

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ  
УНИВЕРСИТЕТ имени академика С.П.КОРОЛЕВА  
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»

**М.А. Чичева**

**Быстрые алгоритмы  
дискретного косинусного преобразования**

*Электронное учебное пособие*

САМАРА

2010

Авторы: ЧИЧЕВА Марина Александровна,

В данном пособии рассмотрены быстрые алгоритмы дискретного косинусного преобразования (ДКП) различных длин. Показано применение ДКП в различных задачах обработки изображений. Для полноты изложения приведены основные сведения из теории быстрых алгоритмов дискретных преобразований. На ДКП основаны известные методы компрессии изображений (стандарты JPEG, MPEG и др.), эффективные методы фильтрации, методы извлечения признаков и т.д. Широкое применение ДКП обусловлено уникальным сочетанием его позитивных качеств. Во-первых, базисные функции ДКП для многих сигналов, описываемых моделями стационарных случайных процессов, близки к базисным функциям преобразования Карунена-Лоэва (ПКЛ), т.е. позволяют описывать сигнал с заданной точностью минимальным числом спектральных компонент. Во-вторых, в отличие от ПКЛ, для ДКП существуют быстрые алгоритмы вычисления. В-третьих, так называемое "четное периодическое" продолжение сигнала, предполагаемое в ДКП, позволяет устранить вредные краевые эффекты при блочной обработке изображений.

Пособие предназначено для магистров по направлению 010400.68 "Прикладная математика и информатика", обучающихся по программе «Математические и компьютерные методы обработки изображений и геоинформатики».

# 1. ПРЕДВАРИТЕЛЬНЫЕ СВЕДЕНИЯ ИЗ ТЕОРИИ БЫСТРЫХ АЛГОРИТМОВ ДИСКРЕТНЫХ ОРТОГОНАЛЬНЫХ ПРЕОБРАЗОВАНИЙ

## 1.1. Постановка задачи, основные идеи

Дискретное косинусное преобразование (ДКП):

$$\hat{x}(m) = \lambda_m \sum_{n=0}^{N-1} x(n) \cos\left(\frac{\pi(n + \frac{1}{2})m}{N}\right), \quad (m = 0, \dots, N-1), \quad (1.1)$$

где  $\lambda_m$  - нормирующие множители [1]:

$$\lambda_m = \begin{cases} \sqrt{\frac{1}{N}}, & \text{при } m = 0, \\ \sqrt{\frac{2}{N}}, & \text{при } m \neq 0, \end{cases} \quad (1.2)$$

является одним из основных дискретных ортогональных преобразований (ДОП), используемых в обработке изображений. В частности, ДКП длины  $N=8$  является базовым преобразованием для целого ряда современных стандартов кодирования (JPEG, MPEG, ITU-T [25, 31]). Однако метод блочного кодирования с преобразованием (на котором основываются перечисленные стандарты) не в полной мере учитывает особенности *конкретного* изображения. Попытки построения *адаптивных* алгоритмов кодирования на основе ДКП [13, 21, 22, 28] связаны, обычно, с резким увеличением вычислительной сложности ДКП при возрастании длины преобразования. Это связано с тем, что быстрые алгоритмы (БА) ДКП при  $N > 8$  изучены в значительно меньшей степени, чем ДКП длины 8, которым посвящено множество работ [1, 9, 20, 25, 29].

Большинство из известных БА ДКП, синтезированных для преобразований произвольных длин обладают относительно низкой *асимптотической* арифметической сложностью, но не учитывают реализационной специфики "коротких" длин: относительно небольшого числа различных значений базисных функций, высоких требований к *структурной* простоте, которая при небольшой длине преобразования является определяющей характеристикой быстродействия и т.д.

Например, в [1] предлагается метод сведения ДКП к дискретному преобразованию Фурье (ДПФ) вещественной последовательности двойной длины:

$$\tilde{x}(m) = \frac{1}{2} \omega^{m/2} \sum_{k=0}^{2N-1} y(k) \omega^{mk}, \quad \omega = \exp\left\{\frac{2\pi i}{2N}\right\} \quad (1.3)$$

где

$$y(k) = \begin{cases} x(k) & \text{при } 0 \leq k \leq N-1 \\ x(2N-k-1) & \text{при } N \leq k \leq 2N-1 \end{cases}.$$

Вычисление преобразования (1.3) для  $N=8$  и при использовании одного из лучших БА ДПФ (сплит-радикс алгоритма вещественного ДПФ [14]) длины 16 требует согласно оценкам работы [18] 34 операции вещественного умножения и 56 операций вещественного сложения, в то время как специфический алгоритм работы [9] требует всего 12 умножений и 29 сложений. Алгоритмы косинусного преобразования "нестандартных" длин  $N \neq 2^k$  [5, 12] также не ориентированы на работу с "короткими" ДКП. В третьем разделе настоящего пособия будут показаны высокоскоростные БА ДКП для обработки двумерных массивов малых объемов (использование блоков  $N \times N$  при  $8 \leq N < 16$ ).

Снижение вычислительной сложности достигается с помощью нового подхода к синтезу алгоритмов, связанного с интерпретацией вычисления ДКП как операций в ассоциированных алгебраических структурах. В качестве таких структур рассматриваются коммутативные конечномерные алгебры малых размерностей. И основной теоретической задачей раздела является анализ связи между алгебраическими характеристиками значений базисных функций ДКП и блочной структурой матрицы преобразования (1.1). А собственно синтез быстрых алгоритмов ДКП сводится к последовательной реализации двух основных идей.

### ***Первая основная идея.***

*(а) Матрица ДКП имеет блочную структуру. Результат умножения такой матрицы на входной вектор сводится к умножению векторов из подпространств сигнального пространства на матрицы меньших размеров со специфическими свойствами "симметрии". Поэтому, шаг 1, давайте считать, что на входной сигнал*

действует линейный оператор, действие которого преобразует сигнал в вектор координат конечномерной алгебры относительно некоторого базиса.

(b) Умножение подматриц на векторы соответствующих подпространств эквивалентно умножениям элементов некоторых конечномерных алгебр. Поэтому, шаг 2, и будем интерпретировать умножение преобразованного вектора на матрицу ДКП как умножение элементов алгебры.

(c) В большинстве рассматриваемых случаев эти алгебры являются групповыми алгебрами циклических групп или их фактор-алгебрами. Умножение элементов таких алгебр эквивалентно умножению в полиномиальных кольцах (или циклической свертке).

Поэтому, шаг 3, будем использовать для умножения элементов алгебры алгоритмы быстрого вычисления циклических сверток.

Перепишем равенство (1.1) в форме

$$\hat{x}(m) = \lambda_m \frac{1}{2} \sum_{n=0}^{N-1} x(n) \left( \exp\left(\frac{2\pi(2n+1)m}{4N}\right) + \exp\left(-\frac{2\pi(2n+1)m}{4N}\right) \right).$$

Формально, в правой части фигурируют комплексные корни степени  $4N$ . В действительности, степень алгебраичности значений функции

$$\cos\left(\frac{2\pi(2n+1)m}{4N}\right) = \frac{1}{2} \left( \exp\left(\frac{2\pi(2n+1)m}{4N}\right) + \exp\left(-\frac{2\pi(2n+1)m}{4N}\right) \right) \quad (1.4)$$

над  $\mathbf{Q}$  меньше в два раза, по крайней мере, и значения функций (1.4) при различных  $m, n$  связаны полиномиальными соотношениями.

### **Вторая основная идея.**

Давайте выберем такой "экзотический" базис в поле алгебраических чисел, которому принадлежат значения функций (1.4), чтобы умножения полиномов, представляющих значения функций (1.4) в этом базисе, реализовывались по возможности наиболее простым образом в предположении, что проблема перевода входных и выходных данных из этого "экзотического" базиса в обычную позиционную систему счисления и обратно удовлетворительно решена аппаратными средствами.

## 1.2. Сложность операции умножения в конечномерных алгебрах

Рассмотрим некоторые  $d$ -мерные алгебры  $\mathbb{A}$  над  $\mathbb{R}$  с базисом  $\{1, \mathbf{e}_1, \dots, \mathbf{e}_{d-1}\}$ .

Типичный элемент  $\mathbf{a} \in \mathbb{A}$  записывается в виде

$$\mathbf{a} = \alpha_0 \cdot 1 + \sum_{j=1}^{d-1} \alpha_j \mathbf{e}_j .$$

Сложение элементов выполняется покомпонентно, а умножение индуцируется заданными правилами умножения базисных элементов и определяет структуру алгебры на векторном пространстве  $\mathbb{R}^d$ .

В пособии рассматриваются следующие конечномерные алгебры.

1. Двумерная алгебра  $\mathcal{A}_1^{(2)}$  с базисом  $\{1, \mathbf{e}_1\}$  и правилами умножения базисных элементов:

$$1^2 = 1 , \mathbf{e}_1^2 = -1 .$$

Эта алгебра изоморфна алгебра  $\mathbb{C}$  комплексных чисел:

$$\mathcal{A}_1^{(2)} \cong \mathbb{C}$$

2. Двумерная алгебра  $\mathcal{A}_2^{(2)}$  с базисом  $\{1, \mathbf{e}_1\}$  и правилами умножения базисных элементов:

$$1^2 = 1 , \mathbf{e}_1^2 = 1 .$$

Эта алгебра "дуальных" чисел [4, 27] изоморфна прямой сумме  $\mathbb{R} \oplus \mathbb{R}$ :

$$\mathcal{A}_2^{(2)} \cong \mathbb{R} \oplus \mathbb{R}$$

3. Трехмерная алгебра  $\mathcal{A}^{(3)}$  с базисом  $\{1, \mathbf{e}_1, \mathbf{e}_2\}$  и правилами умножения базисных элементов:

$$\mathbf{e}_1^2 = \mathbf{e}_2 , \mathbf{e}_2^2 = -\mathbf{e}_1 , \mathbf{e}_1 \mathbf{e}_2 = \mathbf{e}_2 \mathbf{e}_1 = -1 .$$

4. Четырехмерная алгебра  $\mathcal{A}_1^{(4)}$  с базисом  $\{1, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$  и правилами умножения базисных элементов:

$$\mathbf{e}_1^2 = -\mathbf{e}_2, \quad \mathbf{e}_2^2 = -1, \quad \mathbf{e}_3^2 = \mathbf{e}_2,$$

$$\mathbf{e}_1\mathbf{e}_2 = \mathbf{e}_2\mathbf{e}_1 = \mathbf{e}_3, \quad \mathbf{e}_2\mathbf{e}_3 = \mathbf{e}_3\mathbf{e}_2 = -\mathbf{e}_1, \quad \mathbf{e}_1\mathbf{e}_3 = \mathbf{e}_3\mathbf{e}_1 = 1.$$

