УДК 004.383.3:[512.645.3-026.512.1]:[517.51-022.217-025.54] DOI: 10.30748/soi.2018.154.11

А.В. Ізмайлов[1], Л.Б. Петришин[1,2]

[1] *ДВНЗ «Прикарпатський національний університет імені Василя Стефаника», Івано-Франківськ*
[2] *AGH Науково-технологічний університет у Кракові, Краків*

# ЦИФРОВА ОБРОБКА ІНФОРМАЦІЇ В РОЗОСЕРЕДЖЕНИХ СИСТЕМАХ УПРАВЛІННЯ ІЗ ЗАСТОСУВАННЯМ ШВИДКОГО ОРТОГОНАЛЬНОГО ПЕРЕТВОРЕННЯ НА ОСНОВІ ТРІЙКОВИХ СИМЕТРИЧНИХ ФУНКЦІЙ

*Розглядається ортогональне перетворення на основі трійкових симетричних функцій та його ефективність за критерієм операційної складності. На основі даного ортогонального перетворення синтезоване швидке перетворення за принципом divide-and-conquer. Проведений порівняльний аналіз ефективності розробленого швидкого перетворення з ортогональними перетвореннями Уолша-Адамара та Хаара за критерієм операційної складності. Розроблене швидке перетворення дозволяє реалізувати переваги ортогонального перетворення на основі трійкових симетричних функцій у задачах цифрової обробки інформації, пов'язаних з ущільненням даних.*

***Ключові слова:*** *цифрова обробка інформації, трійкові симетричні функції, швидке ортогональне перетворення, divide-and-conquer.*

## Вступ

Цифрова обробка інформації є ключовою системною функцією інформаційних та систем управління розпорошеними об'єктами економіки, зокрема промислового виробництва, сфери послуг, управління та зв'язку [1–13]. З даного факту випливає, що ефективні рішення у галузі цифрової обробки інформації сприятимуть підвищенню ефективності перебігу процесів у відповідних галузях застосування.

Аналіз останніх досліджень у галузі цифрової обробки інформації вказує на те, що актуальним завданням є розробка і впровадження методів ортогональних перетворень інформаційних потоків [1–3; 6; 8–12]. Окрім цього, актуальною залишається проблема ефективної, з точки зору швидкодії, реалізації даних перетворень, тобто синтез швидких перетворень [1; 6; 8; 11–14].

Незважаючи на значну кількість досліджень у визначеному напрямі, жодне з них не ставить за мету реалізувати переваги системи трійкових симетричних функцій та породженої нею системи числення, які є відомими [3–5; 12]. У роботі [2] доведена ефективність застосування ортогонального перетворення на основі трійкових симетричних функцій для задач перетворення сильнокорельованих сигналів у слабкокорельовані коефіцієнти. Звідси випливає необхідність та актуальність завдання синтезу на основі даного ортогонального перетворення відповідного швидкого перетворення.

**Метою дослідження** є синтез швидкого перетворення на основі ортогонального, побудованого з використанням системи трійкових симетричних функцій та оцінювання ефективності його застосування за критерієм операційної складності.

Наукова новизна отриманих результатів полягає в синтезі швидкого ортогонального перетворення на основі трійкових симетричних функцій та порівнянні ефективності його застосування за критерієм операційної складності з перетвореннями Уолша-Адамара та Хаара.

## Ортогональне перетворення на основі трійкових симетричних функцій

Трійкові симетричні функції задаються аналітичним виразом [3]:

$$\mathrm{Ter}_n(x) = \begin{cases} 0, & 0 \le \mathrm{modh}(x - 3^n, 3^{n+1}) < 3^n, \\ 1, & 3^n \le \mathrm{modh}(x - 3^n, 3^{n+1}) < 2 \cdot 3^n, \\ -1, & 2 \cdot 3^n \le \mathrm{modh}(x - 3^n, 3^{n+1}) < 3^{n+1}, \end{cases} \quad (1)$$

де n – порядковий номер функції; x – цілочисельний аргумент; mod(x,p) – допоміжна функція, задана аналітичним виразом (2):

$$\mathrm{modh}(x, p) = \begin{cases} \mathrm{mod}(x, p) + p, & \text{якщо } x < 0, \\ \mathrm{mod}(x, p), & \text{якщо } x \ge 0, \end{cases} \quad (2)$$

де mod(x,p) – функція залишку від ділення числа x на число p.

З метою побудови ортогонального перетворення на основі трійкових симетричних функцій систему (1) модифіковано до вигляду [3–4]:

$$\mathrm{Ter}\,01_{n,i}(\theta) = \mathrm{Ter}_{\left|3^n - 1 - i\right|}(\theta * 3^{3^n}), \quad (3)$$

© Ізмайлов А.В., Петришин Л.Б., 2018

де $n=\log_3 N$ – порядок набору базисних функцій теоретико-числових перетворень;

N – кількість функцій у наборі;

$\theta=t/T$ – параметр часу, тобто час, нормований до інтервалу T;

t – поточне значення часу;

i=0,1,2,…,N-1 – порядковий номер функції;

$Ter_n(\theta)$ – трійкові симетричні функції, які задані аналітичним виразом (1).

Для подолання неповноти системи (3) на її основі побудовано систему добутків [3]:

$$Ter\,01Mult_{ord,num}(\theta) =$$

$$= \prod_{j=1}^{3^{ord}} Ter\,01_{ord,3^{ord}-j}(\theta)^{A^{j}_{num+1}}, \qquad (4)$$

де $ord=\log_3(\log_3 N)$ – порядок набору базисних функцій теоретико-числових перетворень; $N=3^{3^{ord}}$ – кількість функцій у наборі ord; num=0,1,…,N–1 – порядковий номер функції у наборі; $\theta=t/T$ – параметр часу, тобто час, нормований до інтервалу T; t – поточне значення часу; $3^{ord}$ – кількість функцій $Ter01_{n,i}(\theta)$, які задані аналітичним виразом (3), у наборі ord; $A^{j}_{num}$ – елемент рядка num та стовпця j матриці трійкового незваженого коду Грея A.

Функції системи (4) не є взаємно ортогональними і з метою подолання даного недоліку у роботі [3] проведена її ортогоналізація за процедурою Грама-Шмідта. Отримана система ортогоналізованих добутків є лінійно незалежною, ортогональною та повною [3] і для спрощення запису може бути позначена як

$$Ter_n^{(i)}(\theta), \qquad (5)$$

де $n=\log_3(\log_3 N)$ – порядок набору базисних функцій теоретико-числових перетворень; $N=3^{3^n}$ – кількість функцій у наборі; i=0,1,2,…,N-1 – порядковий номер функції у наборі.

Детальний аналіз процедур перетворення системи (1) до вигляду (5) наведений у роботі [3].

Для кожної функції системи (5) можна ввести нормуючий коефіцієнт у вигляді [2]:

$$C_n^{(i)} = \sqrt{\frac{3^{3^n}}{\sum_{t=0}^{3^{3^n}-1} Ter_n^{(i)}\left(\frac{t}{3^{3^n}}\right)^2}}, \qquad (6)$$

де $n=\log_3(\log_3 N)$ – порядок набору базисних функцій теоретико-числових перетворень; $N=3^{3^n}$ – кількість функцій у наборі; i=0,1,2,…,N-1 – порядковий номер функції у наборі; $Ter_n^{(i)}(\theta)$ – ортогоналізовані добутки трійкових симетричних функцій (система (5)).

