\}, AspectRatio \to Automatic,$
$PlotLabel \to'' m_1 =''<> ToString[n];'' \alpha =''<> ToString[a]''''];$

For $m_1 = 0.01$ and $\alpha = 0.8584$ the result of this program is displayed in Figure 1.

4 Concluding remarks

In [3] A. Wintner introduced the concept of central configuration. The research configuration is of this type. The SCS Mathematica gives a possibility to consider various particular cases in the Newtonian problem of n bodies more effectively. The SCS Mathematica offers the opportunity to construct not only approximate but exact solutions of differential equations of motion. It is well known that for investigating the stability of stationary points, it is necessary to determine the equilibrium points of the configuration. In the present article we described an algorithm in the SCS Mathematica for determining the conditions of existence of equilibrium points in the restricted problem of eight bodies, and, in cases of equilibrium points' existence, the method for their building.

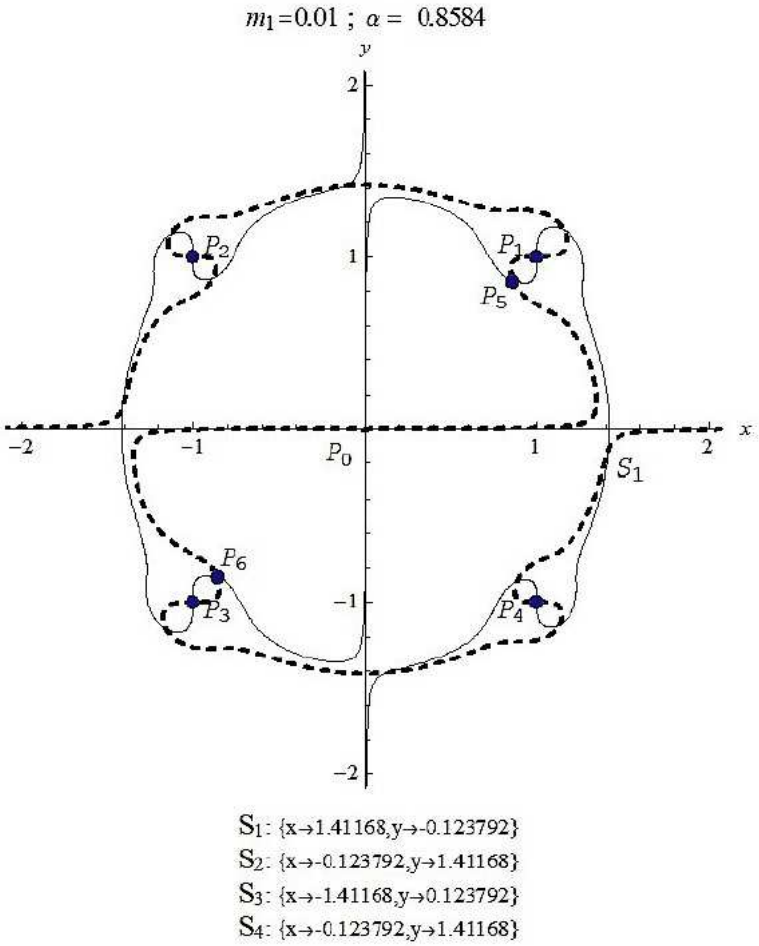


Figure 1. graph[0.01,0.8584]

References

[1] E. A. Grebenikov, *On a mathematical problems of homographic dynamics*, Moscow: MAKS Press, 2010, 253 p. (in Russian)

[2] E. A. Grebenikov, D. Kozak-Skovorodkina, M. Yakubyak, *Computer algebra methods in the many-body problem*, Second edition, revised and supplemented, Moscow: The Peoples' Friendship University of Russia, 2002, 211 p. ISBN: 5-209-01451-7. (in Russian)

[3] A. Wintner, *The Analytical Foundations of Celestial Mechanics,* Princeton, NJ,USA: Princeton University Press, 1941, 448 p.