5. Четырехмерная алгебра  $\mathcal{A}_2^{(4)}$  с базисом  $\{1, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$  и правилами умножения базисных элементов:

$$\mathbf{e}_1^2 = \mathbf{e}_2, \quad \mathbf{e}_2^2 = 1, \quad \mathbf{e}_3^2 = \mathbf{e}_2,$$

$$\mathbf{e}_1\mathbf{e}_2 = \mathbf{e}_2\mathbf{e}_1 = \mathbf{e}_3, \quad \mathbf{e}_2\mathbf{e}_3 = \mathbf{e}_3\mathbf{e}_2 = \mathbf{e}_1, \quad \mathbf{e}_1\mathbf{e}_3 = \mathbf{e}_3\mathbf{e}_1 = 1.$$

6. Четырехмерная алгебра  $\mathcal{A}_3^{(4)}$  с базисом  $\{1, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$  и правилами умножения базисных элементов:

$$\mathbf{e}_1^2 = -1, \quad \mathbf{e}_2^2 = 1, \quad \mathbf{e}_3^2 = -1,$$

$$\mathbf{e}_1\mathbf{e}_2 = \mathbf{e}_2\mathbf{e}_1 = -\mathbf{e}_3, \quad \mathbf{e}_2\mathbf{e}_3 = \mathbf{e}_3\mathbf{e}_2 = -\mathbf{e}_1, \quad \mathbf{e}_1\mathbf{e}_3 = \mathbf{e}_3\mathbf{e}_1 = \mathbf{e}_2.$$

Следующая Лемма 1.1. устанавливает связь сложности арифметических операций в рассматриваемых алгебрах и легко проверяемыми правилами умножения в полиномиальных кольцах. Отметим, что для последних известны (см., например, [8]) *неулучшаемые* оценки сложности реализации таких операций. Вследствие этого, справедливо следующее утверждение.

**Лемма 1.1.**

*а) Умножение элементов*

$$(\alpha_0 + \alpha_1\mathbf{e}_1), (\beta_0 + \beta_1\mathbf{e}_1) \in \mathcal{A}_1^{(2)}$$

*равносильно умножению полиномов*

$$(\alpha_0 + \alpha_1 t)(\beta_0 + \beta_1 t) \pmod{(t^2 + 1)}$$

*и требует 3 умножений и 3 сложений.*

б) Умножение элементов

$$(\alpha_0 + \alpha_1 \mathbf{e}_1), (\beta_0 + \beta_1 \mathbf{e}_1) \in \mathcal{A}_2^{(2)}$$

равносильно умножению полиномов

$$(\alpha_0 + \alpha_1 t)(\beta_0 + \beta_1 t) \pmod{(t^2 - 1)}$$

и требует 2 умножений и 4 сложений.

в) Умножение элементов

$$(\alpha_0 + \alpha_1 \mathbf{e}_1 + \alpha_2 \mathbf{e}_2), (\beta_0 + \beta_1 \mathbf{e}_1 + \beta_2 \mathbf{e}_2) \in \mathcal{A}^{(3)}$$

равносильно умножению полиномов

$$(\alpha_0 + \alpha_1 t + \alpha_2 t^2)(\beta_0 + \beta_1 t + \beta_2 t^2) \pmod{(t^3 + 1)}$$

и требует 4 умножений и 14 сложений.

г) Умножение элементов

$$(\alpha_0 + \alpha_1 \mathbf{e}_1 + \alpha_2 \mathbf{e}_2 + \alpha_3 \mathbf{e}_3) (\beta_0 + \beta_1 \mathbf{e}_1 + \beta_2 \mathbf{e}_2 + \beta_3 \mathbf{e}_3) \in \mathcal{A}_1^{(4)}$$

равносильно умножению полиномов

$$(\alpha_0 + \alpha_1 t - \alpha_2 t^2 - \alpha_3 t^3)(\beta_0 + \beta_1 t - \beta_2 t^2 - \beta_3 t^3) \pmod{(t^4 + 1)}$$

и требует 9 умножений и 15 сложений.

д) Умножение элементов

$$(\alpha_0 + \alpha_1 \mathbf{e}_1 + \alpha_2 \mathbf{e}_2 + \alpha_3 \mathbf{e}_3), (\beta_0 + \beta_1 \mathbf{e}_1 + \beta_2 \mathbf{e}_2 + \beta_3 \mathbf{e}_3) \in \mathcal{A}_2^{(4)} \text{ равносильно умножению}$$

полиномов

$$(\alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3)(\beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3) \pmod{(t^4 - 1)}$$

и требует 5 умножений и 15 сложений.

**Доказательство.** Равносильность умножений в алгебрах и полиномиальных кольцах проверяется непосредственно. Сложность операций умножения последних показана, например, в [8].

**Лемма 1.2.** Алгебра  $\mathcal{A}_3^{(4)}$  изоморфна прямой сумме  $\mathbb{C} \oplus \mathbb{C}$ :

$$\mathbb{C} \oplus \mathbb{C} = \left\{ (z_1, z_2) : z_1 = a_1 + b_1 i_1, z_2 = a_2 + b_2 i_2, \right. \\ \left. i_1^2 = i_2^2 = -1, a_1, a_2, b_1, b_2 \in \mathbb{R} \right\}$$

**Доказательство.** Элементы

$$\mathbf{E}_0 = (1, 1), \quad \mathbf{E}_1 = (i_1, -i_2), \quad \mathbf{E}_2 = (-1, 1), \quad \mathbf{E}_3 = (i_1, i_2) \in \mathbb{C} \oplus \mathbb{C}$$

образуют базис алгебры  $\mathbb{C} \oplus \mathbb{C}$  над  $\mathbb{R}$ . Отображение  $\varphi$ , определенное для базисных элементов алгебр  $\mathcal{A}_3^{(4)}$  и  $\mathbb{C} \oplus \mathbb{C}$  как

$$\varphi: \mathbf{e}_1 \mapsto (i_1, -i_2), \quad \varphi: \mathbf{e}_2 \mapsto (-1, 1), \\ \varphi: \mathbf{e}_3 \mapsto (i_1, i_2), \quad \varphi: 1 \mapsto (1, 1)$$

продолжается  $\mathbb{R}$ -линейно до изоморфизма соответствующих четырехмерных алгебр.

Линейный оператор  $L$ , определенный на пространстве алгебры  $\mathbb{C} \oplus \mathbb{C}$  образами базисных элементов

$$L(\mathbf{E}_0) = \frac{1}{2}(\mathbf{E}_0 - \mathbf{E}_2), \quad L(\mathbf{E}_1) = \frac{1}{2}(\mathbf{E}_1 + \mathbf{E}_3), \\ L(\mathbf{E}_2) = \frac{1}{2}(\mathbf{E}_0 + \mathbf{E}_2), \quad L(\mathbf{E}_3) = \frac{1}{2}(\mathbf{E}_3 - \mathbf{E}_1),$$

преобразует базис  $\{\mathbf{E}_0, \mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3\}$  в "стандартный" базис  $\{\sigma_0, \sigma_1, \sigma_2, \sigma_3\}$  алгебры  $\mathbb{C} \oplus \mathbb{C}$ , рассматриваемой как четырехмерная  $\mathbb{R}$ -алгебра:

$$\sigma_0 = L(\mathbf{E}_0) = (1 + 0 \cdot i_1, 0 + 0 \cdot i_2), \quad \sigma_1 = L(\mathbf{E}_1) = (0 + i_1, 0 + 0 \cdot i_2), \\ \sigma_2 = L(\mathbf{E}_2) = (0 + 0 \cdot i_1, 1 + 0 \cdot i_2), \quad \sigma_3 = L(\mathbf{E}_3) = (0 + 0 \cdot i_1, 0 + i_2).$$

**Следствие 1.1.** Умножение постоянного элемента

$\mathbf{a} = \alpha_0 + \alpha_1 \mathbf{e}_1 + \alpha_2 \mathbf{e}_2 + \alpha_3 \mathbf{e}_3$  на элемент  $\mathbf{b} = \beta_0 + \beta_1 \mathbf{e}_1 + \beta_2 \mathbf{e}_2 + \beta_3 \mathbf{e}_3$  алгебры  $\mathcal{A}_3^{(4)}$  требует 6 вещественных умножений и 10 вещественных сложений.

**Доказательство.** Сложность рассматриваемого умножения складывается из умножения двух пар комплексных чисел (элементов алгебры  $\mathbb{C} \oplus \mathbb{C}$  в

базисе  $\{\sigma_0, \sigma_1, \sigma_2, \sigma_3\}$  ) и сложности преобразования элементов при замене базиса  $\{\mathbf{E}_0, \mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3\}$  на базис  $\{\sigma_0, \sigma_1, \sigma_2, \sigma_3\}$  . Для умножения двух пар комплексных чисел достаточно 3+3 умножений и 3+3 сложений [8]. Для преобразования элементов переменного вектора достаточно 4 сложений.

Другие необходимые примеры алгебр рассматриваются в разделе по мере их использования.

### 1.3. Основные схемы редукции вспомогательного дискретного преобразования Фурье (ДПФ)

#### 1.3.1. Декомпозиция Кули-Тьюки "по основанию 2"

Пусть  $x(n) \in \mathbb{C}$  есть  $N$ -периодическая последовательность,  $N = 2^k$ ,  $X(m)$  - её дискретный спектр Фурье:

$$X(m) = \sum_{n=0}^{N-1} x(n) \omega^{mn}, \quad \omega = \exp\left\{\frac{2\pi i}{N}\right\}, \quad 0 \leq m \leq N-1, \quad (1.5)$$

Сумма в правой части соотношения (1.5) может быть представлена для  $0 \leq m \leq \frac{N}{2}-1$  в виде двух сумм длиной  $\frac{N}{2}$  :

$$\begin{aligned} X(m) &= \sum_{n=0}^{\frac{N}{2}-1} x(2n) (\omega^2)^{mn} + \omega^m \sum_{n=0}^{\frac{N}{2}-1} x(2n+1) (\omega^2)^{mn} \\ &= X_0(m) + \omega^m X_1(m) \end{aligned} \quad (1.6)$$

Здесь

$$X_0(m) = \sum_{n=0}^{\frac{N}{2}-1} x(2n) (\omega^2)^{mn}, \quad X_1(m) = \sum_{n=0}^{\frac{N}{2}-1} x(2n+1) (\omega^2)^{mn}$$

- спектры Фурье  $\frac{N}{2}$ -периодических подпоследовательностей. Таким образом, ДПФ длиной  $N$  сведено к двум преобразованиям Фурье длиной  $\frac{N}{2}$  и к  $\frac{N}{2}$  дополнительным умножениям на степени  $\omega$  для  $0 \leq m \leq \frac{N}{2}-1$ . Так как  $\omega^{\frac{N}{2}} = -1$ , то вычисление  $F(m)$  для  $\frac{N}{2} \leq m \leq N-1$  выполняется без дополнительных умножений:

$$X(m^* + N/2) = X_0(m^*) - \omega^{m^*} X_1(m^*), \quad 0 \leq m^* \leq N/2 - 1. \quad (1.7)$$

Мультипликативная  $M(N)$  и аддитивная  $A(N)$  сложность такого алгоритма равны, соответственно:

$$M(N) \leq \frac{3}{2} N \log_2 N - \frac{9}{2} N, \quad A(N) \leq \frac{7}{2} N \log_2 N - \frac{9}{2} N, \quad (1.8)$$

Изложенный алгоритм принято называть быстрым преобразованием Фурье (БПФ) по основанию 2.