Використовуючи функції (5) та коефіцієнти (6), можна побудувати матрицю ортогонального перетворення на основі трійкових симетричних функцій вигляду [2]:

$$T_n = \frac{1}{\sqrt{N}} T_n', \qquad (7)$$

де $T_n'$ – матриця розмірності $N=3^n$ значень функцій системи (5) помножених на коефіцієнти (6).

Дискретне ортогональне перетворення на основі трійкових симетричних функцій одновимірного інформаційного потоку $X=\{X(0),X(1),\ldots,X(N-1)\}$ в матричному записі подається згідно співвідношення (8):

$$Y = T_n X, \qquad (8)$$

де $Y=[Y(0),Y(1),\ldots,Y(N-1)]^T$ – N-компонентний вектор спектральних коефіцієнтів ортогонального перетворення; $N=3^n$, n=0,1,…; $T_n$ – матриця перетворення розміру N×N, яка задана виразом (7); $X=[X(0),X(1),\ldots,X(N-1)]^T$ – N-компонентний вектор дискретних значень одновимірного сигналу, для якого здійснюється перетворення.

Детальний аналіз формування перетворення (8) наведено у роботі [2].

## Швидке ортогональне перетворення на основі трійкових симетричних функцій

У загальному випадку для реалізації перетворення (8) (прямий метод обчислення) необхідно N(N–1) операцій додавання/віднімання, $N^2$ операцій множення двох елементів та N операцій множення на константу (оскільки мова йде про константу $1/\sqrt{N}$, то даних операцій можна уникнути, попередньо перемноживши на дану константу всі елементи матриці перетворення).

Якщо розглянути матриці $T_n'$ (для спрощення візуального аналізу і без втрати загальності, можна розглядати матрицю $T_2'$ розмірності 9×9 (рис. 1)), то можна дійти наступних висновків: по-перше, частину елементів матриці складають нулі; по-друге, присутні елементи, які рівні між собою або рівні за модулем; по-третє, перший рядок матриці завжди складається з одиниць.

Отримані висновки вказують на можливість спрощення процедури обчислення перетворення (8), тобто на можливість формування на його основі швидкого перетворення.

$$\begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-\sqrt{3/2} & -\sqrt{3/2} & -\sqrt{3/2} & 0 & 0 & 0 & \sqrt{3/2} & \sqrt{3/2} & \sqrt{3/2} \\
1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} & -\sqrt{2} & -\sqrt{2} & -\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \\
-3/2 & 0 & 3/2 & 0 & 0 & 0 & -3/2 & 0 & 3/2 \\
3/2 & 0 & -3/2 & 0 & 0 & 0 & -3/2 & 0 & 3/2 \\
0 & 0 & 0 & -3/\sqrt{2} & 0 & 3/\sqrt{2} & 0 & 0 & 0 \\
1/\sqrt{2} & -\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} & -\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} & -\sqrt{2} & 1/\sqrt{2} \\
-\sqrt{3}/2 & \sqrt{3} & -\sqrt{3}/2 & 0 & 0 & 0 & \sqrt{3}/2 & -\sqrt{3} & \sqrt{3}/2 \\
1/2 & -1 & 1/2 & -1 & 2 & -1 & 1/2 & -1 & 1/2
\end{pmatrix}$$

Рис. 1. Матриця значень функцій перших двох порядків першого набору системи (5) помножених на коефіцієнти (6)

Найбільш універсальним та інтуїтивним способом спрощення обчислення перетворення є використання рекурсивних процедур зменшення розмірності, зокрема, так званого, методу divide-and-conquer (англ. «Розділяй та володарюй»), який можна вважати одним з найпростіших та найбільш уживаних [7; 9; 14–15].

Для того, щоб реалізувати швидке перетворення на основі перетворення (8) за допомогою методу divide-and-conquer, необхідно виділити властивості матриць $T_n'$, які дозволять на основі матриць молодших порядків обчислювати матриці старших порядків.

Оскільки попереднім перемноженням рядків матриць $T_n'$ на $1/\sqrt{N}$ можна досягти зменшення операційної складності перетворення (8), то доцільно у подальшому аналізувати властивості матриць (7) замість $T_n'$.

З метою зменшення кількості операцій множення у перетворенні (8) доцільно увести коефіцієнти, на які достатньо буде помножити результати, обчислені за допомогою операцій додавання/віднімання, для обчислення відповідних спектральних коефіцієнтів. Описані коефіцієнти можна реалізувати у вигляді

$$C_n(i-1) = \sqrt{\frac{1}{3^n}}(T_n')_i^k,\ k = \underset{j\in NZ(n,i)}{\operatorname{argmin}}\left(\left|(T_n')_i^j\right|\right), \qquad (9)$$

де n=1,2,… – порядок матриці $T_n'$; $(T_n')_i^j$ – елемент матриці $T_n'$, який знаходиться у i-му рядку та j-му стовпці; NZ(n,i) – множина номерів стовпців ненульових елементів i-го рядка матриці $T_n'$, i=1,2,…,$3^n$.

Множенням кожного з рядків матриці $T_n$ на обернене до відповідного коефіцієнта (9) число (відбувається ділення рядка на відповідний йому коефіцієнт) отримано допоміжні матриці $TH_n$ (матриця $TH_2$ розмірності 9×9 представлена на рис. 2).

$$\begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & -1 & -1 & 0 & 0 & 0 & 1 & 1 & 1 \\
1 & 1 & 1 & -2 & -2 & -2 & 1 & 1 & 1 \\
-1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 \\
1 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\
1 & -2 & 1 & 1 & -2 & 1 & 1 & -2 & 1 \\
-1 & 2 & -1 & 0 & 0 & 0 & 1 & -2 & 1 \\
1 & -2 & 1 & -2 & 4 & -2 & 1 & -2 & 1
\end{pmatrix}$$

Рис. 2. Допоміжна матриця $TH_2$, отримана на основі матриці $T_2$

Допоміжні матриці $TH_n$ володіють властивостями, які дозволяють рекурентним шляхом на основі матриць молодших порядків обчислювати матриці старших порядків. Функціонування таких властивостей найкраще проілюструвати на прикладі перетворення розмірності N=27, яке може бути обчислене згідно

$$\begin{pmatrix} Y(0) \\ Y(1) \\ Y(2) \\ Y(3) \\ \dots \\ Y(26) \end{pmatrix} = T_3 \begin{pmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ \dots \\ X(26) \end{pmatrix}. \qquad (10)$$

Для спрощення подальших викладок можна, без втрати загальності, опустити множення на коефіцієнти (9) у проміжних етапах перетворення. Першу третину коефіцієнтів можна обчислити наступним чином:

$$\begin{pmatrix} Y(0)\\ Y(1)\\ Y(2)\\ Y(3)\\ Y(4)\\ Y(5)\\ Y(6)\\ Y(7)\\ Y(8) \end{pmatrix} = TH_2 \begin{pmatrix} 1&1&1&0&0&0&0&0&0\\ 0&0&0&1&1&1&0&0&0\\ 0&0&0&0&0&0&1&1&1\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0 \end{pmatrix} \begin{pmatrix} X(0)\\ X(1)\\ X(2)\\ X(3)\\ X(4)\\ X(5)\\ X(6)\\ X(7)\\ X(8) \end{pmatrix} + TH_2 \begin{pmatrix} 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 1&1&1&0&0&0&0&0&0\\ 0&0&0&1&1&1&0&0&0\\ 0&0&0&0&0&0&1&1&1\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0 \end{pmatrix} \begin{pmatrix} X(9)\\ X(10)\\ X(11)\\ X(12)\\ X(13)\\ X(14)\\ X(15)\\ X(16)\\ X(17) \end{pmatrix} +$$

