УДК 519.688

Ольшевська В. А.

DOI: https://doi.org/10.18523/2617-70804202134-40

АЛГОРИТМ ПОШУКУ КІЛЬКОСТІ РУХОМИХ ТОЧОК ПІДСТАНОВОК ІЗ СИЛОВСЬКИХ 2-ПІДГРУП $Syl_2(S_{2^n})$ СИМЕТРИЧНИХ ГРУП S_{2^n}

У статті запропоновано алгоритм пошуку кількості рухомих точок підстановок силовських 2-підгруп $Syl_2(S_{2^n})$ симетричних груп S_{2^n}, $n \in \mathbb{N}$. Для побудови цього алгоритму використано ізоморфізм між групою $Syl_2(S_{2^n})$ та групою бінарних кореневих дерев з мітками. Також обчислено складність запропонованого алгоритму та його середню кількість кроків для силовської 2-підгрупи симетричної групи S_{2^n}. Обраховано кількість підстановок із $Syl_2(S_{2^n})$, що мають максимальну кількість рухомих точок.

Ключові слова: рухомі точки, симетричні групи, силовські підгрупи, алгоритм, складність алгоритму.

Вступ

Симетрична група підстановок S_{2^n} є класичним алгебраїчним об'єктом, який також використовують в інформатиці, теорії кодування, статистиці тощо. Зокрема у теорії кодування розглядаються коди, що визначені на симетричній групі S_n або її підгрупах [1–4]. Ці коди можуть бути отримані за допомогою різних метрик: Хеммінга, Улама, Келі [2; 5]. Обчислення відстані на підстановках залежить від кількості рухомих або нерухомих точок підстановки. Тому природною є задача підрахунку кількості рухомих або нерухомих точок в певній групі підстановок.

У цій статті ми будемо розглядати кількість рухомих точок підстановок, що є елементами силовської 2-підгрупи $Syl_2(S_{2^n})$ симетричної групи S_{2^n}. Нагадаємо, що *силовською p-підгрупою групи G*, що має порядок $p^k \cdot s$, називають таку підгрупу, що має порядок p^k і позначають $Syl_p(G)$, де s не ділиться на p, p — просте, k — деяке натуральне число. Існування такої підгрупи випливає з теореми Силова [6]. Також у статті буде обчислено кількість підстановок із $Syl_2(S_{2^n})$, що мають максимальну або мінімальну ненульову кількість рухомих точок.

Підстановки, що мають максимальну ненульову кількість рухомих точок є підстановками, що переставляють всі точки із області дії групи. Кількість таких підстановок у симетричний групі S_{2^n} добре відома і її вивчення входить в курс «Дискретної математики» (задача про кількість повних безпорядків) [7]. Зрозуміло, що в силовській 2-підгрупі симетричної групи таких підстановок буде менше і їх кількість ми будемо обчислювати рекурсивно.

Необхідні визначення та допоміжні твердження

Відомим є зображення груп $Syl_2(S_{2^n})$ за допомогою таблиць спеціальної форми, яке запропонував Л.Калужнін [8]. Також відоме ще одне представлення групи підстановок через портрети [9]. У цій роботі ми будемо використовувати алгоритми зображення елементів $Syl_2(S_{2^n})$ за допомогою бінарних кореневих дерев з мітками, які були раніше записані в статті [10]. Нагадаємо, що *бінарним кореневим n-рівневим деревом* називають ациклічний простий граф з виділеною вершиною — *коренем дерева*; степінь цієї вершини рівний 2, а решта вершин, окрім висячих, мають степінь 3. Таке дерево будемо позначати T_n, а множину його вершин – $V(T_n)$ [11; 12]. Також позначимо через $LT_{2,n}$ множину всіх бінарних n-рівневих кореневих дерев із мітками 0 або 1 на всіх вершинах з 0-го по $(n-1)$-й рівні.

Нагадаємо (див. [10]), що *координатами вершини v* дерева $D \in LT_{2,n}$ є пара (j, i), де i – це номер вершини на рівні j, $i \in \{1, \ldots, 2^j\}$, $j \in \{0, \ldots, (n-1)\}$. Будемо казати, що $(j, i) < (k, r)$ якщо $j < k$ або $j = k$ та $i < r$. Також для вершин дерева будемо вважати, що $v < w$, якщо v та w мають відповідні координати (j_1, i_1) та (j_2, i_2), причому $(j_1, i_1) < (j_2, i_2)$.