### 1.3.2. Декомпозиция Кули-Тьюки "по основанию 4"

Аналогичным образом строится алгоритм БПФ "по основанию 4" при  $N = 4^k$ . Сумма для  $X(m)$  в (1.5) разбивается на четыре части:

$$\begin{aligned} X(m) &= \sum_{n=0}^{\frac{N}{4}-1} x(4n) (\omega^4)^{mn} + \omega^m \sum_{n=0}^{\frac{N}{4}-1} x(4n+1) (\omega^4)^{mn} + \\ &+ \omega^{2m} \sum_{n=0}^{\frac{N}{4}-1} x(4n+2) (\omega^4)^{mn} + \omega^{3m} \sum_{n=0}^{\frac{N}{4}-1} x(4n+3) (\omega^4)^{mn} = \\ &= X_0(m) + \omega^m X_1(m) + \omega^{2m} X_2(m) + \omega^{3m} X_3(m) \\ & \quad 0 \leq m \leq N/4 - 1. \end{aligned} \quad (1.9)$$

Соотношение (1.8) редуцирует вычисление ДПФ (1.5) к вычислению четырех ДПФ длиной  $N/4$  и к  $3N/4$  дополнительным умножениям на степени  $\omega$ . Так как при стандартном машинном представлении комплексных чисел умножения на степени мнимой единицы  $i$  являются тривиальными, значения спектра при  $N/4 \leq m \leq N-1$  вычисляются без дополнительных умножений следующим образом:

$$\begin{pmatrix} X(m^*) \\ X\left(\frac{N}{4} + m^*\right) \\ X\left(\frac{N}{2} + m^*\right) \\ X\left(\frac{3N}{4} + m^*\right) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} \begin{pmatrix} X_0(m^*) \\ \omega^{m^*} X_1(m^*) \\ \omega^{2m^*} X_2(m^*) \\ \omega^{3m^*} X_3(m^*) \end{pmatrix}, \quad (1.10)$$

$$0 \leq m^* \leq N/4 - 1.$$

Оценки вычислительной сложности такого алгоритма имеют вид:

$$M(N) \leq \frac{9}{8} N \log_2 N - \frac{13}{4} N, \quad A(N) \leq \frac{25}{8} N \log_2 N - \frac{13}{4} N. \quad (1.11)$$

### 1.3.3. Декомпозиция Кули-Тьюки с расщеплением основания (сплит-радикс алгоритм)

Пусть  $N = 2^k$ , тогда преобразование (1.5) для  $0 \leq m \leq \frac{N}{4} - 1$  может быть записано в следующем виде [3]:

$$\begin{aligned} X(m) = & \sum_{n=0}^{\frac{N}{2}-1} x(2n) (\omega^2)^{mn} + \omega^m \sum_{n=0}^{\frac{N}{4}-1} x(4n+1) (\omega^4)^{mn} + \\ & + \omega^{3m} \sum_{n=0}^{\frac{N}{4}-1} x(4n+3) (\omega^4)^{mn} \end{aligned}$$

Здесь ДПФ длиной  $N$  сведено к одному ДПФ длиной  $\frac{N}{2}$ , двум ДПФ длиной  $\frac{N}{4}$  и к  $2\frac{N}{4}$  дополнительным умножениям на степени  $\omega$ . Вычисление  $X(m)$  для  $\frac{N}{4} \leq m \leq N-1$  выполняются без дополнительных умножений:

$$\begin{pmatrix} X(m^*) \\ X\left(\frac{N}{4} + m^*\right) \\ X\left(\frac{N}{2} + m^*\right) \\ X\left(\frac{3N}{4} + m^*\right) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & i & 1 & -i \\ 1 & -1 & 0 & -1 \\ 0 & -i & 1 & i \end{pmatrix} \begin{pmatrix} X_0(m^*) \\ \omega^{m^*} X_1(m^*) \\ X_0\left(\frac{N}{4} + m^*\right) \\ \omega^{3m^*} X_3(m^*) \end{pmatrix},$$

$$0 \leq m^* \leq \frac{N}{4} - 1$$

где

$$\begin{aligned} X_0(m) &= \sum_{n=0}^{\frac{N}{2}-1} x(2n) (\omega^2)^{mn}, \\ X_1(m) &= \sum_{n=0}^{\frac{N}{4}-1} x(4n+1) (\omega^4)^{mn}, \quad X_3(m) = \sum_{n=0}^{\frac{N}{4}-1} x(4n+3) (\omega^4)^{mn}. \end{aligned}$$

Оценки вычислительной сложности для этого алгоритма равны:

$$M(N) \leq N \log_2 N - 3N, \quad A(N) \leq 3N \log_2 N - 3N. \quad (1.12)$$

### 1.3.4. Декомпозиция ДПФ Гуда-Томаса

Другим известным способом быстрого вычисления ДПФ является декомпозиция Гуда-Томаса [17, 30], применяемая в тех случаях, когда длина преобразования  $N = P \cdot Q$ , где  $P$  и  $Q$  - взаимно просты.

Пусть  $\alpha = \exp\left\{\frac{2\pi i}{P}\right\}$ ,  $\beta = \exp\left\{\frac{2\pi i}{Q}\right\}$  - первообразные комплексные корни из единицы степени  $P$  и  $Q$ , соответственно. Представим индексы входной и выходной последовательности в виде:

$$\begin{cases} n \equiv Pn_1 + Qn_2 \\ m \equiv P a m_1 + Q b m_2 \end{cases}, \quad (1.13)$$

где  $a$  и  $b$  определяются из условий

$$\begin{cases} P a \equiv 1 \pmod{Q} \\ Q b \equiv 1 \pmod{P} \end{cases}. \quad (1.14)$$

После введения обозначений  $\tilde{x}(n_1, n_2) = x(Pn_1 + Qn_2)$ ,

$\tilde{X}(m_1, m_2) = X(P a m_1 + Q b m_2)$  соотношение (1.5) примет вид:

$$\begin{aligned} \tilde{X}(m_1, m_2) &= \sum_{n_1=0}^{Q-1} \sum_{n_2=0}^{P-1} \tilde{x}(n_1, n_2) \omega^{(Pn_1 + Qn_2)(P a m_1 + Q b m_2)} = \\ &= \sum_{n_1=0}^{Q-1} \sum_{n_2=0}^{P-1} \tilde{x}(n_1, n_2) \omega^{P^2 a n_1 m_1 + P Q (b n_1 m_2 + a n_2 m_1) + Q^2 b n_2 m_2}. \end{aligned}$$

В силу (1.14) справедливо равенство:

$$\begin{aligned} \tilde{X}(m_1, m_2) &= \sum_{n_1=0}^{Q-1} \sum_{n_2=0}^{P-1} \tilde{x}(n_1, n_2) \omega^{P n_1 m_1} \omega^{P Q (b n_1 m_2 + a n_2 m_1)} \omega^{Q n_2 m_2} = \\ &= \sum_{n_1=0}^{Q-1} \sum_{n_2=0}^{P-1} \tilde{x}(n_1, n_2) \beta^{n_1 m_1} \alpha^{n_2 m_2}, \end{aligned} \quad (1.15)$$

где

$$\beta = \exp\left\{\frac{2\pi i P}{N}\right\} = \exp\left\{\frac{2\pi i}{Q}\right\} = \omega^P,$$

$$\alpha = \exp\left\{\frac{2\pi i Q}{N}\right\} = \exp\left\{\frac{2\pi i}{P}\right\} = \omega^Q.$$

Из последнего равенства (1.15) следует:

$$\tilde{X}(m_1, m_2) = \sum_{n_2=0}^{P-1} \left( \sum_{n_1=0}^{Q-1} \tilde{x}(n_1, n_2) \beta^{n_1 m_1} \right) \alpha^{n_2 m_2} \quad (1.16)$$

или

$$X(Pam_1 + Qbm_2) = \sum_{n_2=0}^{P-1} \left( \sum_{n_1=0}^{Q-1} x(Pn_1 + Qn_2) \beta^{n_1 m_1} \right) \alpha^{n_2 m_2}. \quad (1.17)$$

Так как для описанного шага декомпозиции справедливо неравенство:

$$M(N) = M(P) \cdot Q + M(Q) \cdot P \leq QP^2 + PQ^2 = N(P+Q) < N^2, \quad (1.18)$$

то применение этого приема тем эффективнее, чем на большее число взаимно простых сомножителей разлагается число  $N$ .

### 1.3.5. Декомпозиция Кули-Тьюки "по основанию $p$ "

Пусть  $N = p^r$ , преобразование входной последовательности  $x(n)$  определено соотношением (1.7). Тогда при  $m = 0, 1, \dots, \frac{N}{p} - 1$  спектр  $X(m)$  может быть представлен в виде:

$$\begin{pmatrix} X(m) \\ X\left(m + \frac{N}{p}\right) \\ X\left(m + 2\frac{N}{p}\right) \\ \dots \\ X\left(m + (p-1)\frac{N}{p}\right) \end{pmatrix} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & \gamma & \dots & \gamma^{p-1} \\ 1 & \gamma^2 & \dots & \gamma^{2(p-1)} \\ \dots & \dots & \dots & \dots \\ 1 & \gamma^{p-1} & \dots & \gamma^{(p-1)^2} \end{pmatrix} \begin{pmatrix} X_0(m) \\ w^m X_1(m) \\ w^{2m} X_2(m) \\ \dots \\ w^{(p-1)m} X_{p-1}(m) \end{pmatrix}, \quad (1.19)$$

где  $X_j(m)$  ( $j = 0, 1, \dots, p-1$ ) есть ДПФ длиной  $\frac{N}{p}$ :

$$X_j(m) = \sum_{n=0}^{N-1} x(pn+j) (\omega^p)^{mn}, \quad (1.20)$$

$\omega = \exp\left\{2\pi i/N\right\}$ ,  $\gamma = \exp\left\{2\pi i/p\right\}$  - первообразные корни из единицы степени  $N$  и  $p$  соответственно.