[4] S. Wolfram, *The Mathematica Book,* Cambridge: University Press, 1996.

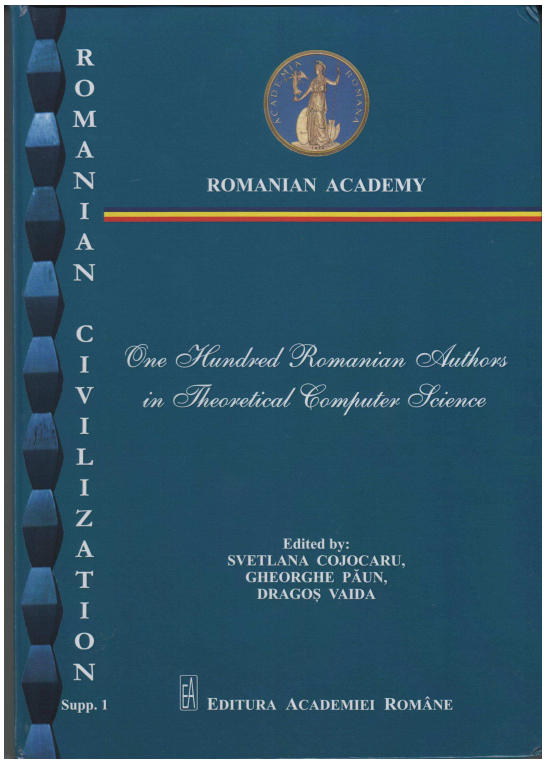
Elena Cebotaru Received July 16, 2018

Elena Cebotaru
Technical University of Moldova
Ştefan cel Mare str. 168, MD-2004 Chişinău, Republic of Moldova
Phone: 022-50-99-31
E–mail: elena.cebotaru@mate.utm.md

Computer Science Journal of Moldova, vol.**26**, no.2(77), 2018

100 Romanian Authors in Theoretical Computer Science (presentation of the book)

Svetlana Cojocaru, Gheorghe Păun, Dragoş Vaida



This book [1] may look like a Who's Who in the Romanian Theoretical Computer Science (TCS), it is a considerable step towards such an ambitious goal, but the title should warn us in several aspects.

From the very beginning we started working with the book with the idea of collecting exactly 100 short CVs. This was an artificial decision with respect to the number of Romanian computer scientists, but natural in view of the circumstances the volume got born:it belongs to a series initiated by the Romanian Academy on the occasion of celebrating one century since the Great Romania was formed, in the end of the First World War.

This series (coordinated by acad. Victor Spinei, the vice president of Romanian Academy) is simply monumental: Over 30 large volumes,

with the common title *The Romanian Civilization*, covering many scientific and cultural areas, from history and geography to mathematics, physics, and chemistry, from geology and medicine to literature and music – including computer science.

The general volume dealing with computer science in *The Romanian Civilization* series appears under the coordination of acad. Florin Gheorghe Filip. It contains a chapter called *From the History of Romanian Theoretical Computer Science,* written by one of the present editors (Gh.P.). It is a quick overview of the Romanian theoretical computer science, in less than three dozens of pages, starting from the "pre-history" of computer science, in general, not only Romanian, namely, with the 1927 paper by Gabriel Sudan, who produced, at the same time and independently of W. Ackermann, the first example of a recursive function which is not primitive recursive. (We say that this is "pre-history", because at that time computer science did not even exist, the terminology itself we used before was coined after 1930.) After that, the chapter shortly describes the activity of the two forerunners-patriarchs of the Romanian computer science, Grigore C. Moisil (see Fig.1) and Solomon Marcus (see Fig.2), both of them members of the

Figure 1. Grigore C. Moisil

Romanian Academy, and further it proceeds until our times, mentioning ideas, authors, groups, referring to both Romanians working in Romania and to Romanians spread over all the world, both to living persons and to persons passed away.