Для дерева $D \in LT_{2,n}$ позначимо $OC(D)$ –

множина координат вершин з помітками 1.

Для повноти викладення матеріалу, наведемо два алгоритми (алг. 1, алг. 2) із статті [10]. Ці алгоритми визначають два взаємообернені відображення, які будемо використовувати надалі: $\psi : LT_{2,n} \to Syl_2(S_{2^n})$ (за алгоритмом 1), $\tau : Syl_2(S_{2^n}) \to LT_{2,n}$ (за алгоритмом 2).

Algorithm 1: Алгоритм перетворення дерева у підстановку

```
Input: OC(D) be a set of coordinates of
       all vertices labeled by 1 of a tree D
. Output: (a_{i_1}, a_{i_2}, ··· a_{i_{2^n}}) is the second
       row of permutation.
(a_1, a_2, ··· a_{2^n}) = (1, 2, ··· , 2^n) ;
for (j, i) ∈ (OC(D), <) do
    m := 2^{n-j-1} ( is count of elements in
     one block) ;
    for l := 1 to m do
        b := a_{(2i-2)m+l};
        a_{(2i-2)m+l} := a_{(2i-1)m+l};
        a_{(2i-1)m+l} := b;
```

Algorithm 2: Алгоритм перетворення підстановки у дерево

```
Input: (a_1, a_2, ..., a_{2^n}) is the second row
       of 2-separated permutation.
Output: OC(D).
OC(D) := ∅;
for j := 0 to n − 1 do
    m := 2^{n-j-1} (length of block);
    for i := 1 to 2^j do
        if a_{(2i-2)m+1} > a_{(2i-1)m+1} then
            OC(D) := OC(D) ⋃ {(j, i)}
```

Як випливає з алгоритмів 1 та 2, довільну підстановку $\pi \in Syl_2(S_{2^n})$ ми можемо зображати *відповідним* їй бінарним кореневим n-рівневим деревом $D = \tau(\pi) \in LT_{2,n}$ і навпаки.

Другий рядок $a = (a_1, a_2, \ldots, a_{2^n})$ підстановки $\pi = \begin{pmatrix} 1 & 2 & \ldots & 2^n \\ a_1 & a_2 & \ldots & a_{2^n} \end{pmatrix}$ будемо називати *блоком елементів.*

Також нагадаємо означення 2-роздільної підстановки із [10], що було використано для алгоритму 2 та буде нам необхідне далі.

Означення 1. Підстановка π є 2-*роздільною,* якщо для неї можна виконати такі кроки:
1) спочатку розділимо блок a навпіл на два підблоки: $u_1 = (a_1, \ldots, a_{2^{n-1}})$ та $u_2 = (a_{2^{n-1}+1}, \ldots, a_{2^n})$; потім перевіримо, чи кожен елемент із u_1 більший (або менший) за кожен елемент із u_2;
2) якщо крок 1 виконується, то повторюємо дії та розділяємо кожен блок u_1 та u_2 відповідно на два підблоки $u_{1,1}$, $u_{1,2}$ та $u_{2,1}$ $u_{2,2}$; після чого перевіряємо величини елементів між відповідними блоками. І так далі, поки не отримаємо блоки, що містять лише по одному елементу.

Кількість рухомих точок підстановки із $Syl_2(S_{2^n})$

Означення 2. Кількість рухомих точок підстановки π – це кількість позицій, де елементи нижньої стрічки підстановки відрізняються від елементів верхньої стрічки [5]. Позначимо: $h(\pi)$.

Приклад 1. $\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & \mathbf{4} & \mathbf{3} & 5 & 6 & \mathbf{8} & \mathbf{7} \end{pmatrix}$
Кількість рухомих точок для цієї підстановки: $h(\pi) = 4$.

Кількість рухомих точок підстановки будемо рахувати, використовуючи ізоморфізм між силовською 2-підгрупою симетричної групи $Syl_2(S_{2^n})$ та групою бінарних кореневих дерев з мітками $LT_{2,n}$.

Нагадаємо, що 2^{n-j} – це кількість висячих вершин n-рівневого дерева, що розташовані під вершиною з координатами (j, i).