(11)

$$+TH_2 \begin{pmatrix} 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 1&1&1&0&0&0&0&0&0\\ 0&0&0&1&1&1&0&0&0\\ 0&0&0&0&0&0&1&1&1 \end{pmatrix} \begin{pmatrix} X(18)\\ X(19)\\ X(20)\\ X(21)\\ X(22)\\ X(23)\\ X(24)\\ X(25)\\ X(26) \end{pmatrix} = TH_2 \begin{pmatrix} X(0)+X(1)+X(2)\\ X(3)+X(4)+X(5)\\ X(6)+X(7)+X(8)\\ X(9)+X(10)+X(11)\\ X(12)+X(13)+X(14)\\ X(15)+X(16)+X(17)\\ X(18)+X(19)+X(20)\\ X(21)+X(22)+X(23)\\ X(24)+X(25)+X(26) \end{pmatrix} = TH_2 \begin{pmatrix} X_1(0)\\ X_1(1)\\ X_1(2)\\ X_1(3)\\ X_1(4)\\ X_1(5)\\ X_1(6)\\ X_1(7)\\ X_1(8) \end{pmatrix},$$

де $X_1(i)=X(0+3i)+X(1+3i)+X(2+3i)$, $i=0,1,\ldots,8$.

Аналогічним чином обчислюється третя третина коефіцієнтів:

$$\begin{pmatrix} Y(18)\\ Y(19)\\ Y(20)\\ Y(21)\\ Y(22)\\ Y(23)\\ Y(24)\\ Y(25)\\ Y(26) \end{pmatrix} = TH_2 \begin{pmatrix} X(0)-2X(1)+X(2)\\ X(3)-2X(4)+X(5)\\ X(6)-2X(7)+X(8)\\ X(9)-2X(10)+X(11)\\ X(12)-2X(13)+X(14)\\ X(15)-2X(16)+X(17)\\ X(18)-2X(19)+X(20)\\ X(21)-2X(22)+X(23)\\ X(24)-2X(25)+X(26) \end{pmatrix} = TH_2 \begin{pmatrix} X_1(18)\\ X_1(19)\\ X_1(20)\\ X_1(21)\\ X_1(22)\\ X_1(23)\\ X_1(24)\\ X_1(25)\\ X_1(26) \end{pmatrix} \quad (12)$$

де $X_1(i+18)=X(0+3i)-2X(1+3i)+X(2+3i)$, $i=0,1,\ldots,8$.

Для обчислення другої третини коефіцієнтів перетворення за допомогою рекурентної процедури подібної до (11) та (12) необхідно увести матриці рекурентного перетворення

$$MT_1 = \begin{pmatrix} 1&1&1\\ -1&0&1\\ 1&-2&1 \end{pmatrix} = TH_1., \quad (13)$$

$$MT_2 = \begin{pmatrix} 1&0&1\\ -1&0&1\\ 0&1&0 \end{pmatrix}. \quad (14)$$

За умови, що $MTH_1=MT_2$, допоміжні матриці рекурентного перетворення $MTH_p$ для $p>1$ визначаються за допомогою аналітичного виразу

$$MTH_p = \begin{bmatrix} MT_2 \otimes MTH_{p-1,1}\\ MT_1 \otimes MTH_{p-1,2}\\ MT_2 \otimes MTH_{p-1,3}\\ \ldots\\ MT_1 \otimes MTH_{p-1,3^{p-2}}\\ MT_2 \otimes MTH_{p-1,3^{p-1}} \end{bmatrix}, \quad (15)$$

де $\otimes$ – добуток Кронекера; $MTH_{p,i}$ – і-й рядок матриці $MTH_p$.

З використанням матриць (15), другу третину коефіцієнтів перетворення (10) можна обчислити наступним чином:

$$\begin{pmatrix} Y(9) \\ Y(10) \\ Y(11) \\ Y(12) \\ Y(13) \\ Y(14) \\ Y(15) \\ Y(16) \\ Y(17) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ -1 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & -1 & 0 & 1 & -1 & 0 & 1 \\ 1 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 \\ -1 & 0 & 1 & 2 & 0 & -2 & -1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} -X(0)+X(2) \\ -X(3)+X(5) \\ -X(6)+X(8) \\ -X(9)+X(11) \\ -X(12)+X(14) \\ -X(15)+X(17) \\ -X(18)+X(20) \\ -X(21)+X(23) \\ -X(24)+X(26) \end{pmatrix} = MTH_2 \begin{pmatrix} X_1(9) \\ X_1(10) \\ X_1(11) \\ X_1(12) \\ X_1(13) \\ X_1(14) \\ X_1(15) \\ X_1(16) \\ X_1(17) \end{pmatrix}, \quad (16)$$

де $X_1(i+9)=(-X(0+3i))+X(2+3i)$, $i=0,1,\ldots,8$.

У результаті проведених процедур зменшення розмірності з перетворення (10) розмірності N=27 отримано три перетворення (11–12) та (16) розмірності N=9. Кожне з них за схожим принципом може бути поділене на три перетворення розмірності N=3. Перша третина перетворення (11) може бути обчислена наступним чином:

$$\begin{pmatrix} Y(0) \\ Y(1) \\ Y(2) \end{pmatrix} = TH_1 \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} X_1(0) \\ X_1(1) \\ X_1(2) \end{pmatrix} +$$

$$+TH_1 \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} X_1(3) \\ X_1(4) \\ X_1(5) \end{pmatrix} + TH_1 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} X_1(6) \\ X_1(7) \\ X_1(8) \end{pmatrix} =$$

$$= TH_1 \begin{pmatrix} X_1(0)+X_1(1)+X_1(2) \\ X_1(3)+X_1(4)+X_1(5) \\ X_1(6)+X_1(7)+X_1(8) \end{pmatrix} = TH_1 \begin{pmatrix} X_2(0) \\ X_2(1) \\ X_2(2) \end{pmatrix}, \quad (17)$$

де $X_2(i)=X_1(0+3i)+X_1(1+3i)+X_1(2+3i)$, $i=0,1,2$.

Для коефіцієнтів перетворення з 3 по 5 спрощення відбувається за наступною процедурою:

$$\begin{pmatrix} Y(3) \\ Y(4) \\ Y(5) \end{pmatrix} = MT_2 \begin{pmatrix} -X_1(0)+X_1(2) \\ -X_1(3)+X_1(5) \\ -X_1(6)+X_1(8) \end{pmatrix} = MT_2 \begin{pmatrix} X_2(3) \\ X_2(4) \\ X_2(5) \end{pmatrix}, \quad (18)$$

де $X_2(i+3)=(-X_1(0+3i))+X_1(2+3i)$, $i=0,1,2$.

Аналогічним до (17) чином обчислюються коефіцієнти з 6 по 8:

$$\begin{pmatrix} Y(6) \\ Y(7) \\ Y(8) \end{pmatrix} = TH_1 \begin{pmatrix} X_2(6) \\ X_2(7) \\ X_2(8) \end{pmatrix}, \quad (19)$$

де $X_2(i+6)=X_1(0+3i)-2X_1(1+3i)+X_1(2+3i)$, $i=0,1,2$.