Равенства (1.19), (1.20) сводят вычисление ДПФ длиной  $N$  к вычислению  $p$  раз ДПФ длиной  $N/p$  с последующей последовательной редукцией к вычислению одноточечных преобразований. Спецификой случая  $N = p^r$  при  $p \neq 2, 4$  является наличие в правой части (1.19) умножений на степени константы  $\gamma$ , что увеличивает вычислительную сложность алгоритма по сравнению с БПФ по основанию 2 и 4 [2, 4], где аналогичные умножения тривиальны (умножение на  $\pm 1, \pm i$ ).

В работе [11] предложено специальное представление данных (значений преобразуемого сигнала  $x(n)$  и комплексных параметров) в, так называемых,  $\gamma$ -кодах, которое позволяет сделать эти умножения тривиальными.

Пусть  $\gamma = \exp\left\{2\pi i/p\right\}$  - первообразный комплексный корень степени  $p$  из единицы. Тогда для комплексного числа  $c$  наряду с обычной алгебраической формой представления  $c = a + bi$  возможна и форма

$$c = c_1\gamma + c_2\gamma^2 + \dots + c_{p-1}\gamma^{p-1} = a + bi, \quad (1.21)$$

где вещественные  $c_1, \dots, c_{p-1}$  связаны с вещественными  $a, b$  соотношениями

$$\begin{cases} a = \sum_{k=1}^{p-1} c_k \cos 2\pi k/p, \\ b = \sum_{k=1}^{p-1} c_k \sin 2\pi k/p. \end{cases}$$

Упорядоченный набор из  $(p-1)$  числа  $(c_1, \dots, c_{p-1})$ , ассоциированный с представлением  $c$  в форме (1.21), будем называть  $\gamma$ -кодом числа  $z$ .

Арифметические действия над комплексными числами индуцируют правила действий над кодами. Сложение чисел в  $\gamma$ -кодах производится покомпонентно,

умножение чисел в  $\gamma$ -кодах сводится к нахождению циклической свертки  $\gamma$ -кодов. Умножения на  $\gamma, \gamma^2, \dots, \gamma^{p-1}$  выполняются с помощью  $\gamma$ -кодов без вещественных умножений и сводятся лишь к смене знака, суммированию и перестановке части компонент кода [11].

Так как вычисление циклической свертки произвольной длины есть типичная (и непростая) задача цифровой обработки сигналов, то в настоящей главе мы ограничиваемся рассмотрением случая  $p=3$ , для которого переход к  $\gamma$ -кодам наиболее эффективен и нагляден.

### 1.3.6. Алгоритм одномерного ДПФ при длине преобразования $N=3^k$

Пусть  $p=3, N=3^k$ ,

$$\gamma = \exp\left(\frac{2\pi j}{3}\right) = \frac{1}{2}(-1 + i\sqrt{3}), \quad \bar{\gamma} = \frac{1}{2}(-1 - i\sqrt{3}),$$

тогда равенство (1.21) примет вид (см. рис. 1.1):

$$c = a + bi = x\gamma + y\bar{\gamma},$$

$$\text{где } x = \left(\frac{b/\sqrt{3}}{\sqrt{3}}\right) - a, \quad y = \left(-\frac{b/\sqrt{3}}{\sqrt{3}}\right) - a.$$

Арифметические операции над кодами определяются равенствами:

$$(x, y) + (u, v) = (x + u, y + v), \quad (1.22)$$

$$(x, y) \cdot (u, v) = ((y - x)(v - u) - xu, (y - x)(v - u) - yv).$$

Поэтому сложность операций сложения и умножения в кодах совпадает со сложностью сложения и умножения комплексных чисел, а именно, сложение в кодах реализуется при помощи двух вещественных сложений, умножение в кодах реализуется через три вещественных умножения и три вещественных сложения (как обычно, считается, что сложения компонент кода базисных функций выполнены заранее).

Умножение на числа  $\gamma$  и  $\bar{\gamma}$ , имеющие коды (1,0) и (0,1), соответственно, определяется равенствами:

$$\begin{aligned}(1,0) \cdot (u, v) &= (-v, u - v) \\ (0,1) \cdot (u, v) &= (v - u, -u) \end{aligned}$$

и не содержит нетривиальных вещественных умножений.

Этот факт позволяет снизить вычислительную сложность алгоритма БПФ именно благодаря простой реализации умножений на степени  $\gamma$ , удельный вклад которых в быстрый алгоритм ДПФ, реализуемый на основе декомпозиции (1.19) весьма высок.

При  $p=3$  соотношение (1.19) принимает вид:

$$\begin{pmatrix} X(m) \\ X(m + N/3) \\ X(m + 2N/3) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & \gamma & \bar{\gamma} \\ 1 & \bar{\gamma} & \gamma \end{pmatrix} \begin{pmatrix} X_0(m) \\ \omega^m X_1(m) \\ \omega^{2m} X_2(m) \end{pmatrix} \quad (1.23)$$

и нетрудно показать, что оценки вычислительной сложности такого алгоритма для вещественного сигнала имеют вид [7, 11]:

$$M(N) \leq N \log_3 N - N, \quad A(N) \leq 3N \log_3 N + \frac{N}{3}. \quad (1.24)$$

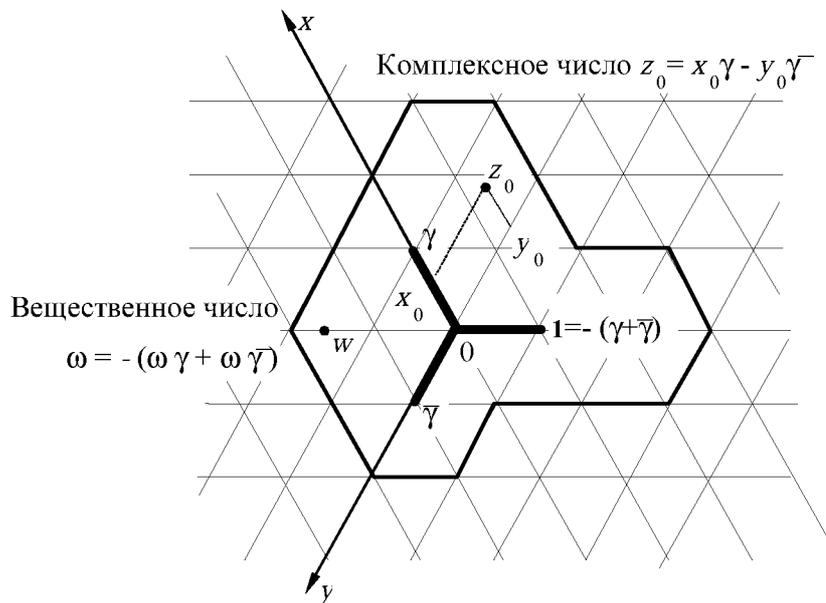


Рис.1.1. Представление комплексных чисел в  $\gamma$ -кодах

## 2. АЛГОРИТМЫ ОДНОМЕРНОГО ДКП

### 2.1. Традиционный способ вычисления ДКП

Описанный в [1] быстрый алгоритм сводит ДКП

$$\hat{x}(m) = \sum_{n=0}^{N-1} x(n) \cos\left(\pi \frac{(n+1/2)m}{N}\right)$$

к вычислению ДПФ вещественной последовательности длины  $2N$ :

$$\begin{aligned} \hat{x}(m) &= \frac{1}{2} \left( \omega^{m/2} \sum_{n=0}^{N-1} x(n) \omega^{mn} + \omega^{-m/2} \sum_{n=0}^{N-1} x(n) \omega^{-mn} \right) = \\ &= \frac{1}{2} \omega^{m/2} \sum_{k=0}^{2N-1} y(k) \omega^{mk} \end{aligned} \quad (2.1)$$

где  $\omega = \exp\left\{\frac{2\pi j}{2N}\right\}$ ,  $y(k)$  - вещественная  $2N$ -периодическая последовательность, полученная четным продолжением  $x(n)$ :

$$y(k) = \begin{cases} x(k) & \text{при } 0 \leq k \leq N-1 \\ x(2N-k-1) & \text{при } N \leq k \leq 2N-1 \end{cases} \quad (2.2)$$

Сложность такого алгоритма ДКП связана со сложностью ДПФ вещественной последовательности двойной длины соотношением:

$$M(N) = M_F^R(2N) + 3N, \quad A(N) = A_F^R(2N) + 3N. \quad (2.3)$$

### 2.2 Современный алгоритм ДКП четной длины

Пусть вещественная последовательность  $y(k)$  длины  $2N$  получена четным продолжением исходной последовательности  $x(n)$  по формуле (2.2), и ДКП связано с ДПФ соотношением (2.1).

Для ДПФ

$$Y(m) = \sum_{k=0}^{2N-1} y(k) \omega^{km} \quad (2.4)$$

длины  $2N$  проведем один шаг стандартной декомпозиции Кули-Тьюки:

$$Y(m) = \sum_{k=0}^{N-1} y(2k)\omega^{2km} + \omega^m \sum_{k=0}^{N-1} y(2k+1)\omega^{2km}. \quad (2.5)$$

Из (2.2) следует, что последовательности четных  $y(2k)$  и нечетных  $y(2k+1)$  отсчетов связаны между собой соотношением:

$$y(2k+1) = y(2N - (2k+1) - 1) = y(2(N - k - 1)) = y(2l)$$

при  $l = N - k - 1$ .

Пусть

$$\tilde{y}(m) = \sum_{k=0}^{N-1} y(2k)\omega^{2km},$$

тогда

$$\begin{aligned} \omega^m \sum_{k=0}^{N-1} y(2k+1)\omega^{2km} &= \omega^m \sum_{l=0}^{N-1} y(2l)\omega^{2(N-l-1)m} \\ &= \omega^{-m} \sum_{l=0}^{N-1} y(2l)\omega^{-2lm} = \overline{\omega^m \tilde{y}(m)} \end{aligned} \quad (2.6)$$

Из (2.5) и (2.6) следует:

$$Y(m) = \tilde{y}(m) + \overline{\omega^m \tilde{y}(m)}. \quad (2.7)$$

Тогда равенство (2.1) с учетом (2.5), (2.7) примет вид:

$$\begin{aligned} \hat{x}(m) &= \frac{1}{2} \omega^{m/2} \left( \tilde{y}(m) + \overline{\omega^m \tilde{y}(m)} \right) \\ &= \frac{1}{2} \left( \omega^{m/2} \tilde{y}(m) + \overline{\omega^{m/2} \tilde{y}(m)} \right) = \operatorname{Re} \left\{ \omega^{m/2} \tilde{y}(m) \right\}, \end{aligned} \quad (2.8)$$

где  $\tilde{y}(m)$  - ДПФ вещественного сигнала длины  $N$ ,  $0 \leq m \leq N-1$ .

Из соотношения (2.8) следует, что при выполнении комплексных умножений на  $\omega^{m/2}$  достаточно вычислять только действительную часть произведения, что потребует двух умножений и одного сложения на отсчет:

$$\operatorname{Re} \left\{ (a + ib)(\mu + i\nu) \right\} = a\mu - b\nu. \quad (2.9)$$

Таким образом, ДКП сведено к ДПФ вещественного сигнала той же длины,  $2N$  дополнительным умножениям и  $N$  сложениям. Обобщенная блок-схема изложенного алгоритма приведена на рис.2.1.