Лема 1. *Нехай $\pi \in Syl_2(S_{2,n})$ – така підстановка, що відповідне їй дерево $D \in LT_{2,n}$ має єдину мітку 1 на вершині з координатами (j, i). Тоді:*

$$h(\pi) = 2^{n-j}.$$

Доведення. Доведення леми 1 випливає безпосередньо із перевірки.

Лема 2. *Нехай підстановка π має відповідне їй дерево D, в якого мітки 1 розташовані на вершинах $v_1, v_2, \ldots, v_r$ з координатами $(j_1, i_1), (j_2, i_2) \ldots, (j_r, i_r)$. Причому жодна вершина v_k, $k \in \{1, 2, \ldots, r\}$ не лежить на шляху, що сполучає будь-яку іншу вершину цієї множин з коренем. Тоді:*

$$h(\pi) = 2^{n-j_1} + 2^{n-j_2} + \ldots + 2^{n-j_r} = \sum_{k=1}^{r} 2^{n-j_k}.$$

Доведення. За умовою леми ми маємо, що під кожною вершиною v_k з міткою 1, $k \in \{1, \ldots, r\}$, містяться різні висячі вершини. А тому, за визначенням множення дерев із [10], дерево D розкладається у добуток:

$$D = D_1 \cdot D_2 \cdot \ldots \cdot D_r,$$

де D_k має єдину мітку 1 на вершині v_k з координатами (j_k, i_k), $k \in \{1, \dots r\}$.

Тоді для кожного $k \in \{1, \dots r\}$ за лемою 1:

$$h(\pi_k) = 2^{n-j_k},$$

де π_k відповідає дереву D_k. Оскільки підстановки $\pi_1, \dots \pi_r$ діють на різні елементи, то:

$$h(\pi) = h(\pi_1) + h(\pi_2) + \ldots + h(\pi_r) =$$

$$= 2^{n-j_1} + 2^{n-j_2} + \ldots + 2^{n-j_r} = \sum_{k=1}^{r} 2^{n-j_k}$$

Лему доведено.

Нехай $v_0, v, w \in V(D)$. Будемо казати, що *вершина v розташована під вершиною w (вершина w над вершиною v)*, якщо w належить шляху, що з'єднує v з коренем дерева v_0. Позначимо: $v \succ w$.

Також позначимо $L(v)$ – *множина всіх висячих вершин, що розташовані лише під вершиною v.*

Лема 3. *Нехай підстановка π така, що на відповідному їй дереві D лише дві вершини v_k та v_q з координатами (j_k, i_k) та (j_q, i_q) мають мітки 1, причому $v_q \succ v_k$. Тоді:*

$$h(\pi) = 2^{n-j_k}.$$

Доведення. Оскільки $v_q \succ v_k$, то $j_q > j_k$ та мають місце такі співвідношення:
1. $|L(v_q)| = 2^{n-j_q}$ і це менше ніж $|L(v_k)| = 2^{n-j_k}$;
2. $L(v_q) \subset L(v_k)$.

Оскільки v_q міститься у лівій або правій гілці, для яких v_k є коренем, то і всі вершини під v_q будуть розташовуватися у тій самій гілці. А тому під час застосування алгоритму 1 перетворення дерева у підстановку, дія на координатах (j_q, i_q) не повертає висячі вершини у початкову гілку. Таким чином, за лемою 1, кількість рухомих точок підстановки π буде залежати від кількості висячих вершин із множини $L(v_k)$ та буде рівною її потужності:

$$h(\pi) = |L(v_k)| = 2^{n-j_k}.$$

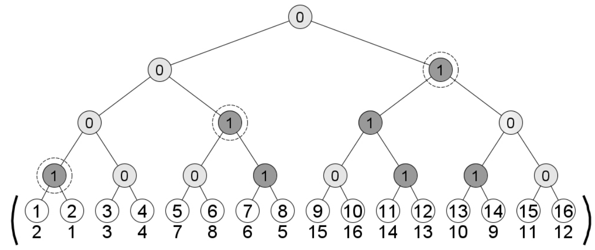
Лему доведено.