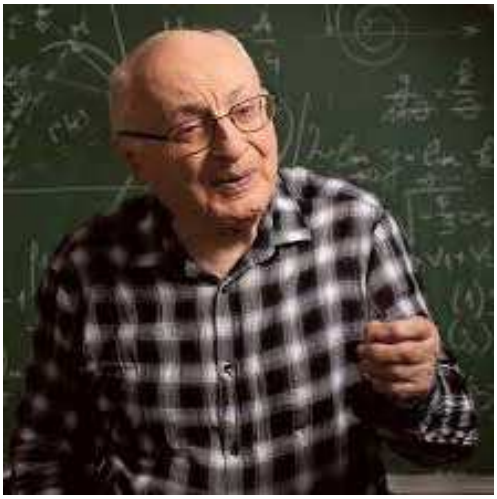


Figure 2. Solomon Marcus

Therefore, the term "history" should be understood in this framework in a broad sense, concerning the evolution of the field from the origins until the present days.

The mentioned *From the History...* chapter has obvious limits in what concerns the provided information and actually this was the starting point of the present book. The chapter is informal, hence the need for more technical details, at least at the bibliographical level, occurred. The present book provides such an information – including hints (e.g., personal web pages) on how and where more comprehensive details can be found.

However, this is done limiting to one hundred the number of presented scientists!... (One hundred, with the Centennial background, but also having in mind other limits such as the number of pages the

volume might have or the time to complete the project – also knowing that a comprehensive-complete Who's Who of an encyclopedic-dictionary type is not very realistic for computer science, even restricting ourselves to the theoretical one.)

Thus, an obvious consequence follows: this volume does not suggest a hierarchy, an ordering of any type. Just 100 Romanian authors who have significantly contributed to the (theoretical) computer science, who (1) we were able to contact (hence we knew their email addresses), and (2) they answered positively to our invitation to contribute to the volume. Precisely, we have proposed to all of them to send us, until a certain deadline, a short presentation structured along of the items most of the presentations contain.

Many computer scientists colleagues, both from Romania and from other countries, are regretfully missing from this book. A tentative list, clearly incomplete, contains both "classic" names (acad. Mircea Maliţa, Peter Hammer-Ivănescu, Samuel Abraham, Teodor Rus, Constantin Popovici) and rather young persons (many in Romania, many abroad; hundred of successful doctorates in computer science were completed in the last decades, in universities from many places), including computer scientists with a well-established career (Grigore Roşu, Cătălin Dima, Răzvan Andonie, Cătălin Ioniţă, Toader Jucan, Mihaela Maliţa, Mircea Sularia, Daniela Rus, Horia Georgescu, Virgil Emil Căzănescu, and so on and so forth). Also, only a few computer scientists who passed away were presented (in most cases with the help of disciples or colleagues of them present in the book), several others were omitted – among them: Andrei Baranga, Şerban Buzeţeanu, Dumitru (Dan) Dumitrescu, Emil Muntean, Nicolae Ţăndăreanu.

A second edition of the book, if any, will maybe contain most if not all of these names.

Another important point concerns the "definition" of the *set* of scientists to be present(ed) in the volume. This is definitely a *fuzzy set,* from two points of view: (1) what means *Romanian*, and (2) what means *theoretical.*

Because the book is dedicated to the Centenary of the Great Unity, we interpreted the term "Romanian" in a rather comprehensive man-

ner: *persons born in Romania or speaking Romanian language,* no matter which is their formal citizenship or where they are living-working. In particular, this was applied to our colleagues from Republic of Moldova – and this is valid also for the team of the three co-editors...

A much more difficult issue is that of what does theoretical computer science mean. The point was touched also in the beginning of the *From the History...* chapter. This is an endless discussion, about an undecidable (even false) problem. There is no borderline between theoretical and... what else, in the area of computer science? Hardware, software, applications? All of these areas involve "theory". Furthermore, which is the difference between theoretical computer science and mathematics (applied to computing)?

Of course, we have proceeded intuitively (to come back to the idea of fuzzy sets: also there the membership function is often defined *subjectively...*).