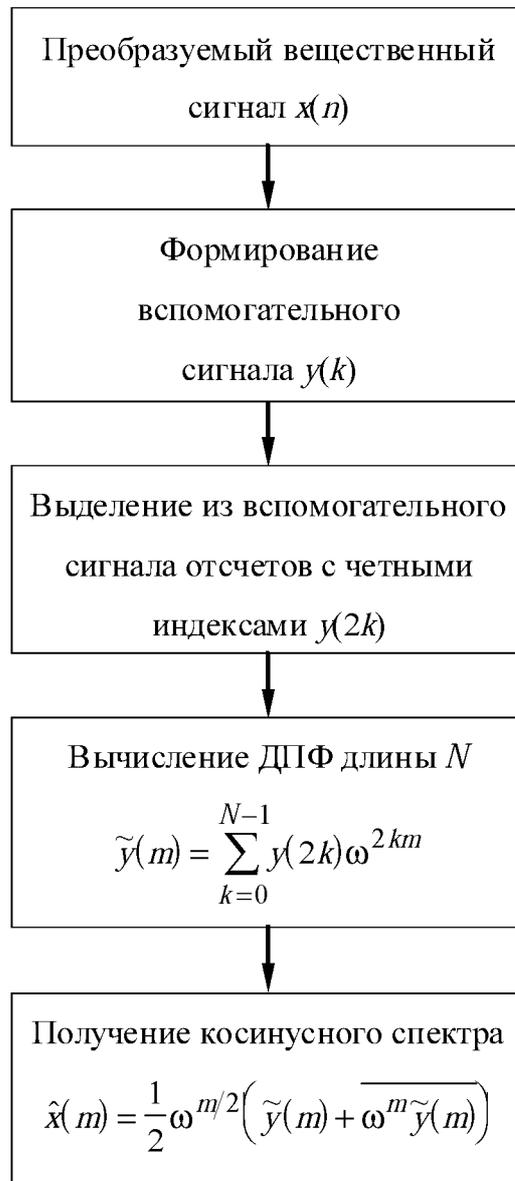


Рис.2.1. Обобщенная блок-схема алгоритма ДКП.

Мультипликативная и аддитивная сложности вычисления ДКП таким способом равны:

$$M(N) = M_F^R(N) + 2N, \quad A(N) = A_F^R(N) + N. \quad (2.10)$$

где  $M_F^R(N)$ ,  $A_F^R(N)$  - оценки сложности ДПФ вещественной последовательности той же длины  $N$ .

В таблице 2.1 приведены сложности традиционного (см. п. 2.1) и современного алгоритмов ДКП при разных вариантах реализации вспомогательного ДПФ, полученные на основании оценок (1.8), (1.11), (1.12), (2.3), (2.10). Оценки сложности вычисления ДПФ вещественной последовательности рассчитаны в предположении, что при этом используется “совмещенный” алгоритм [2].

В таблице 2.2 приведены численные данные о количестве операций, необходимых для вычисления ДКП при использовании вспомогательного алгоритма ДПФ с декомпозицией по основанию 2.

Таблица 2.1. Сложности алгоритмов

Способ вычисления вспомогательного ДПФ	Алгоритм п.2.1.	Алгоритм п.2.2.
По основанию 2	$M(N) \leq \frac{3N}{2} \log_2 N + \frac{3N}{2}$ $A(N) \leq \frac{7N}{2} \log_2 N + \frac{9N}{2}$	$M(N) \leq \frac{3N}{4} \log_2 N + \frac{N}{2}$ $A(N) \leq \frac{7N}{4} \log_2 N$
По основанию 4	$M(N) \leq \frac{9N}{8} \log_2 N + \frac{11N}{4}$ $A(N) \leq \frac{25N}{8} \log_2 N + \frac{23N}{4}$	$M(N) \leq \frac{9N}{16} \log_2 N + \frac{21N}{16}$ $A(N) \leq \frac{25N}{16} \log_2 N + \frac{13N}{16}$
Сплит-радикс	$M(N) \leq N \log_2 N + 3N$ $A(N) \leq 3N \log_2 N + 6N$	$M(N) \leq \frac{N}{2} \log_2 N + \frac{3N}{2}$ $A(N) \leq \frac{3N}{2} \log_2 N + N$

Таблица 2.2. Количество операций

N	Алгоритм п.2.1.		Алгоритм п.2.2.		Отношение суммарной сложности
	операций умножения	операций сложения	операций умножения	операций сложения	
8	63	135	31	51	2,41
16	147	323	71	127	2,37
32	339	755	163	307	2,33
64	771	1731	371	723	2,29
128	1731	3907	835	1667	2,25

### 2.3. Сведение ДКП нечетной длины к ДПФ вещественной последовательности той же длины

Следуя [6], представим ДКП нечетной длины  $N$  в виде

$$\hat{x}(m) = \sum_{n=0}^{N-1} x(n) \cos\left(\pi \frac{(n + 1/2)m}{N}\right) = \operatorname{Re} \left\{ \sum_{n=0}^{N-1} x(n) \omega^{(2n+1)m} \right\}, \quad (2.12)$$

где  $\omega = \exp\left\{\frac{2\pi i}{4N}\right\}$  - первообразный корень степени  $4N$  из единицы. Пусть

$$y(k) = \begin{cases} x(n) & \text{при } k = 2n + 1; \\ 0 & \text{при } k = 2n; \end{cases}$$

тогда соотношение (2.12) примет вид:

$$\hat{x}(m) = \operatorname{Re} \left\{ \sum_{n=0}^{2N-1} y(k) \omega^{km} \right\}. \quad (2.13)$$

При нечетном  $N$  числа 4 и  $N$  взаимно просты, декомпозиция Гуда-Томаса [2, 16, 26] по формулам (1.16), (1.17) при  $P=4$  и  $Q=N$  выполняется без дополнительных умножений.

Преобразование индексов (1.13), ограничения в (2.12) и (2.13) на диапазон изменения индексов  $m$  и  $k$ , а также обращение в нуль функции  $y(k)$  при четных  $k$  выделяют в двумерных массивах размера  $4 \times N$  “допустимые” подмножества  $K$  и  $M$  для пар  $(k_1, k_2)$  (аналог  $(n_1, n_2)$  в (1.13) - (1.15)) и  $(m_1, m_2)$ . Кроме того, при  $P=4$  корень  $\alpha$  в (1.13) равен мнимой единице  $i$ . Тогда из (2.13) по аналогии с (1.14) получается:

$$Y(m_1, m_2) = \sum_{k_2=0}^3 \left( \sum_{k_1=0}^{N-1} \tilde{y}(k_1, k_2) \beta^{k_1 m_1} \right) i^{k_2 m_2}. \quad (2.14)$$

Так как “допустимое” подмножество индексов  $K$  сформировано так, что  $\tilde{y}(k_1, k_2)$  отлично от нуля только при  $(k_1, k_2) \in K$ , то и суммирование в (2.14) выполняется только при  $(k_1, k_2) \in K$ , то есть при  $k_2 = 1, 3$ . Этот факт позволяет привести выражение (2.14) к виду:

$$\hat{x}(m) = \operatorname{Re} \left\{ i^{m_2} \sum_{k_1=0}^{N-1} z(k_1) \beta^{k_1 m_1} \right\}, \quad (2.15)$$

где

$$z(k_1) = \begin{cases} \tilde{y}(k_1, 1) & \text{при } (k_1, 1) \in K \\ \tilde{y}(N - k_1, 3) & \text{при } (k_1, 3) \in K \end{cases}, \quad \beta = \exp \left\{ \frac{2\pi i}{N} \right\}.$$

Таким образом, ДКП нечетной длины  $N$  сведено к вещественному преобразованию Фурье той же длины.

#### 2.4. Пример. Алгоритм дискретного косинусного преобразования длиной $N=3^k$

Следуя [7], представим ДКП нечетной длины  $N$  в виде

$$\hat{x}(m) = \sum_{n=0}^{N-1} x(n) \cos \left( \pi \frac{(n + 1/2)m}{N} \right) = \operatorname{Re} \left\{ \sum_{n=0}^{N-1} x(n) \omega^{(2n+1)m} \right\}, \quad (2.16)$$

где  $\omega = \exp \left\{ \frac{2\pi i}{4N} \right\}$  - первообразный корень степени  $4N$  из единицы. Пусть

$$g(k) = \begin{cases} x(n) & \text{при } k = 2n + 1; \\ 0 & \text{при } k = 2n; \end{cases}$$

тогда соотношение (2.16) примет вид:

$$\hat{x}(m) = \operatorname{Re} \left\{ \sum_{n=0}^{2N-1} g(k) \omega^{km} \right\}. \quad (2.17)$$

При нечетном  $N$  числа 4 и  $N$  взаимно просты, декомпозиция Гуда-Томаса (см. п.1.3.4) по формулам (1.16), (1.17) при  $P=4$  и  $Q=N$  выполняется без дополнительных умножений.

Преобразование индексов (1.13), ограничения в (2.16) и (2.17) на диапазон изменения индексов  $m$  и  $k$ , а также обращение в нуль функции  $g(k)$  при четных  $k$  выделяют в двумерных массивах размера  $4 \times N$  “допустимые” подмножества  $K$  и  $M$  для пар  $(k_1, k_2)$  (аналог  $(n_1, n_2)$  в (1.13) - (1.15)) и  $(m_1, m_2)$  (см. рис 2.2). Кроме

того, при  $P=4$  корень  $\alpha$  в (1.15) равен мнимой единице  $i$ . Тогда из (2.17) по аналогии с (1.18) получается:

$$G(m_1, m_2) = \sum_{k_2=0}^3 \left( \sum_{k_1=0}^{N-1} \tilde{g}(k_1, k_2) \beta^{k_1 m_1} \right) i^{k_2 m_2}. \quad (2.18)$$

$\omega$	$k_2$	$K$								
3		27	31	35	<b>3</b>	<b>7</b>	<b>11</b>	<b>15</b>	19	23
2		18	22	26	30	34	2	6	10	14
1		<b>9</b>	<b>13</b>	<b>17</b>	21	25	29	33	<b>1</b>	<b>5</b>
0		0	4	8	12	16	20	24	28	32
		0	1	2	3	4	5	6	7	8
		$k_1$								

	$m_2$	$M$								
3		27	19	11	<b>3</b>	31	23	15	<b>7</b>	35
2		18	10	<b>2</b>	30	22	14	<b>6</b>	34	26
1		9	<b>1</b>	29	21	13	<b>5</b>	33	25	17
0		<b>0</b>	28	20	12	<b>4</b>	32	24	16	<b>8</b>
		0	1	2	3	4	5	6	7	8
		$m_1$								