Означення 3. Нехай вершина v дерева має мітку 1. Якщо на шляху між нею та коренем решта вершин мають мітки 0, то v будемо називати *головною вершиною.*

Наслідок 4. *Кількість рухомих точок підстановки π, що задається деревом D, рівна сумі кількості висячих вершин, що розміщені під головними вершинами дерева.*

Доведення. Доведення випливає із леми 2 та леми 3.

Приклад 2. Розглянемо підстановку $\pi \in Syl_2(S_{2^4})$ та відповідне їй дерево $D \in LT_{2,4}$:



Тут головні вершини мають координати: $(1,2), (2,2), (3,1)$. Відповідно до означення 3 та наслідку 4, маємо кількість рухомих точок:

$$h(\pi) = 2^{4-1} + 2^{4-2} + 2^{4-3} = 8 + 4 + 2 = 14.$$

Якщо виконати перевірку стандартного означення 2 про кількість рухомих точок підстановки, то отримаємо таке саме значення.

Алгоритм знаходження кількості рухомих точок підстановки із $Syl_2(S_{2^n})$.
Введемо позначення:
$a[k]$ – k-та координата стрічки a;
$a[b, c]$ – стрічка, що має від b-ї до c-ї координат стрічки a, тобто є частиною стрічки a;
$len(a)$ – функція, що визначає кількість координат стрічки a.

Наприклад, нехай маємо стрічку $a = (5, 10, 2, 3, 7, 4)$, тоді:
$a[2] = 10$, $a[2, 5] = (10, 2, 3, 7)$, $len(a) = 6$.

Використовуючи алгоритм перетворення підстановки у дерево 2, задамо рекурсивний алгоритм для обчислення кількості рухомих точок підстановки $\pi \in Syl_2(S_{2^n})$.

Algorithm 3: Алгоритм знаходження кількості рухомих точок $h(\pi)$

Input: $a = (a[1], a[2], \cdots a[2^n])$ – нижня стрічка підстановки π.
Output: MovCount – кількість рухомих точок
Створюємо рекурсивну підпрограму з аргументом a;
MovCount(a);
if $a[1] > a[\frac{len(a)}{2} + 1]$ **then**
  return $len(a)$;
if $len(a) = 2$ **then**
  return 0;
return $MovCount\Big(a\big[1\ ,\frac{len(a)}{2}\big]\Big)$
$+MovCount\Big(a\big[\frac{len(a)}{2} + 1\ ,\ len(a)\big]\Big)$

Твердження 5. *Часова складність алгоритму знаходження кількості рухомих точок підстановки* $\pi \in Syl_2(S_{2^n})$ *рівна* $O(2^n)$.

Доведення. Кожна пара елементів зі стрічки a на кроці порівняння відповідає певній вершині дерева із 0-го по $(n-1)$-й рівні. У найгіршому випадку потрібно буде виконати стільки порівнянь, скільки таких вершин, а саме $2^n - 1$. Тому маємо оцінку

$$O(2^n - 1) = O(2^n).$$

Твердження доведено.

Ця оцінка є оцінкою зверху та показує кількість кроків у найгіршому випадку. Таких випадків значно менше, ніж решти, завдяки властивості 2-роздільності підстановок.

Створено програму, яка для алгоритму 3 рахує середнє значення кількості порівнянь для кожного n. Цю програму вдалося виконати та перевірити для невеликих $n = 2, 3, 4$ за допомогою мови комп'ютерної алгебри Sage.

n	2^n довжина підстановки	Потужність $Syl_2(S_{2^n})$	Кількість порівнянь за означенням	Середня кількість порівнянь алгоритму
2	4	$\lvert Syl_2(S_4)\rvert = 2^3 = 8$	4	2
3	8	$\lvert Syl_2(S_8)\rvert = 2^7 = 128$	8	3
4	16	$\lvert Syl_2(S_{16})\rvert = 2^{15} = = 32768$	16	4
5	32	$\lvert Syl_2(S_{32})\rvert = 2^{31}$	32	5

Можна побачити таку залежність: середнє значення кількості порівнянь дорівнює n для відповідного значення n. Тоді для загального випадку маємо таку теорему.