At this stage it would be interesting and instructive to remind some significant ideas of our forerunner Professors. In the preface written in March 1968 of his Pergamon Press book from 1969 (*The Algebraic Theory of Switching Circuits*), Gr.C. Moisil gives three examples concerning the usefulness of the "new mathematics", which ceases to be a quantitative science and begins to be a structural science: the mathematical linguistics, the pseudo-Boolean programming, and the theory of the programming languages, all of them pointing out to some priorities in the field of Theoretical Computer Science at the time. Moreover, he warns us not to forget that the first technical discipline based on this structural mathematics is the algebraic theory of switching circuits.

We would also like to cite the works of Solomon Marcus constantly emphasizing the pilot role of (formal) linguistics and mathematics, together with their cultural values, in other different bodies of knowledge such as economy, biology or physics. Accordingly, in his well-known *Reception Speech in the Romanian Academy,* he deals – perhaps for the first time in our context – with what should be understood by the common expression "mathematical language". One shows that this language becomes *even* the *existential mode* for the object of the knowledge, a most interesting point of view for a philosophy of knowl-

edge viewed in a close solidarity with practical existence.

In short, this volume is only a first step on a long way, towards a comprehensive presentation of the Romanian computer scientists – but this is a volume providing a lot of information, both at a precise, bibliographical level, and at the global level, as a general view about the Romanian (theoretical) computer science. Many conclusions can be drawn starting from the data contained in this book. We only point out, very shortly, a few observations, a more detailed analysis remains for another framework.

Both in *time* (from "old" times to present) and in *space* (looking to the map of Romania and to the map of the Earth as well), the Romanian computer science is rather developed, diversified, synchronized and very well connected with the international computer science, both historically and geographically. Besides space and time, we can also add a third "dimension", an internal one, the *range of topics* addressed by Romanian computer scientists. Again, we can note the close links (we may call it "isomorphism") with the computer science in general: practically, all branches were explored, from logic to natural computing, from automata and language theory to algebraic approaches. A powerful science, indeed. The names of Romanian computer scientists can be found from top level bibliographies of theoretical contributions to managing positions in successful companies. The next pages is a proof, especially for the theoretical area. Another observation is related to the sociology of the domain: a large number of scientists presented in this book started their research activity, for master and doctoral studies, in the theoretical area, many of them in relation with faculties of mathematics and under the scientific guidance of mathematicians (open to computer science or even working in theoretical computer science), and then they became devoted mainly to teaching or to applications of computer science, in a wide range of domains.

A few words about the style of the volume could be here in order. We solicited presentations with a precise format, but the contributions correspond to (hence reflect) a great variety of personalities and career experiences, not to mention ages and cultural environments. We tried to unify as much as possible the materials, thus losing in a great ex-

tent the personal flavor of some of the received texts. This suggests a possible continuation of this endeavour: to put together in a volume personal recollections of computer scientists with an interesting life or working experience, especially of those who witnessed the beginning of Romanian computer science or they have participated in significant developments, theoretical or of other nature. We would like to come back to such a project.

Of course, this volume was only possible due to the participation of the computer scientists present in it – we are indebted to all of them for the efficient and smooth cooperation. Several colleagues, some of them mentioned in the pages which follow, have contributed with texts, with comments, corrections – our thanks and indebtedness.

References

[1] *One Hundered Romanian Authors in Theoretical Computer Science* (Romanian Civilisation), S.Cojocaru, G.Păun and D.Vaida, Eds. Bucharest: The Publ. House of the Romanian Academy, 2018, 304 p.

Svetlana Cojocaru, Gheorghe Păun, Dragoş Vaida Received August 16, 2018

Svetlana Cojocaru

Institute of Mathematics and Computer Science,
5, Academiei street, Chisinau, Republic of Moldova, MD 2028
E–mail: Svetlana.Cojocaru@math.md

Gheorghe Păun

E–mail: curteadelaarges@gmail.com

Dragoş Vaida

E–mail: dravaida2@gmail.com