Рис. 2.2. Допустимые области значений пар  $(k_1, k_2)$  и  $(m_1, m_2)$  при  $N=9$

Так как “допустимое” подмножество индексов  $K$  сформировано так, что  $\tilde{g}(k_1, k_2)$  отлично от нуля только при  $(k_1, k_2) \in K$ , то и суммирование в (2.18) выполняется только при  $(k_1, k_2) \in K$ , то есть при  $k_2 = 1, 3$ . Этот факт позволяет привести выражение (2.18) к виду:

$$\hat{x}(m) = \text{Re} \left\{ i^{m_2} \sum_{k_1=0}^{N-1} z(k_1) \beta^{k_1 m_1} \right\}, \quad (2.19)$$

где

$$z(k_1) = \begin{cases} \tilde{g}(k_1, 1) & \text{при } (k_1, 1) \in K \\ \tilde{g}(N - k_1, 3) & \text{при } (k_1, 3) \in K \end{cases}, \quad \beta = \exp \left\{ \frac{2\pi i}{N} \right\}.$$

Таким образом, ДКП нечетной длины  $N$  сведено к вещественному преобразованию Фурье той же длины. При  $N=3^r$  и использовании ДПФ, описанного в п.1.3.6, оценки вычислительной сложности такого алгоритма ДКП имеют вид [7]:

$$M(N) = N \log_3 N - N, \quad A(N) = 3N \log_3 N - \frac{2}{3} N. \quad (2.20)$$

### 3. АЛГОРИТМЫ ДКП КОРОТКИХ ДЛИН

#### 3.1. Алгебраические принципы синтеза алгоритмов ДКП коротких длин

##### 3.1.1. Дискретное косинусное преобразование длины 8

В ненормализованной матричной форме ДКП (1.1) длины  $N=8$  принимает вид  $\mathbf{X} = \mathbf{F}\mathbf{x}$ , где  $\mathbf{X}, \mathbf{x}$  - восьмимерные векторы-столбцы выходного и входного сигналов, соответственно;

$$\mathbf{X}^t = (X(0), \dots, X(7)), \quad \mathbf{x}^t = (x(0), \dots, x(7)),$$

$\mathbf{F}$  - матрица ДКП,  $t$  - знак транспонирования.

После переупорядочивания компонент входного и выходного векторов

$$\begin{aligned} \mathbf{Y}^t &= (Y(0), \dots, Y(7)) \\ &= (X(1), X(5), X(7), X(3), X(2), X(6), X(4), X(0)) \end{aligned} \quad (3.1),$$

$$\begin{aligned} \mathbf{y}^t &= (y(0), \dots, y(7)) \\ &= (x(0), x(2), x(4), x(6), x(7), x(5), x(3), x(1)) \end{aligned} \quad (3.2)$$

матричное представление ДКП может быть записано в форме  $\mathbf{Y} = \mathbf{T}_8\mathbf{y}$ , где

$$\mathbf{T}_8 = \left( \begin{array}{cc|cc|cc|cc} a & c & -d & -b & -a & -c & d & b \\ c & d & -b & a & -c & -d & b & -a \\ d & b & a & c & -d & -b & -a & -c \\ b & -a & c & d & -b & a & -c & -d \\ \hline f & -l & -f & l & f & -l & -f & l \\ l & f & -l & -f & l & f & -l & -f \\ \hline g & -g & g & -g & g & -g & g & -g \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right), \quad (3.3)$$

где

$$a = \cos\left(\frac{\pi}{16}\right), \quad b = \cos\left(\frac{3\pi}{16}\right), \quad c = \cos\left(\frac{5\pi}{16}\right), \quad d = \cos\left(\frac{7\pi}{16}\right),$$

$$l = \cos\left(\frac{6\pi}{16}\right), \quad f = \cos\left(\frac{2\pi}{16}\right), \quad g = \cos\left(\frac{4\pi}{16}\right).$$

Формирование из компонент вектора  $y$  вспомогательного массива:

$$\begin{aligned}
 z(0) &= y(0) - y(4), \quad z(1) = y(1) - y(5), \\
 z(2) &= y(2) - y(6), \quad z(3) = y(3) - y(7), \\
 z(4) &= (y(0) + y(4)) - (y(2) + y(6)), \\
 z(5) &= (y(1) + y(5)) - (y(3) + y(7)), \\
 z(6) &= [(y(0) + y(4)) + (y(2) + y(6))] - [(y(1) + y(5)) + (y(3) + y(7))], \\
 z(7) &= [(y(0) + y(4)) + (y(2) + y(6))] + [(y(1) + y(5)) + (y(3) + y(7))].
 \end{aligned} \tag{3.4}$$

требует 14 операций вещественного сложения. После этого выполнение косинусного преобразования сводится к следующим матричным вычислениям:

$$\begin{pmatrix} Y(0) \\ Y(1) \\ Y(2) \\ Y(3) \end{pmatrix} = \begin{pmatrix} a & c & -d & -b \\ c & d & -b & a \\ d & b & a & c \\ b & -a & c & d \end{pmatrix} \begin{pmatrix} z(0) \\ z(1) \\ z(2) \\ z(3) \end{pmatrix}, \tag{3.5}$$

$$\begin{pmatrix} Y(4) \\ Y(5) \end{pmatrix} = \begin{pmatrix} f & -1 \\ 1 & f \end{pmatrix} \begin{pmatrix} z(4) \\ z(5) \end{pmatrix}, \tag{3.6}$$

$$\begin{aligned}
 Y(6) &= g z(6), \\
 Y(7) &= z(7).
 \end{aligned} \tag{3.7}$$

**Лемма 3.1.** а) Вычисление матричного произведения (3.5) эквивалентно вычислению произведения элементов  $\mathbf{s}, \mathbf{p} \in \Lambda_1^{(4)}$ :

$$\mathbf{sp} = (c + a\mathbf{e}_1 + b\mathbf{e}_2 + d\mathbf{e}_3)(z(0) + z(1)\mathbf{e}_1 + z(2)\mathbf{e}_2 + z(3)\mathbf{e}_3)$$

и, в соответствии с утверждением (з) Леммы 8.1, требует 9 операций вещественного умножения и 15 операций вещественного сложения.

б) Вычисление матричного произведения (3.6) эквивалентно вычислению произведения элементов  $\mathbf{q}, \mathbf{r} \in \Lambda_1^{(2)}$ :

$$\mathbf{qr} = (f + 1\mathbf{e}_1)(z(4) + z(5)\mathbf{e}_1)$$

и, в соответствии с утверждением (а) Леммы 8.1 требует 3 операций вещественного умножения и 3 операций вещественного сложения.

в) Вычисление по формуле (3.7) требует одной операции вещественного умножения.

Суммарная сложность алгоритма ДКП длины 8 с учетом формирования вспомогательных переменных  $z(0), \dots, z(7)$  составляет  $9+3+1=13$  операций умножения и  $14+15+3=32$  операции сложения.

**Замечание 3.1.** Структура рассмотренного алгоритма не зависит от конкретных значений параметров  $a, b, \dots, g$ . Пусть

$$g' = f' = c' = 1, \quad f' = f/1, \quad a' = a/c, \quad d' = d/c, \quad b' = b/c.$$

Тогда умножение в (3.7) становится тривиальным, в матричном произведении (3.6) остается два умножения. Вычисление правой части соотношения (3.5) требует 8 операций умножения. А умножения на  $g, e, c$  объединяются с нормализацией компонент косинусного спектра (с умножениями на коэффициент  $\lambda_m$ ) в (1.1). Таким образом, предложенный алгоритм ДКП длины 8 требует  $2+8=10$  операций умножения и 32 операции сложения.

**Замечание 3.2.** Анализ утверждений Леммы 8.3 позволяет утверждать, что алгоритм ДКП длины  $N = 8$  может быть реализован как "макрооперация" умножения элементов восьмимерной алгебры

$$\mathbb{B}_8 \cong \mathcal{A}_1^{(4)} \oplus \mathbb{C} \oplus \mathbb{R} \oplus \mathbb{R}.$$

Таким образом, принципиальная схема алгоритма представляет собой следующую последовательность действий

$\mathbb{R}^8$	$\longrightarrow$	$\mathbb{B}_8 \cong \mathcal{A}_1^{(4)} \oplus \mathbb{C} \oplus \mathbb{R} \oplus \mathbb{R}$	$\longrightarrow$	$\mathbb{R}^8$
Входные данные	Действие линейного оператора	Умножение элементов алгебры	Действие линейного оператора (проекция)	Выходные данные (спектр)

Подробная схема алгоритма (направленный граф) изображена на рис.3.1.