Теорема 6. *Для знаходження кількості рухомих точок підстановки* $\pi \in Syl_2(S_{2^n})$ *за алгоритмом 3 в середньому необхідно виконати* n *порівнянь.*

Доведення. Припустимо, що для множини $Syl_2(S_{2^n})$, що містить підстановки довжини 2^n (позначимо π) та кожна з яких задається відповідним n-рівневим деревом (позначимо D), в середньому для алгоритму 3 необхідно n порівнянь. Тобто:

$$\frac{\Sigma}{2^{2^n - 1}} = n,$$

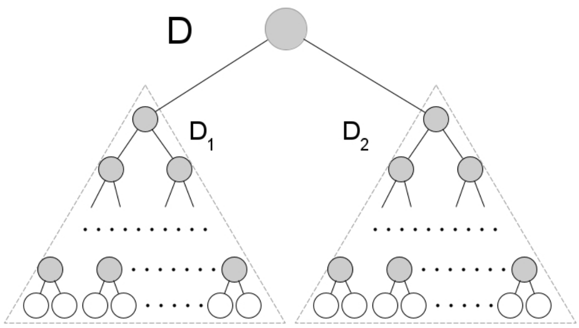
$$\text{де } \Sigma = 1 \cdot N_1 + 2 \cdot N_2 + \ldots + (2^n - 1) \cdot N_{2^n - 1},$$

N_k – це кількість підстановок, для яких алгоритм 3 виконується за k кроків (порівнянь).

Тобто маємо:

$$n \cdot 2^{2^n - 1} = 1 \cdot N_1 + 2 \cdot N_2 + \ldots + (2^n - 1) \cdot N_{2^n - 1}. \quad (1)$$

Перевіримо кількість кроків для $n + 1$.



Зауважимо, що дерево D має $n + 1$ рівні та його корінь сполучає дві головні гілки, які є n-рівневими кореневими деревами D_1 та D_2 відповідно. Розглянемо випадки:

1. Нехай корінь дерева D має мітку 1. Тоді для знаходження кількості рухомих точок за алгоритмом 3 буде виконано лише 1 крок, бо на корені ми і зупинимося. Таких дерев із множини $LT_{2,n+1}$, що мають мітку 1 на корені, всього $2^{2^{n+1} - 2}$. Тобто для знаходження середньої оцінки будемо використовувати:

$$1 \cdot 2^{2^{n+1} - 2}. \quad (2)$$

2. Нехай корінь дерева D має мітку 0. Тоді алгоритм 3 розгляне корінь (1 дія) та перейде на рівень нижче, до лівого та правого піддерев. Тобто далі маємо таку таблицю:

Корінь дерева D	Кількість вершин лівого піддерева, D_1	Кількість вершин правого піддерева, D_2	Результат для обчислень
1	1	1	$(1+1+1)N_1N_1$
1	1	2	$(1+1+2)N_1N_2$
....	....	....	....
1	1	2^n-1	$(1+1+(2^n-1))N_1N_{2^n-1}$
1	2	1	$(1+2+1)N_2N_1$
1	2	2	$(1+2+2)N_2N_2$
....	....	....	....
1	2	2^n-1	$(1+2+(2^n-1))N_2N_{2^n-1}$
....	....	....	....
1	2^n-1	1	$(1+(2^n-1)+1)N_{2^n-1}N_1$
1	2^n-1	2	$(1+(2^n-1)+2)N_{2^n-1}N_2$
....	....	....	....
1	2^n-1	2^n-1	$(1+(2^n-1)+$ $+(2^n-1))N_{2^n-1}N_{2^n-1}$

Тому загальна сума для $n+1$ в цьому випадку буде:

$$(1+1+1)N_1N_1+(1+1+2)N_1N_2+\ldots+(1+$$
$$+1+(2^n-1))N_1N_{2^n-1}+$$
$$+(1+2+1)N_2N_1+(1+2+2)N_2N_2+\ldots+(1+$$
$$+2+(2^n-1))N_2N_{2^n-1}+\ldots$$
$$\ldots+(1+(2^n-1)+1)N_{2^n-1}N_1+(1+(2^n-1)+$$
$$+2)N_{2^n-1}N_2+\ldots+(1+(2^n-1)+(2^n-$$
$$-1))N_{2^n-1}N_{2^n-1}.$$