Коефіцієнти перетворення (12) обчислюються аналогічно перетворенню (11) і у них фігурують наступні значення відповідних величин $X_2$:

$$\begin{gathered} X_2(i+18)=X_1(18+(0+3i))+X_1(18+(1+3i))+ \\ +X_1(18+(2+3i)),\ i=0,1,2. \\ X_2(i+21)=(-X_1(18+(0+3i)))+X_1(18+(2+3i)), \quad (20) \\ i=0,1,2. \\ X_2(i+24)=X_1(18+(0+3i))-2X_1(18+(1+3i))+ \\ +X_1(18+(2+3i)),\ i=0,1,2. \end{gathered}$$

Рекурсивна процедура зменшення розмірності перетворення для рівності (16) полягає у поділі на наступні 3 частини:

$$\begin{pmatrix} Y(9) \\ Y(10) \\ Y(11) \end{pmatrix} = \left[ MT_2 \otimes MT_{2,1} \right] \begin{pmatrix} X_1(9) \\ X_1(10) \\ X_1(11) \\ X_1(12) \\ X_1(13) \\ X_1(14) \\ X_1(15) \\ X_1(16) \\ X_1(17) \end{pmatrix} =$$

$$= MT_2 \begin{bmatrix} MT_{2,1} \begin{pmatrix} X_1(9) \\ X_1(10) \\ X_1(11) \end{pmatrix} \\ MT_{2,1} \begin{pmatrix} X_1(12) \\ X_1(13) \\ X_1(14) \end{pmatrix} \\ MT_{2,1} \begin{pmatrix} X_1(15) \\ X_1(16) \\ X_1(17) \end{pmatrix} \end{bmatrix} = MT_2 \begin{pmatrix} X_2(9) \\ X_2(10) \\ X_2(11) \end{pmatrix}, \quad (21)$$

де $X_2(i+9)=X_1(9+(0+3i))+X_1(9+(2+3i))$, $i=0,1,2$.

$$\begin{pmatrix} Y(12) \\ Y(13) \\ Y(14) \end{pmatrix} = MT_1 \begin{pmatrix} X_2(12) \\ X_2(13) \\ X_2(14) \end{pmatrix}, \quad (22)$$

де $X_2(i+12)=(-X_1(9+(0+3i)))+X_1(9+(2+3i))$, $i=0,1,2$.

$$\begin{pmatrix} Y(15) \\ Y(16) \\ Y(17) \end{pmatrix} = MT_2 \begin{pmatrix} X_1(10) \\ X_1(13) \\ X_1(16) \end{pmatrix}. \quad (23)$$

Рівності (17–23) полягають в обчислені 3-точкового перетворення (8), що є, водночас, найменшою можливою розмірністю даного перетворення. Тому наступним кроком у реалізації швидкого перетворення на основі перетворення (10) є підстановка числових значень відповідних матриць та безпосереднє обчислення матричних добутків. Вирази для всіх коефіцієнтів Y перетворення (10) зручно представити у вигляді таблиці (табл. 1).

Таблиця 1

Вирази для обчислення коефіцієнтів перетворення (10)

| Коефіцієнт перетворення | Вираз для обчислення відповідного коефіцієнта |
|---|---|
| Y(0) | $C_3(0)*(X_2(0)+X_2(1)+X_2(2))$ |
| Y(1) | $C_3(1)*(-X_2(0)+X_2(2))$ |
| Y(2) | $C_3(2)*(X_2(0)-2X_2(1)+X_2(2))$ |
| Y(3) | $C_3(3)*(X_2(3)+X_2(5))$ |
| Y(4) | $C_3(4)*(-X_2(3)+X_2(5))$ |
| Y(5) | $C_3(5)*(X_2(4))$ |
| Y(6) | $C_3(6)*(X_2(6)+X_2(7)+X_2(8))$ |
| Y(7) | $C_3(7)*(-X_2(6)+X_2(8))$ |
| Y(8) | $C_3(8)*(X_2(6)-2X_2(7)+X_2(8))$ |
| Y(9) | $C_3(9)*(X_2(9)+X_2(11))$ |
| Y(10) | $C_3(10)*(-X_2(9)+X_2(11))$ |
| Y(11) | $C_3(11)*(X_2(10))$ |
| Y(12) | $C_3(12)*(X_2(12)+X_2(13)+X_2(14))$ |
| Y(13) | $C_3(13)*(-X_2(12)+X_2(14))$ |
| Y(14) | $C_3(14)*(X_2(12)-2X_2(13)+X_2(14))$ |
| Y(15) | $C_3(15)*(X_1(10)+X_1(16))$ |
| Y(16) | $C_3(16)*(-X_1(10)+X_1(16))$ |
| Y(17) | $C_3(17)*(X_1(13))$ |
| Y(18) | $C_3(18)*(X_2(18)+X_2(19)+X_2(20))$ |
| Y(19) | $C_3(19)*(-X_2(18)+X_2(20))$ |
| Y(20) | $C_3(20)*(X_2(18)-2X_2(19)+X_2(20))$ |
| Y(21) | $C_3(21)*(X_2(21)+X_2(23))$ |
| Y(22) | $C_3(22)*(-X_2(21)+X_2(23))$ |
| Y(23) | $C_3(23)*(X_2(22))$ |
| Y(24) | $C_3(24)*(X_2(24)+X_2(25)+X_2(26))$ |
| Y(25) | $C_3(25)*(-X_2(24)+X_2(26))$ |
| Y(26) | $C_3(26)*(X_2(24)-2X_2(25)+X_2(26))$ |

Із табл. 1 та рекурсивної природи властивостей матриць $TH_n$ та $MTH_n$ випливає, що для обчислення усіх коефіцієнтів перетворення (8) застосовуються наступні операційні блоки:

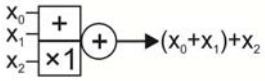


Рис. 3. Операційний блок, який реалізовує функцію $f(x_0, x_1, x_2)=(x_0+x_1)+x_2$

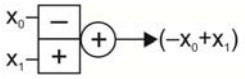


Рис. 4. Операційний блок, який реалізовує функцію $f(x_0, x_1)=(-x_0+x_1)$

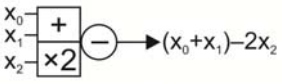


Рис. 5. Операційний блок, який реалізовує функцію $f(x_0, x_1, x_2)=(x_0+x_1)-2x_2$

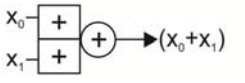


Рис. 6. Операційний блок, який реалізовує функцію $f(x_0, x_1)=(x_0+x_1)$

Рис. 7. Операційний блок, який реалізовує функцію $f(x_0)=(x_0)$ (повторювач)

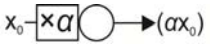


Рис. 8. Операційний блок, який реалізовує функцію $f(x_0)=(\alpha x_0)$ (помножувач на константу)

Обернене перетворення до перетворення (10) можна подати у вигляді:

$$\begin{pmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ \dots \\ X(26) \end{pmatrix} = T_3{}^T \begin{pmatrix} Y(0) \\ Y(1) \\ Y(2) \\ Y(3) \\ \dots \\ Y(26) \end{pmatrix} = TH_3{}^T \begin{pmatrix} Y_0(0) \\ Y_0(1) \\ Y_0(2) \\ Y_0(3) \\ \dots \\ Y_0(26) \end{pmatrix}, \quad (24)$$

де $T_3{}^T$ – транспонована матриця $T_3$; $Y_0(i)=C_3(i)*Y(i)$; i=0,1,…,26.