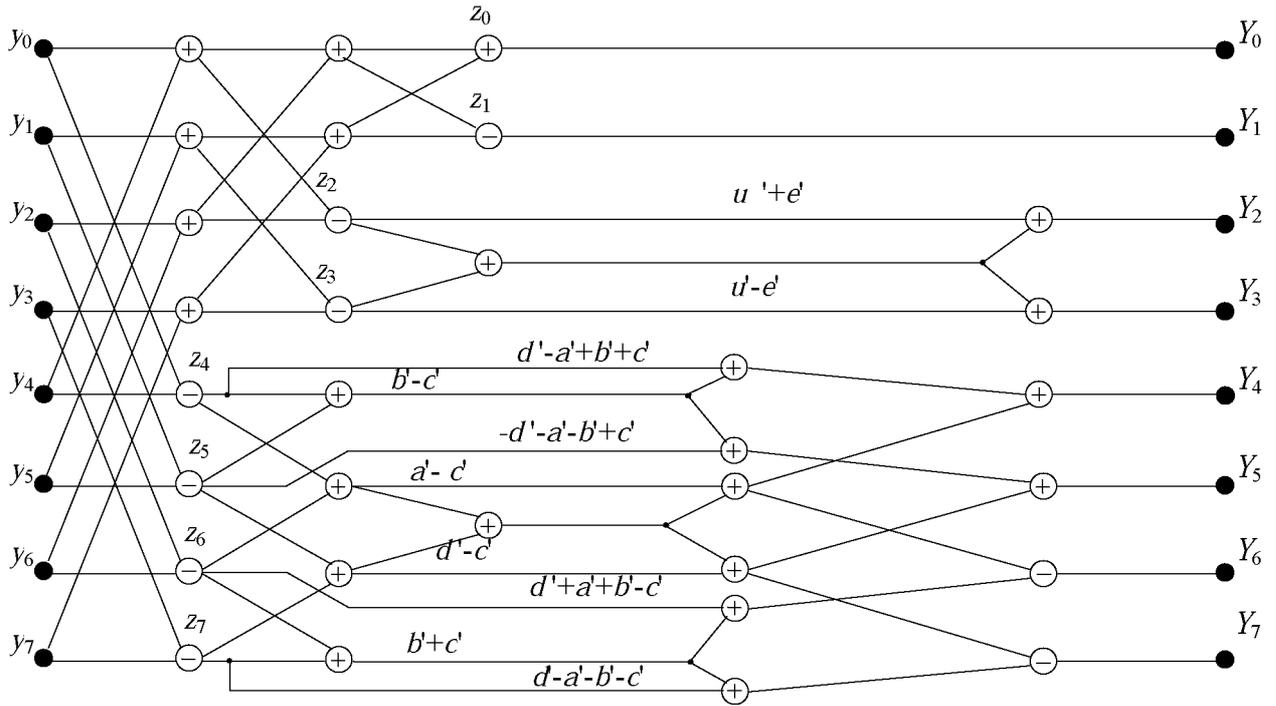


Рис. 3.1. Схема (направленный граф) алгоритма ДКП длины  $N = 8$

### 3.1.2. Связь между алгебраическими свойствами значений базисных функций ДКП и структурой матрицы преобразования

Нетрудно заметить, что матрица  $T_8$  представима в факторизованном виде

$$T_8 = A \cdot B \cdot C,$$

где  $A = E_8$  - единичная матрица восьмого порядка,

$$B = \begin{pmatrix} a & c & -d & -b & & & & \\ c & d & -b & a & & & & \\ d & b & a & c & & & & \\ b & -a & c & d & & & & \\ & & & & f & -1 & & \\ & & & & 1 & f & & \\ & & & & & & g & \\ & & & & & & & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & & & & -1 & & & \\ & 1 & & & & & -1 & \\ & & 1 & & & & & -1 \\ & & & 1 & & & & -1 \\ 1 & -1 & & 1 & & 1 & & -1 \\ & 1 & & -1 & & 1 & & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

(нулевые элементы матриц  $B, C$  не указаны).

Такое факторизованное представление матриц преобразований характерно для многих алгоритмов дискретных ортогональных преобразований:  $A$  - "матрица постсложений",  $C$  - "матрица предсложений",  $B$  - собственно матрица

преобразования, которая может быть факторизована далее и приведена к форме, обеспечивающей реализацию "быстрых" алгоритмов умножения матриц специфического вида. Вне рассмотрения в этом случае остается арифметическая природа зависимости значений базисных функций преобразования, и возможности такой факторизации. Проанализируем эту зависимость.

Как уже отмечалось, формально базисные функции ДКП (1.1)

$$\cos\left(\frac{2\pi(2n+1)m}{4N}\right) = \frac{1}{2}\left(\exp\left(\frac{2\pi i(2n+1)m}{4N}\right) + \exp\left(-\frac{2\pi i(2n+1)m}{4N}\right)\right)$$

выражаются через комплексные корни степени  $4N$  из единицы, которые, в свою очередь, принадлежат круговому полю  $\mathbb{Q}(\omega_{4N})$ , где

$$\omega_{4N} = \exp\left(\frac{2\pi i}{4N}\right).$$

Круговое поле  $\mathbb{Q}(\omega_M)$ , как поле алгебраических чисел имеет степень  $\varphi(M)$  над  $\mathbb{Q}$ , где  $\varphi(M)$  - функция Эйлера:

$$\varphi(M) = \begin{cases} p-1, & \text{если } M = p, \text{ где } p - \text{ простое число;} \\ p^{\alpha-1}(p-1), & \text{если } M = p^\alpha, \text{ где } p - \text{ простое число;} \\ \varphi(M_1)\varphi(M_2), & \text{если } M = M_1M_2, \text{ где н.о.д.}(M_1M_2) = 1, \end{cases}$$

то есть имеет, в рассматриваемом случае  $N = 8$ , степень 16 над  $\mathbb{Q}$ . Но значения базисных функций

$$h_m(n) = \cos\left(\frac{2\pi(2n+1)m}{4N}\right)$$

принадлежат подполю  $\mathbb{Q}(\omega_{4N} + \omega_{4N}^{-1})$ , этого поля, имеющего, как поле алгебраических чисел степень в два раза меньшую над  $\mathbb{Q}$ :

$$\left[\mathbb{Q}(\omega_{4N} + \omega_{4N}^{-1}) : \mathbb{Q}\right] = \frac{1}{2}\varphi(4N) = 8$$

Принадлежность значений  $h_m(n)$  при конкретных  $m, n$  подполям меньшей степени алгебраичности, в силу нечетности числа  $(2n+1)$ , определяется



## 3.2. Примеры алгоритмов ДКП с минимальной арифметической сложностью

### 3.2.1. Алгоритм ДКП длины 9

Аналогичная, но более сложная картина распределения значений базисных функций имеет место и для других длин преобразования (1.1). Это усложнение связано с тем, что в отличие от случая  $N=8$ , число  $(2n+1)$  в выражении базисных функций

$$h_m(n) = \cos\left(\frac{2\pi(2n+1)m}{4N}\right)$$

может иметь общий множитель с числом  $N$  – длиной преобразования, что влечет более сложное распределение степени алгебраичности значений базисных функций преобразования не только в столбцах матрицы преобразования, но и в ее строках.

Действительно, пусть  $N=9$ . Как и ранее, формально базисные функции ДКП (1.1) выражаются через комплексные корни степени  $4N$ , то есть, в рассматриваемом случае через корни степени 36 из единицы, которые, в свою очередь, принадлежат круговому полю  $\mathbb{Q}(\omega_{4N}) = \mathbb{Q}(\omega_{36})$  и

$$[\mathbb{Q}(\omega_{36}) : \mathbb{Q}] = 12.$$

В действительности, значения базисных функций, именно, соответствующих значений косинусной функции, имеют степень алгебраичности, как минимум, в два раза меньшую, то есть принадлежат полю алгебраических чисел  $\mathbb{Q}(\omega_{36} + \omega_{36}^{-1})$  и

$$[\mathbb{Q}(\omega_{36} + \omega_{36}^{-1}) : \mathbb{Q}] = \frac{1}{2}\varphi(36) = 6.$$

Кроме того, при конкретных значениях  $m, n$  числитель дроби

$$\frac{m(2n+1)}{4N} = \frac{m(2n+1)}{4 \cdot 9}$$

может иметь общий числитель со знаменателем, что еще более понижает алгебраичность соответствующих значений базисных функций. В таблице 3.3 приведены данные о распределении степени алгебраичности этих функций.

$m \setminus n$	0	1	2	3	4	5	6	7	8
0	1	1	1	1	1	1	1	1	1
1	6	2	6	6	0	6	6	2	6
2	3	1	3	3	0	3	3	1	3
3	2	0	2	2	0	2	2	0	2
4	3	1	3	3	1	3	3	1	3
5	6	2	6	6	0	6	6	2	6
6	2	1	2	2	1	2	2	1	2
7	6	2	6	6	0	6	6	2	6
8	3	1	3	3	1	3	3	1	3

После перестановки ряда строк и столбцов матрица ДКП длины 9 принимает "блочный" вид, определяемый, как и в случае  $N=8$ , распределением степени алгебраичности базисных функций преобразования.

$$T_9 = \left( \begin{array}{ccc|ccc|cc|c} a & c & d & -d & -c & -a & b & -b & 0 \\ c & -d & a & -a & d & -c & -b & b & 0 \\ d & a & -c & c & -a & -d & -b & b & 0 \\ \hline e & -f & -g & -g & -f & e & h & h & -1 \\ g & -e & f & f & -e & g & -h & -h & 1 \\ f & g & -e & -e & g & f & -h & -h & 1 \\ \hline b & -b & -b & b & b & -b & 0 & 0 & 0 \\ \hline h & h & h & h & h & h & -1 & -1 & -1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right),$$

**Лемма 3.2.** Матричные умножения

$$\begin{pmatrix} a & c & d \\ c & -d & a \\ d & a & -c \end{pmatrix} \begin{pmatrix} z(0) \\ z(1) \\ z(2) \end{pmatrix}, \text{ и } \begin{pmatrix} e & -f & -g \\ g & -e & f \\ f & g & -e \end{pmatrix} \begin{pmatrix} z(5) \\ z(6) \\ z(7) \end{pmatrix},$$

эквивалентны умножению элементов алгебры  $\mathcal{A}^{(3)}$  :

$$(-ae_2 - ce_1 + d)(z(0)e_1 + z(1)e_2 + z(2)) \text{ и}$$

$$(fe_2 - ge_1 - e)(z(6)e_2 - z(5)e_1 + z(7))$$

соответственно и требуют, согласно утверждению (в) Леммы 8.1, по 4 вещественных умножения и 14 вещественных сложений каждое.

**Следствие 3.1.** ДКП длины 9, реализуемое посредством умножения на матрицу  $T_9$  выполняется за 8 умножений и 44 сложения (то есть, требует менее одного умножения и около пяти сложений на отсчет).

**Замечание 3.3.** Анализ утверждений Леммы 8.4 позволяет утверждать, что алгоритм ДКП длины  $N = 9$  может быть реализован как "макрооперация" умножения элементов девятимерной алгебры

$$\mathcal{B}_9 \cong \mathcal{A}^{(3)} \oplus \mathcal{A}^{(3)} \oplus \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R}.$$

Схема алгоритма (направленный граф) изображена на рис.3.2.