З кожної стрічки винесемо за дужки N_1, N_2, ... та N_{2^n-1} відповідно:

$$N_1\Big((1+1+1)N_1+(1+1+2)N_2+\ldots+(1+1+(2^n-$$
$$-1))N_{2^n-1}\Big)+N_2\big((1+2+1)N_1+(1+2+2)N_2+\ldots$$
$$\ldots+(1+2+(2^n-1))N_{2^n-1}\Big)+\ldots+N_{2^n-1}\Big((1+$$
$$+(2^n-1)+1)N_1+(1+(2^n-1)+2)N_2+\ldots+(1+$$
$$+(2^n-1)+(2^n-1))N_{2^n-1}\Big)=$$
$$=N_1\Big(2(N_1+N_2+\ldots+N_{2^n-1})+(1N_1+2N_2+\ldots$$
$$\ldots+(2^n-1)N_{2^n-1})\Big)+$$
$$+N_2\Big(3(N_1+N_2+\ldots+N_{2^n-1})+(1N_1+2N_2+\ldots$$
$$\ldots+(2^n-1)N_{2^n-1})\Big)+\ldots$$
$$\ldots+N_{2^n-1}\Big(2^n(N_1+N_2+\ldots+N_{2^n-1})+(1N_1+2N_2+$$
$$\ldots+(2^n-1)N_{2^n-1})\Big).$$

Зауважимо, що $N_1+N_2+\ldots+N_{2^n-1}$ – це вся кількість підстановок довжини 2^n з групи $Syl_2(S_{2^n})$, тобто $N_1+N_2+\ldots+N_{2^n-1}= =|Syl_2(S_{2^n})|=2^{2^n-1}$.

Також застосуємо рівність (1):

$$N_1\Big(2\cdot2^{2^n-1}+n\cdot2^{2^n-1}\Big)+N_2\Big(3\cdot2^{2^n-1}+n\cdot2^{2^n-1}\Big)+\ldots$$
$$\ldots+N_{2^n-1}\Big(2^n\cdot2^{2^n-1}+n\cdot2^{2^n-1}\Big)=2^{2^n-1}\Big(2N_1+$$
$$+nN_1+3N_2+nN_2+\ldots+2^nN_{2^n-1}+nN_{2^n-1}\Big)=$$
$$=2^{2^n-1}\Big(n(N_1+N_2+\ldots+N_{2^n-1})+(1N_1+2N_2+\ldots$$
$$\ldots+(2^n-1)N_{2^n-1})+(N_1+N_2+\ldots+N_{2^n-1})\Big)=$$
$$=2^{2^n-1}\Big((n+1)\cdot2^{2^n-1}+n\cdot2^{2^n-1}\Big)=$$
$$=2^{2^{n+1}-2}\Big(2n+1\Big). \qquad (3)$$

За результатами обох випадків (2) та (3) маємо середню оцінку для $n+1$:

$$\frac{1\cdot2^{2^{n+1}-2}+2^{2^{n+1}-2}\cdot(2n+1)}{2^{2^{n+1}-1}}=$$
$$=\frac{2^{2^{n+1}-2}\cdot(2n+2)}{2^{2^{n+1}-1}}=\frac{2n+2}{2}=n+1.$$

Отже, теорему доведено.

Кількість підстановок із $Syl_2(S_{2^n})$, що мають мінімальну ненульову або максимальну кількість рухомих точок

Мінімальна ненульова кількість рухомих точок. Оскільки ми розглядаємо підстановки $\pi\in Syl_2(S_{2^n})$, то для них зберігається властивість 2-роздільності. А тому найменшою ненульовою кількість рухомих точок для таких підстановок є $h(\pi)=2$.

Така рівність може виконуватися лише для підстановок, які задаються таким деревом $D\in \in LT_{2,n}$, що має лише одну вершину з міткою 1 і причому на $(n-1)$-му рівні (випливає із лем 1, 2 та 3). Тобто кількість таких підстановок буде рівна кількості таких дерев. А їх відповідно буде стільки, скільки різних вершин на $(n-1)$-рівні, тобто:

$$\Big|\Big\{\pi\mid h(\pi)=2 \text{ та } \pi\in Syl_2(S_{2^n})\Big\}\Big|=2^{n-1}$$

Максимальна кількість рухомих точок. Максимальною кількістю рухомих точок підстановки $\pi \in Syl_2(S_{2^n})$ є число, яке рівне довжині самої підстановки. Тобто $h(\pi) = 2^n$.