Враховуючи ортогональність матриць $T_n$ [2], властивості операції транспонування матриць (у тому числі відносно добутку Кронекера) та властивість асоціативності матричного множення можна сформувати рекурентну реалізацію перетворення (24), яку можна проілюструвати виразами (25) для обчислення відновлених значень вектора вхідних даних X:

$$\begin{gathered} X(0+3i)=Y_2(0+i)-Y_2(9+i)+Y_2(18+i), \\ X(1+3i)=Y_2(0+i)-2Y_2(18+i), \\ X(2+3i)=Y_2(0+i)+Y_2(9+i)+Y_2(18+i), \end{gathered} \quad (25)$$

де i=0,1,…,8,

$$\begin{gathered} Y_2(0+3i)=Y_1(0+i)-Y_1(3+i)+Y_1(6+i), \\ Y_2(1+3i)=Y_1(0+i)-2Y_1(6+i), \\ Y_2(2+3i)=Y_1(0+i)+Y_1(3+i)+Y_1(6+i), \\ Y_2(9+3i)=Y_1(9+i)-Y_1(12+i), \\ Y_2(10+3i)=Y_1(15+i), \\ Y_2(11+3i)=Y_1(9+i)+Y_1(12+i), \\ Y_2(18+3i)=Y_1(18+i)-Y_1(21+i)+Y_1(24+i), \\ Y_2(19+3i)=Y_1(18+i)-2Y_1(24+i), \\ Y_2(20+3i)=Y_1(18+i)+Y_1(21+i)+Y_1(24+i), \end{gathered}$$

де i=0,1,2,

$$\begin{gathered} Y_1(0+6i)=Y_0(0+(0+6i))-Y_0(1+(0+6i))+Y_0(2+(0+6i)), \\ Y_1(1+6i)=Y_0(-1+(1+6i))-2Y_0(1+(1+6i)), \\ Y_1(2+6i)=Y_0(-2+(2+6i))+Y_0(-1+(2+6i))+Y_0(0+(2+6i)), \end{gathered}$$

де i=0,1,…,4,

$$\begin{gathered} Y_1(3+6i)=Y_0(0+(3+6i))-Y_0(1+(3+6i)), \\ Y_1(4+6i)=Y_0(1+(4+6i)), \\ Y_1(5+6i)=Y_0(-2+(5+6i))+Y_0(-1+(5+6i)), \end{gathered}$$

де i=0,1,…,3.

Із рівностей (25) випливає, що для відновлення вхідних даних перетворення (10) за допомогою оберненого перетворення окрім операційних блоків, які зображені на рис. 3–8, застосовуються наступні операційні блоки:

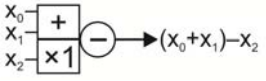


Рис. 9. Операційний блок, який реалізовує функцію $f(x_0, x_1, x_2)=(x_0+x_1)-x_2$

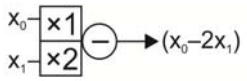


Рис. 10. Операційний блок, який реалізовує функцію $f(x_0, x_1)=(x_0-2x_1)$

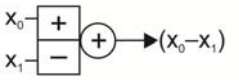


Рис. 11. Операційний блок, який реалізовує функцію $f(x_0, x_1)=(x_0-x_1)$ (вводиться з метою спрощення візуального сприйняття схеми перетворення)

Використовуючи операційні блоки, які зображені на рис. 3–11, можна реалізувати графічну схему (рис. 12) швидкого перетворення (8) для кількості відліків N=27.

Зі схеми швидкого ортогонального перетворення (рис. 12) випливає, що кількість операцій додавання/віднімання та множення на константу в прямому та оберненому перетвореннях співпадають і рівні, відповідно, 134 та 27. Окрім цього, перша ітерація прямого і остання ітерація оберненого перетворення містять однакову кількість операцій додавання, аналогічно, друга ітерація прямого і передостання ітерація оберненого і т.д. Необхідно зазначити, що при підрахунку операцій операція множення на 2 розглядалась як сума двох однакових чисел.

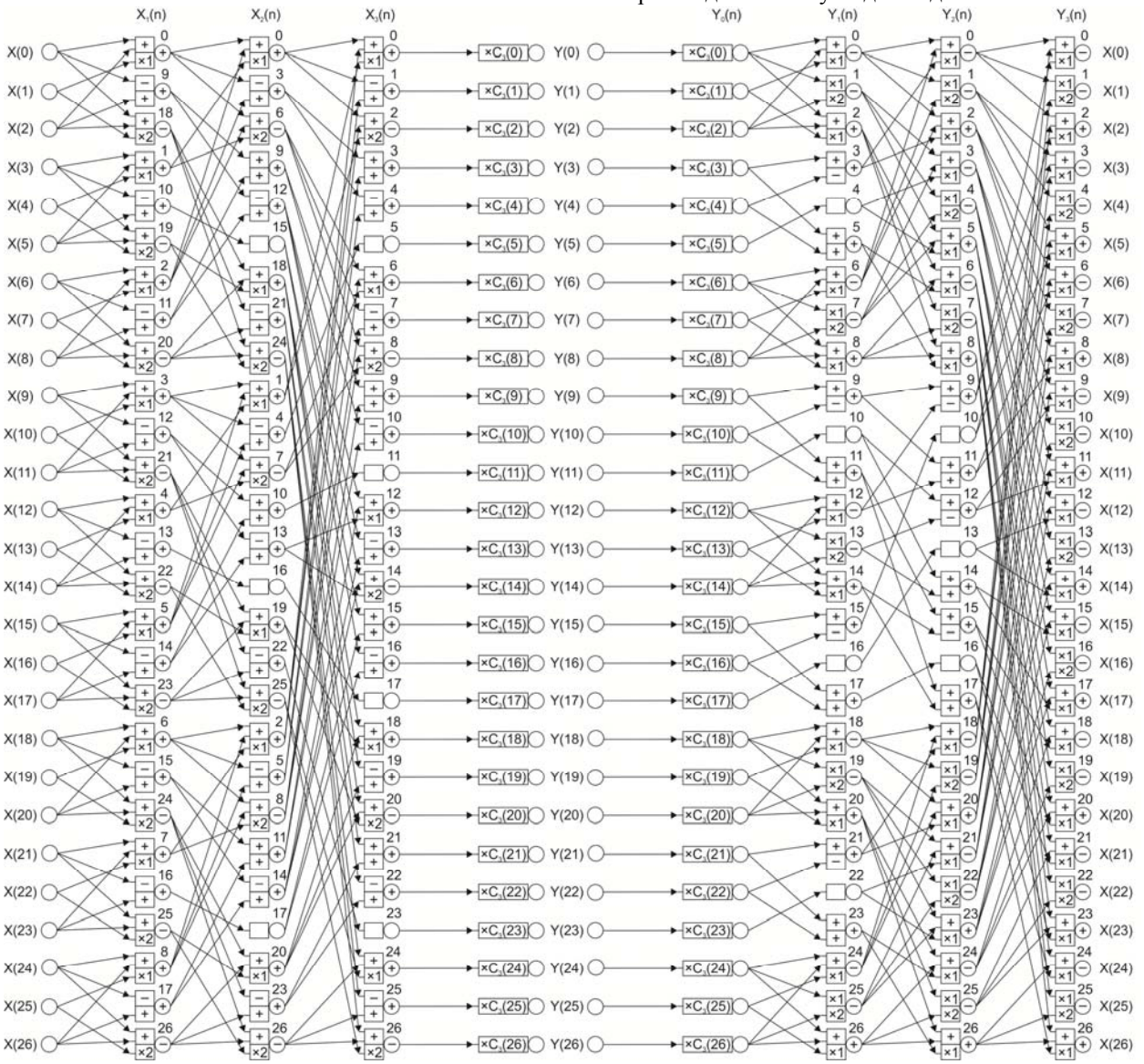


Рис. 12. Схема швидкого ортогонального перетворення на основі трійкових симетричних функцій розмірності N=27, реалізованого за принципом divide-and-conquer

Аналіз схеми прямого перетворення на рис. 12 вказує, що для реалізації першої ітерації швидкого перетворення необхідно

$$\frac{N}{3}*2+\frac{N}{3}*1+\frac{N}{3}*3=2N \qquad (26)$$

операцій додавання/віднімання. Дане твердження можна узагальнити для довільного допустимого N (яке вказує на розмірність перетворення), враховуючи, що із рівностей (11–12) та (16) випливає, що на першій ітерації співвідношення кількостей операційних блоків залишається незмінним. У випадку

другої ітерації кількість операцій додавання/віднімання випливає зі співвідношення (27), а у випадку третьої – зі співвідношення (28).

$$\left(\frac{N}{3}-\frac{N}{9}\right)*(2+1+3)+\frac{N}{9}(1+1). \quad (27)$$

$$\left(\frac{N}{3}-\left(\frac{N}{9}+\frac{N}{27}\right)\right)*(2+1+3)+\left(\frac{N}{9}+\frac{N}{27}\right)*(1+1). \quad (28)$$

Співвідношення (27) та (28) не можна узагальнити для довільного допустимого N, якщо розглядати їх у відношенні до другої та третьої ітерацій перетворення, однак це можна зробити, якщо розглядати їх у відношенні до, відповідно, передостанньої та останньої ітерацій перетворення. Очевидним є факт, що вирази для обчислення кількості операцій додавання/віднімання для ітерацій від другої до останньої мають подібну (у певній мірі, рекурентну) структуру, на основі якої можна сформувати єдиний аналітичний вираз для обчислення кількості операцій на кожній зі згаданих ітерацій. Послідовність чисел $N/9$, $N/27$ і т.д. утворює геометричну прогресію зі знаменником $1/3$, а їх сума може бути обчислена, як сума перших $\log_3 N–j$ (j=1 у випадку останньої ітерації і $j=\log_3 N–1$ у випадку другої) членів відповідної прогресії. Виконавши елементарні перетворення описаного виразу суми геометричної прогресії отримано загальний вираз (29) кількості операцій додавання/віднімання в одній ітерації для ітерацій від другої до останньої.

$$\left(\frac{N}{3}-\frac{N-3^j}{6}\right)*6+\frac{N-3^j}{6}*2=\frac{4N}{3}+2*3^{j-1}, \quad (29)$$

де $j=\log_3 N–(i-1)$, i=2,3,…,$\log_3 N$ – номер ітерації, $N=3^n$ – розмірність перетворення, n=2,3,….

На основі виразів (26) та (29) і очевидного факту, що кількість ітерацій у швидкому перетворенні (8) розмірності N рівна $\log_3 N$, можна сформувати загальний вираз (30) для обчислення сумарної кількості операцій додавання/віднімання для перетворення розмірності N. При цьому, у виразі (30) враховано, що послідовність других доданків правої частини рівності (29) при збільшенні величини j утворює геометричну прогресію з першим членом 2 та знаменником 3.

$$2N+(\log_3 N-1)\frac{4N}{3}+\frac{2*(3^{\log_3 N-1}-1)}{3-1}= \quad (30)$$

$$=N-1+\frac{4}{3}N\log_3 N.$$

З принципів формування коефіцієнтів (9) та механізмів реалізації швидких прямого та оберненого перетворень випливає, що кількість операцій множення на константу для реалізації швидкого перетворення на основі перетворення (8) розмірності N рівна N. Даний факт та вираз (30) дозволяють здійснити порівняння ефективності застосування для цифрової обробки інформації перетворення (8) з іншими ортогональними перетвореннями за критерієм операційної складності.

## Оцінка ефективності за критерієм операційної складності

Критерій операційної складності дозволяє оцінити швидкодію ортогонального перетворення незалежно від конкретної апаратної реалізації [7; 9; 15]. Оцінка здійснюється за кожним типом операцій окремо, вказуючи на переваги і недоліки з точки зору конкретного типу операцій. За даним критерієм перетворення є більш ефективним у порівнянні з іншими, якщо воно володіє меншою операційною складністю за певним типом операцій. У цифровій обробці інформації операційну складність прийнято описувати функціями (аналітичними виразами) від розмірності вхідного вектора (N) [1; 7–9; 15].

Як і у випадку критерію ступеня зменшення кореляції коефіцієнтів перетворення порівняно з елементами вхідного вектора (робота [2]) порівняння за критерієм операційної складності проведено між перетворенням (8) (Ter) та ортогональними перетвореннями Уолша-Адамара (Wal) і Хаара (Har). Результати порівнянь за критерієм операційної складності наведено у табл. 2.

Таблиця 2

Операційна складність ортогональних перетворень за типами арифметичних операцій

| Перетворення | Операційна складність за типами арифметичних операцій | |
|---|---|---|
| | додавання/ віднімання | множення на константу |
| Wal | $N\log_2 N$ | N (з урахуванням множення на 1/N) |
| Har | 2(N-1) | N |
| Ter | $N-1+\frac{4}{3}N\log_3 N$ | N |

Із табл. 2 можна зробити висновок, що за складністю відносно операції множення на константу проаналізовані перетворення мають однакову ефективність, тому більш детальному аналізу підлягає складність відносно операцій додавання/віднімання (табл. 3).

Таблиця 3

Кількість операцій додавання/віднімання для різних значень розмірності ортогональних перетворень

| N | Wal | Har | Ter |
|---|---|---|---|
| 8 | 24 | 14 | 27,19 |
| 9 | 28,53 | 16 | 32 |
| 16 | 64 | 30 | 68,84 |
| 27 | 128,38 | 52 | 134 |
| 32 | 160 | 62 | 165,60 |

Закінчення табл. 3

| | | | |
|---|---|---|---|
| 64 | 384 | 126 | 386,04 |
| 81 | 513,53 | 160 | 512 |
| 128 | 896 | 254 | 880,75 |
| 243 | 1925,74 | 484 | 1862 |
| 256 | 2048 | 510 | 1977,86 |

Дані у табл. 3 дозволяють зробити висновок про те, що перетворення (8) володіє більшою операційною складністю за операціями додавання/віднімання у порівнянні з перетворенням Хаара (в середньому на 156%) та перетворенням Уолша-Адамара при N<81 (в середньому на 9%), але меншою у порівнянні з останнім при N≥81 (в середньому на 4,5%). Це означає, що, у загальному випадку, розроблене перетворення (8) у даному вигляді не рекомендоване до використання у системах цифрової обробки інформації, для яких головним пріоритетом є швидкодія. Однак, у системах, у яких головним пріоритетом є ступінь ущільнення даних, дане перетворення рекомендоване до використання у зв'язку з результатами отриманими за критерієм ступеня зменшення кореляції коефіцієнтів перетворення порівняно з елементами вхідного вектора [2]. Окрім цього, у випадку пріоритету швидкодії, винятком із загальних рекомендацій є системи, у яких на основі перетворення Уолша-Адамара опрацьовуються послідовності даних довжиною >64 відліків, оскільки застосування перетворення (8) у таких системах дозволить збільшити швидкість функціонування системи.