Теорема 7. *Кількість підстановок* $\pi \in \in Syl_2(S_{2^n})$, *які мають* 2^n *рухомих точок, дорівнює* $f(n)$, *що визначається рекурсивно так:*

$$f(n) = \begin{cases} 1, \text{ коли } n = 1 \\ 2^{2^n-2} + f(n-1) \cdot f(n-1). \end{cases}$$

Доведення. Доведення будемо виконувати за Методом математичної індукції.

База індукції: $n = 1$. Для $Syl_2(S_{2^1})$ єдиною не тотожною підстановкою є транспозиція $(1, 2)$. А тому $f(1) = 1$ – визначено коректно.

Індуктивний крок. Нехай для індексів, менших за n, умова теореми виконується. Перевіримо для n. Нехай $D \in LT_{2,n}$, розглянемо такі випадки.
1. Нехай дерево D на корені має мітку 1. Тоді за лемою 3: $h(\pi) = 2^n$, де $\pi \in Syl_2(S_{2^n})$ відповідає дереву D.

За означенням дерев із множини $LT_{2,n}$, кількість таких різних дерев становить 2^{2^n-2}.
2. Нехай дерево D на корені має мітку 0 (тобто не має мітку 1). Також нехай $D_1, D_2 \in LT_{2,n-1}$ – його ліве та праве піддерево відповідно. Тоді за наслідком 4:

$$h(\pi) = 2^n \text{ тоді і тільки тоді, коли}$$

$$h(\pi_1) = h(\pi_2) = 2^{n-1}.$$

За припущенням індукції, кількість різних підстановок із $Syl_2(S_{2^{n-1}})$ (та відповідно і їх дерев із $LT_{2,n-1}$), що мають 2^{n-1} рухомих точок, становить $f(n-1)$.

Тому кількість таких дерев $D \in LT_{2,n}$ рівна $f(n-1) \cdot f(n-1)$.

Оскільки випадки 1. та 2. неперетинні, то:

$$f(n) = 2^{2^n-2} + f(n-1) \cdot f(n-1).$$

Отже, теорему доведено.

Список літератури

1. Blake Ian F., Cohen Gerard, Deza Mikhail. Coding with permutations. *Information and Control.* Academic Press, New York etc. 1979. Vol. 43. Pp. 1–19.
2. Cameron Peter J. Permutation codes. *European Journal of Combinatorics.* 2010. Vol. 31, Issue 2. Pp. 482–490.
3. Dénes J. On some connections between permutations and coding. *Discrete Mathematics.* 1985. Vol. 56. Pp. 141–146.
4. Sobhani R., Abdollahi A., Bagherian J., Khatami M. A note on good permutation codes from Reed-Solomon codes. *Designs, Codes and Cryptography.* 2019. Vol. 87. Pp. 2335–2340.
5. Irurozki E., Calvo B., Lozano J. A. Sampling and learning the Mallows and Weighted Mallows models under the Hamming distance. *Technical Report.* University of the Basque Country. URL: http://hdl.handle.net/10810/11240. 2014.
6. Ленг С. Алгебра. Москва : Наука, 1965.
7. Боднарчук Ю. В., Олійник Б. В. Основи дискретної математики. Києво-Могилянська академія, 2009.
8. Калужнин Л. А. Избранные главы теории групп. Киев : Киевский государственный университет им. Т. Г. Шевченко, 1979. С.22–26.
9. Nekrashevych V. Self-similar groups. *Mathematical Surveys and Monographs.* 2005. Vol. 117. P. xi + 231.
10. Olshevska V. A. Algorithms for computations with Sylow 2-subgroups of symetric groups. *Silesian Journal of Pure and Applied Mathematics.* 2020. Vol. 10. Pp. 103–120.
11. Grigorchuk R. I., Nekrashevich V. V., Sushchanskii V. I. Automata, dynamical systems, and groups. Moscow: MAIK Nauka/Interperiodica Publishing, 2000. Pp. 128–203.
12. Diestel Reinhard. Graph theory. *Graduate Texts in Mathematics.* 2017. Vol. 173. Pp. xviii + 428.