## Висновки

Побудоване на основі трійкових симетричних функцій ортогональне перетворення (8) у формі матричного добутку не реалізує усі можливості спрощення процедури його обчислення, якими володіють матриці даного перетворення. Реалізувати їх дозволяє обчислення даного перетворення за принципом divide-and-conquer.

Реалізоване швидке перетворення на основі ортогонального перетворення (8) за принципом divide-and-conquer дозволило зменшити, у порівнянні з безпосереднім обчисленням у вигляді матричного добутку, операційну складність перетворення (8) за операціями типу додавання/віднімання на 81,35%, відмовитись від операцій множення двох чисел з плаваючою комою (складність за даним типом операцій зменшилась на 100%) та залишити незмінною кількість необхідних операцій множення на константу, яка дорівнює розмірності перетворення і у певній мірі може вважатись мінімально можливою для рекурентної процедури перетворення.

Отримані покращення з точки зору операційної складності дозволяють у більшій мірі реалізувати досліджені [2] переваги перетворення (8) в завданнях ущільнення даних у системах цифрової обробки інформації, для яких швидкодія функціонування не є першочерговим пріоритетом. Для систем з першочерговим пріоритетом швидкодії застосування перетворення (8) у даному вигляді є обмеженим, на що вказали результати порівняльної оцінки ефективності розробленого швидкого перетворення (8) за критерієм операційної складності (табл. 2–3).

Подальші дослідження полягають у розробці та дослідженні альтернативних шляхів реалізації швидкого перетворення на основі перетворення (8) та порівняльному оцінюванні ефективності перетворення (8) за відмінними від операційної складності та ступеня декореляції коефіцієнтів перетворення критеріями та визначенні кола задач, для вирішення яких перетворення (8) є найбільш пристосованим.

## Список літератури

1. Thompson A. The Cascading Haar Wavelet Algorithm for Computing the Walsh–Hadamard Transform / A. Thompson // IEEE Signal Processing Letters. – July 2017. – Vol. 24, No. 7. – P. 1020-1023. https://doi.org/10.1109/LSP.2017.2705247.

2. Ізмайлов А.В. Застосування ортогонального перетворення на основі трійкових симетричних функцій для цифрової обробки інформації / А.В. Ізмайлов // Методи та засоби кодування, захисту й ущільнення інформації: тези доповідей Шостої Міжнародної науково-практичної конференції, м. Вінниця, 24-25 жовтня 2017 року. – Вінниця: ВНТУ, 2017. – С. 93-96.

3. Izmailov A. Symmetric ternary functions and their application in orthogonal transforms / A. Izmailov, L. Petryshyn // 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), Kyiv, 2017. – P. 836-841. https://doi.org/10.1109/UKRCON.2017.8100364.

4. Ізмайлов А.В. Трійкові симетричні функції та їх застосування у цифровій обробці інформації / А.В. Ізмайлов, Л.Б. Петришин // Системи обробки інформації. – Харків: ХНУПС, 2016. – № 4(141). – С. 41-44.

5. Hayes B. Computing science. Third base. A reprint from American Scientist, the magazine of Sigma Xi, the Scientific Research Society, Vol. 89, No. 6. November–December 2001. – P. 490-494.

6. Jain P. Fast computation of the discrete Hartley transform / P. Jain, B. Kumar, S.B. Jain // International Journal of Circuit Theory and Applications. – 2010. – Vol. 38, No. 4. – P. 409-417. https://doi.org/10.1002/cta.574.

7. Ahmed N. Orthogonal Transforms for Digital Signal Processing / N. Ahmed, K.R. Rao // Springer-Verlag, 1975. – 263 p.

8. Блатов И.А. Оценка эффективности применения быстрого дискретного сплайнового вейвлет-преобразования для ослабления коррелированности дискретно заданных данных / И.А. Блатов, Ю.А. Герасимова // Вестник ВГТУ. – 2015. – №5. – С.34-36.

9. Wang R. Introduction to Orthogonal Transforms with Applications in Data Processing and Analysis / R. Wang. – Cambridge, 2010. –522 p.

10. Превисокова Н.В. Аналіз ефективності методу кодування інформації на основі ортогонального перетворення Ґалуа / Н.В. Превисокова // Вісник національного технічного університету «Харківський політехнічний інститут». Збірник наукових праць. Серія: Інформатика та моделювання. – №21 (1193). – Харків: НТУ «ХПІ», 2016. – С. 92-101.

11. Yaroslavsky L.P. Fast Transforms in Image Processing: Compression, Restoration, and Resampling / L.P. Yaroslavsky // Advances in Electrical Engineering. – August 2014. – Vol. 2014. – Article ID 276241. – P. 23. https://doi.org10.1155/2014/276241.

12. Minasyan S. Ternary Haar-Like Transform and Its Application in Spectral Representation of Ternary-Valued Functions / S. Minasyan, R. Stankovic, J. Astola // Computer Aided Systems Theory – EUROCAST 2009. Lecture Notes in Computer Science. – Vol 5717. – Berlin: Springer, 2009. – P. 518-525. https://doi.org/10.1007/978-3-642-04772-5_67.

13. Prasad S. Information Fusion in the Redundant-Wavelet-Transform Domain for Noise-Robust Hyperspectral Classification / S. Prasad, W. Li, J.E. Fowler, L.M. Bruce // IEEE Transactions on Geoscience and Remote Sensing. – September 2012. – Vol. 50, No. 9. – P. 3474-3486. https://doi.org/10.1109/TGRS.2012.2185053.

14. Fu Ch. Ternary recursive fast transforms properties, mutual relations, and circuit realization / Ch. Fu, B.J. Falkowski // Journal of Circuits, Systems and Computers. – April 2007. – Vol. 16, No. 02. – P. 155-168. https://doi.org/10.1142/S0218126607003459.

15. Блейхут Р.Э. Быстрые алгоритмы цифровой обработки сигналов / Р.Э. Блейхут. – М.: Мир, 1989. – 448 с.

## References

1. Thompson, A. (July 2017), The Cascading Haar Wavelet Algorithm for Computing the Walsh–Hadamard Transform, *IEEE Signal Processing Letters*, Vol. 24, No. 7, pp. 1020-1023. https://doi.org/10.1109/LSP.2017.2705247.

2. Izmailov, A.V. (2017), ”Zastosuvannia ortohonalnoho peretvorennia na osnovi triikovykh symetrychnykh funktsii dlia tsyfrovoi obrobky informatsii” [Application of orthogonal transform on the basis of symmetric ternary functions for digital information processing], *6th International Scientific Conference : Methods and Means of Encoding, Protection and Compression of Information, October 24–25, 2017*, Vinnytsia, pp. 93-96.

3. Izmailov, A. and Petryshyn, L. (2017), Symmetric ternary functions and their application in orthogonal transforms, *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, Kyiv, pp. 836-841. https://doi.org/10.1109/UKRCON.2017.8100364.

4. Izmailov, A.V. and Petryshyn, L.B. (2016), “Triikovi symetrychni funktsii ta yikh zastosuvannia u tsyfrovii obrobtsi informatsii” [Symmetric ternary functions and their application in digital information processing], *Information Processing Systems*, No. 4(141), pp. 41-44.

5. Hayes, B. (November–December 2001), Computing science. Third base. A reprint from *American Scientist, the magazine of Sigma Xi, the Scientific Research Society*, Vol. 89, No. 6, pp. 490-494.

6. Jain, P., Kumar, B. and Jain, S.B. (2010), Fast computation of the discrete Hartley transform, *International Journal of Circuit Theory and Applications*, Vol. 38, No. 4, pp. 409-417. https://doi.org/10.1002/cta.574.

7. Ahmed, N. and Rao, K.R. (1975), *Orthogonal Transforms for Digital Signal Processing*, Springer-Verlag, 263 p.

8. Blatov, I.A. and Gerasimova, Yu.A. (2015), “Otsenka effektyvnosty prymenenyia bystroho dyskretnoho splainovoho veivlet-preobrazovanyia dlia oslablenyia korrelyrovannosty dyskretno zadannykh dannykh” [Efficiency evaluation of application of fast discrete spline wawelet transformation for loosening correlation of discretely set data], *Herald of Voronezh National Technical University*, No. 5, pp. 34-36.

9. Wang, R. (2010), *Introduction to Orthogonal Transforms with Applications in Data Processing and Analysis*, Cambridge, 522p.

10. Prevysokova, N.V. (2016), “Analiz efektyvnosti metodu koduvannia informatsii na osnovi ortohonalnoho peretvorennia Galua” [Efficiency analysis of the method transform coding information on the Galois functions base], *Herald of the National Technical University "KhPI". Subject issue: Information Science and Modeling*, No. 21(1193), NTU "KhPI", Kharkiv, pp. 92-101.

11. Yaroslavsky, L.P. (August 2014), Fast Transforms in Image Processing: Compression, Restoration, and Resampling, *Advances in Electrical Engineering*, Vol. 2014, Article ID 276241, p. 23. https://doi.org/10.1155/2014/276241.

12. Minasyan, S., Stankovic, R. and Astola, J. (2009), Ternary Haar-Like Transform and Its Application in Spectral Representation of Ternary-Valued Functions, *Computer Aided Systems Theory – EUROCAST 2009. Lecture Notes in Computer Science*, Vol 5717, Springer, Berlin, pp. 518-525. https://doi.org/10.1007/978-3-642-04772-5_67.

13. Prasad, S., Li, W., Fowler, J.E. and Bruce, L.M. (September 2012), Information Fusion in the Redundant-Wavelet-Transform Domain for Noise-Robust Hyperspectral Classification, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 50, No. 9, pp. 3474-3486. https://doi.org/10.1109/TGRS.2012.2185053.

14. Fu, Ch. and Falkowski, B.J. (April 2007), Ternary recursive fast transforms properties, mutual relations, and circuit realization, *Journal of Circuits, Systems and Computers*, Vol. 16, No. 02, pp. 155-168. https://doi.org/10.1142/S0218126607003459.

15. Blahut, R.E. (1989), “*Bystrye alhorytmy tsyfrovoi obrabotky sygnalov*” [Fa*st Algorithms for Digital Signal Processing*], Mir, Moscow, p. 448.

*Надійшла до редколегії 6.08.2018*
*Схвалена до друку 11.09.2018*

***Відомості про авторів:***

**Ізмайлов Артем Вікторович**
магістр
аспірант кафедри інформатики Прикарпатського національного університету ім. Василя Стефаника,
Івано-Франківськ, Україна
https://orcid.org/0000-0002-6165-7490

**Петришин Любомир Богданович**
доктор технічних наук професор
професор кафедри
Науково-технологічного університету AGH,
Краків, Польща
завідувач кафедри Прикарпатського національного університету ім. Василя Стефаника,
Івано-Франківськ, Україна
https://orcid.org/0000-0003-4168-3891

***Information about the authors:***

**Artem Izmailov**
Master
Postgraduate Student of Vasyl Stefanyk Precarpathian National University,
Ivano-Frankivsk, Ukraine
https://orcid.org/0000-0002-6165-7490

**Lubomyr Petryshyn**
Doctor of Technical Sciences Professor
Professor of the Department of AGH University of Science and Technology,
Cracov, Poland
Head of the Department of Vasyl Stefanyk Precarpathian National University,
Ivano-Frankivsk, Ukraine
https://orcid.org/0000-0003-4168-3891

## ЦИФРОВАЯ ОБРАБОТКА ИНФОРМАЦИИ В РАССРЕДОТОЧЕННЫХ СИСТЕМАХ УПРАВЛЕНИЯ С ИСПОЛЬЗОВАНИЕМ БЫСТРОГО ОРТОГОНАЛЬНОГО ПРЕОБРАЗОВАНИЯ НА ОСНОВЕ ТРОИЧНЫХ СИММЕТРИЧНЫХ ФУНКЦИЙ

А.В. Измайлов, Л.Б. Петришин

*Рассматривается ортогональное преобразование на основе троичных симметричных функций и его эффективность по критерию операционной сложности. На основе данного ортогонального преобразования синтезировано быстрое преобразование по принципу divide-and-conquer. Проведен сравнительный анализ эффективности разработанного быстрого преобразования с ортогональными преобразованиями Уолша-Адамара и Хаара по критерию операционной сложности. Разработанное быстрое преобразование позволяет реализовать преимущества ортогонального преобразования на основе троичных симметричных функций в задачах цифровой обработки информации, связанных с уплотнением данных.*

***Ключевые слова:*** *Цифровая обработка информации, троичные симметричные функции, быстрое ортогональное преобразование, divide-and-conquer.*

## DIGITAL INFORMATION PROCESSING IN DISPERSED MANAGEMENT SYSTEMS WITH APPLICATION OF THE FAST ORTHOGONAL TRANSFORM BASED ON SYMMETRIC TERNARY FUNCTIONS

A. Izmailov, L. Petryshyn

*The paper deals with orthogonal transform based on symmetric ternary functions and its efficiency due to the criterion of operational complexity. Digital information processing (DIP) is an important part of most of the information technologies applied in different areas of economy, production, medicine, etc. Thus, effective solutions in DIP give efficiency increase in all its application areas. Orthogonal transforms play a significant role in DIP processes and therefore, the problem of synthesis of new efficient orthogonal transforms has high practical significance. The former efficiency analysis of orthogonal transform based on symmetric ternary functions due to the criterion of decorrelation degree of the transform coefficients proved its application efficiency for problems of data compression. However, implementation of this orthogonal transform straight in form of matrix multiplication has maximal possible operation complexity. At the same time transform matrices of the described transform give many opportunities for simplifying its calculation. From properties analysis of transform matrices it follows that it is possible to build a fast transform based on the given orthogonal transform using recursive calculation procedure known as divide-and-conquer principle. The described fast transform was successfully built and its efficiency was tested in comparison with Walsh-Hadamard and Haar transforms using the criterion of operational complexity.*

***Keywords:*** *Digital information processing, symmetric ternary functions, fast orthogonal transform, divide-and-conquer.*