References

1. Ian F. Blake, Gerard Cohen and Mikhail Deza, “Coding with permutations”, Information and Control, **43**, 1–19 (1979).
2. Peter J. Cameron, “Permutation codes”, European Journal of Combinatorics. **31** (2), 482–490 (2010).
3. J. Dénes, “On some connections between permutations and coding”, Discrete Mathematics. **56**, 141–146 (1985).
4. R. Sobhani, A. Abdollahi, J. Bagherian and M. Khatami, “A note on good permutation codes from Reed-Solomon codes”, Designs, Codes and Cryptography. **87**, 2335–2340 (2019).
5. E. Irurozki, B. Calvo and J. A. Lozano. Sampling and learning the Mallows and Weighted Mallows models under the Hamming distance, in: *Technical Report*, University of the Basque Country., http://hdl.handle.net/10810/11240. 2014.
6. С. Ленг, *Алгебра* (Наука, Москва, 1965).
7. Ю.vВ. Боднарчук і Б. В. Олійник, *Основи дискретної математики* (Києво-Могилянська академія, 2009).
8. Л. А. Калужнин, *Избранные главы теории групп* (Киевский государственный университет им. Т. Г. Шевченко, Киев, 1979), с.22–26.
9. V. Nekrashevych, “Self-similar groups”, Mathematical Surveys and Monographs.**117**, xi + 231 (2005).
10. V. A. Olshevska, “Algorithms for computations with Sylow 2-subgroups of symetric groups”, Silesian Journal

of Pure and Applied Mathematics. **10**, 103–120 (2020).
11. R. I. Grigorchuk, V. V. Nekrashevich and V. I. Sushchanskii, *Automata, dynamical systems, and groups* (MAIK Nauka/Interperiodica Publishing, Moscow, 2000), pp. 128–203.
12. Reinhard Diestel, “Graph theory”, Graduate Texts in Mathematics. 5th edition. **173**, xviii + 428 (2017).

V. Olshevska

SEARCH ALGORITHM OF THE NUMBER OF UNFIXED POINTS OF PERMUTATIONS FROM SYLOW 2-SUBGROUPS $Syl_2(S_{2^n})$ OF SYMMETRIC GROUPS S_{2^n}

The Symmetric permutation group S_{2^n} is a classical algebraic object that is also used in Computer science, Coding theory, Statistics, etc. In particular, the coding theory considers codes defined on the symmetric group S_n or its subgroups. The research of permutation codes has been started from 1970s. These codes can be obtained with using different distances: Hamming, Ulam, Cailey, Levenshtein. The finding distance on permutations depends on their number of fixed or unfixed points. Therefore, it is natural to count the number of unfixed points in a certain group of permutations.

In this paper, we consider the number of unfixed points of permutations that are elements of the Sylow 2-subgroup $Syl_2(S_{2^n})$ of symmetric groups S_{2^n}. Leo Kaluzhnin used tables to represent the elements of these groups [8]. Volodymyr Nekrashevych represented permutations by their portraits [9]. We use algorithms that describe the connection between the permutation group $Syl_2(S_{2^n})$ and the group of labeled binary rooted trees [10].

An algorithm for finding the number of unfixed points for permutations of the Sylow 2-subgroup $Syl_2(S_{2^n})$ of the symmetric group S_{2^n} is proposed in the article. An isomorphism between the group $Syl_2(S_{2^n})$ and a group of labeled binary root trees was used to construct this algorithm. It is proved, that the algorithm of searching the number of unfixed point for permutations of the Sylow 2-subgroup $Syl_2(S_{2^n})$ of the symmetric group S_{2^n} has complexity $O(2^n)$. In addition, the average number of steps of the algorithm for the Sylow 2-subgroup of the symmetric group S_{2^n} is found. The result for small n (n = 2, 3, 4) was verified with a program, that is written in the language of the computer algebra Sage.

At the end of the article we find the number of permutations from $Syl_2(S_{2^n})$ that have a maximum number of unfixed points. The number of such permutations in the symmetric group S_{2^n} is well known. Obviously that this number is smaller for the Sylow 2-subgroup of the symmetric group $Syl_2(S_{2^n})$. In this case, we calculate the maximum number of unfixed points using a recursive formula.

Keywords: unfixed points, symmetric groups, sylow subgroups, algorithm, complexity.

Матеріал надійшов 03.09.2021