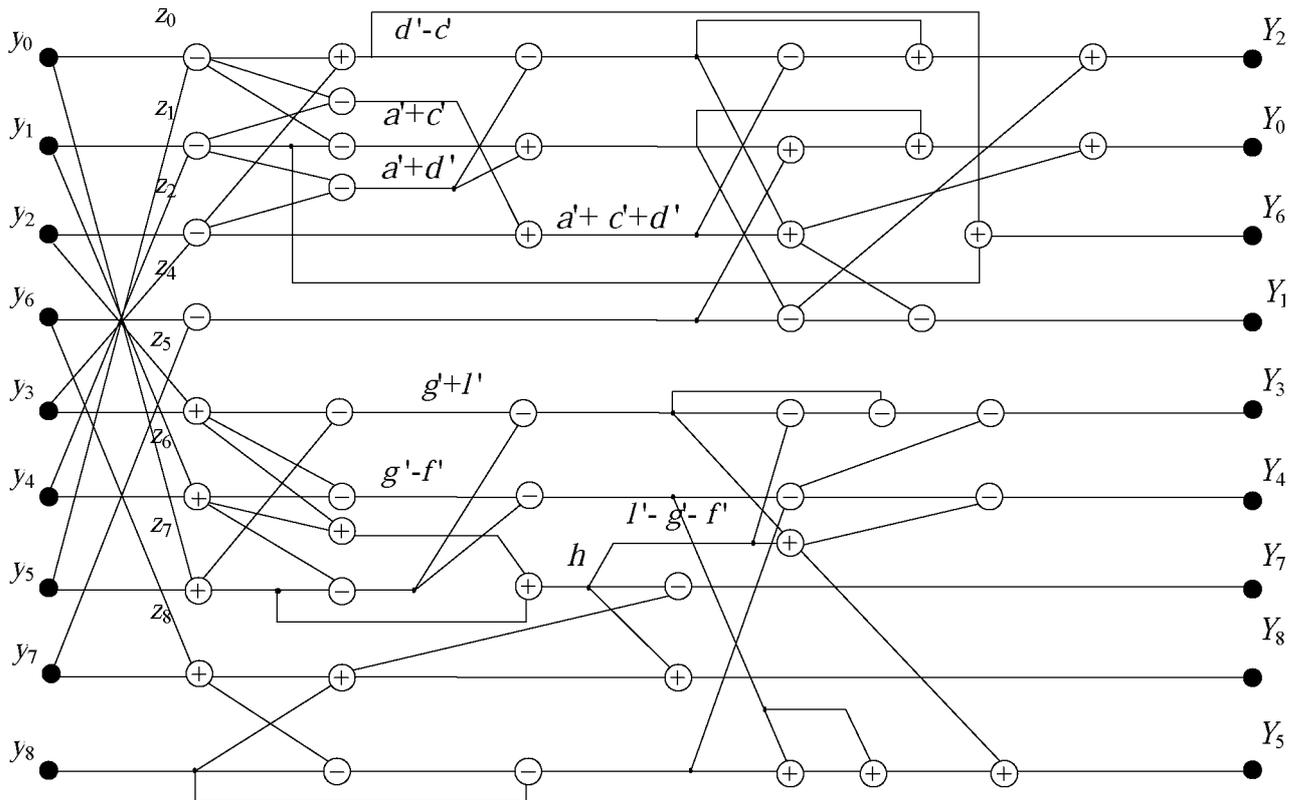


Рис. 3.2. Схема (направленный граф) алгоритма ДКП длины  $N = 9$

### 3.2.2. Алгоритм ДКП длины 10

С точностью до перестановки ряда строк и столбцов и переобозначений элементов (значений базисных функций преобразования) матрица преобразования имеет вид:

$$\mathbf{T}_{10} = \left( \begin{array}{cccc|cccc|cc} a & b & c & d & -d & -c & -b & -a & q & -q \\ b & d & -a & -c & c & a & -d & -b & -q & q \\ c & -a & d & -b & b & -d & a & -c & q & -q \\ d & -c & -b & a & -a & b & c & -d & q & -q \\ \hline 1 & f & -f & -1 & -1 & -f & f & 1 & 0 & 0 \\ f & -1 & 1 & -f & -f & 1 & -1 & f & 0 & 0 \\ \hline g & -h & -h & g & g & -h & -h & g & -1 & -1 \\ h & -g & -g & h & h & -g & -g & h & 1 & 1 \\ \hline q & -q & q & q & -q & -q & q & -q & -q & q \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right).$$

**Лемма 3.3.** а) Матричное умножение

$$\begin{pmatrix} a & b & c & d \\ b & d & -a & -c \\ c & -a & d & -b \\ d & -c & -b & a \end{pmatrix} \begin{pmatrix} z(0) \\ z(1) \\ z(2) \\ z(3) \end{pmatrix}$$

эквивалентно вычислению произведения элементов алгебры  $\mathcal{A}_2^{(4)}$

$$(be_1 + ae_2 - ce_3 + d)(z(0)e_1 + z(1)e_2 + z(4)e_3 - z(3)).$$

и требует согласно утверждению (д) Леммы 8.1 пяти вещественных умножений и 15 вещественных сложений.

б) Вычисление матричного произведения

$$\begin{pmatrix} 1 & f \\ f & -1 \end{pmatrix} \begin{pmatrix} z(4) \\ z(5) \end{pmatrix}$$

эквивалентно вычислению произведения элементов алгебры  $\mathcal{A}_1^{(2)}$  :

$$(1 + fe_1)(z(4) - z(5)e_1)$$

и, в соответствии с утверждением (а) Леммы 1.1, требует 3 операций вещественного умножения и 3 операций вещественного сложения.

в) Вычисление матричного произведения

$$\begin{pmatrix} g & -h \\ h & -g \end{pmatrix} \begin{pmatrix} z(6) \\ z(7) \end{pmatrix}$$

эквивалентно вычислению произведения элементов алгебры  $\mathcal{A}_2^{(2)}$  :

$$(g + he_1)(z(6) - z(7)e_1)$$

и, в соответствии с Леммой 1.1(б) требует 2 операций вещественного умножения и 4 операций вещественного сложения.

**Следствие 3.2.** ДКП длины 10 посредством умножения на матрицу  $\mathbf{T}_{10}$  выполняется за 9 умножений и 43 сложения и эквивалентно умножению элементов в алгебре

$$\mathbb{B}_{10} \cong \mathcal{A}_2^{(4)} \oplus \mathbb{C} \oplus (\mathbb{R} \oplus \mathbb{R}) \oplus \mathbb{R} \oplus \mathbb{R}$$

Схема алгоритма (направленный граф) изображена на рис.3.3.

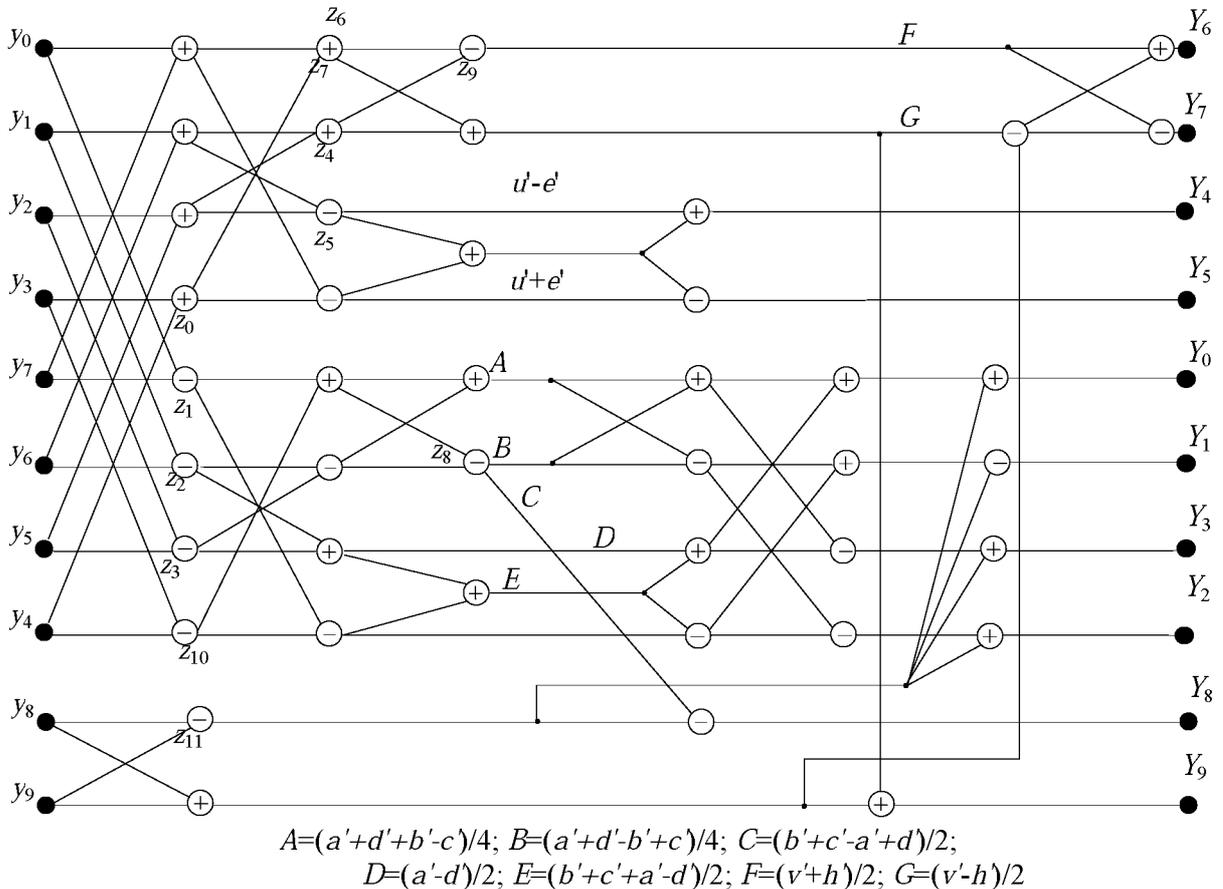


Рис. 3.3. Схема (направленный граф) алгоритма ДКТ длины  $N = 10$

### 3.2.3. Алгоритм ДКП длины 12

Аналогичные утверждения справедливы для  $N=12$ .

С точностью до перестановки ряда строк и столбцов матрица преобразования имеет вид:

$$T_{12} = \left( \begin{array}{cccc|cccc|cc|cc} a & c & -f & -d & d & f & -c & -a & b & e & -e & -b \\ c & -a & -d & f & -f & d & a & -c & -e & b & -b & e \\ f & d & a & c & -c & -a & -d & -f & -e & b & -b & e \\ d & -f & c & -a & a & -c & f & -d & -b & -e & e & b \\ \hline g & l & -g & -l & -l & -g & l & g & h & -h & -h & h \\ l & g & -l & -g & -g & -l & g & l & -h & h & h & -h \\ \hline b & -e & e & b & -b & -e & e & -b & e & -b & b & -e \\ e & b & -b & e & -e & b & -b & -e & -b & -e & e & b \\ \hline p & -p & p & -p & -p & p & -p & p & 0 & 0 & 0 & 0 \\ \hline q & q & q & q & q & q & q & q & -1 & -1 & -1 & -1 \\ \hline h & -h & -h & h & h & -h & -h & h & -h & h & h & -h \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right),$$

где

$$a = \cos\left(\frac{\pi}{24}\right), \quad c = \cos\left(\frac{5\pi}{24}\right), \quad d = \cos\left(\frac{7\pi}{24}\right), \quad f = \cos\left(\frac{11\pi}{24}\right),$$

$$g = \cos\left(\frac{\pi}{12}\right), \quad l = \cos\left(\frac{5\pi}{12}\right), \quad e = \cos\left(\frac{3\pi}{8}\right), \quad b = \cos\left(\frac{\pi}{8}\right),$$

$$h = \cos\left(\frac{\pi}{4}\right), \quad p = \cos\left(\frac{\pi}{6}\right), \quad q = \cos\left(\frac{\pi}{3}\right).$$

**Лемма 3.4.** Умножение матрицы ДКП длины 12 на входной вектор эквивалентно

а) умножению переменного элемента алгебры  $A_3^{(4)}$  на постоянный элемент этой же алгебры;

б) умножению переменного элемента алгебры  $A_1^{(2)}$  на постоянный элемент этой же алгебры;

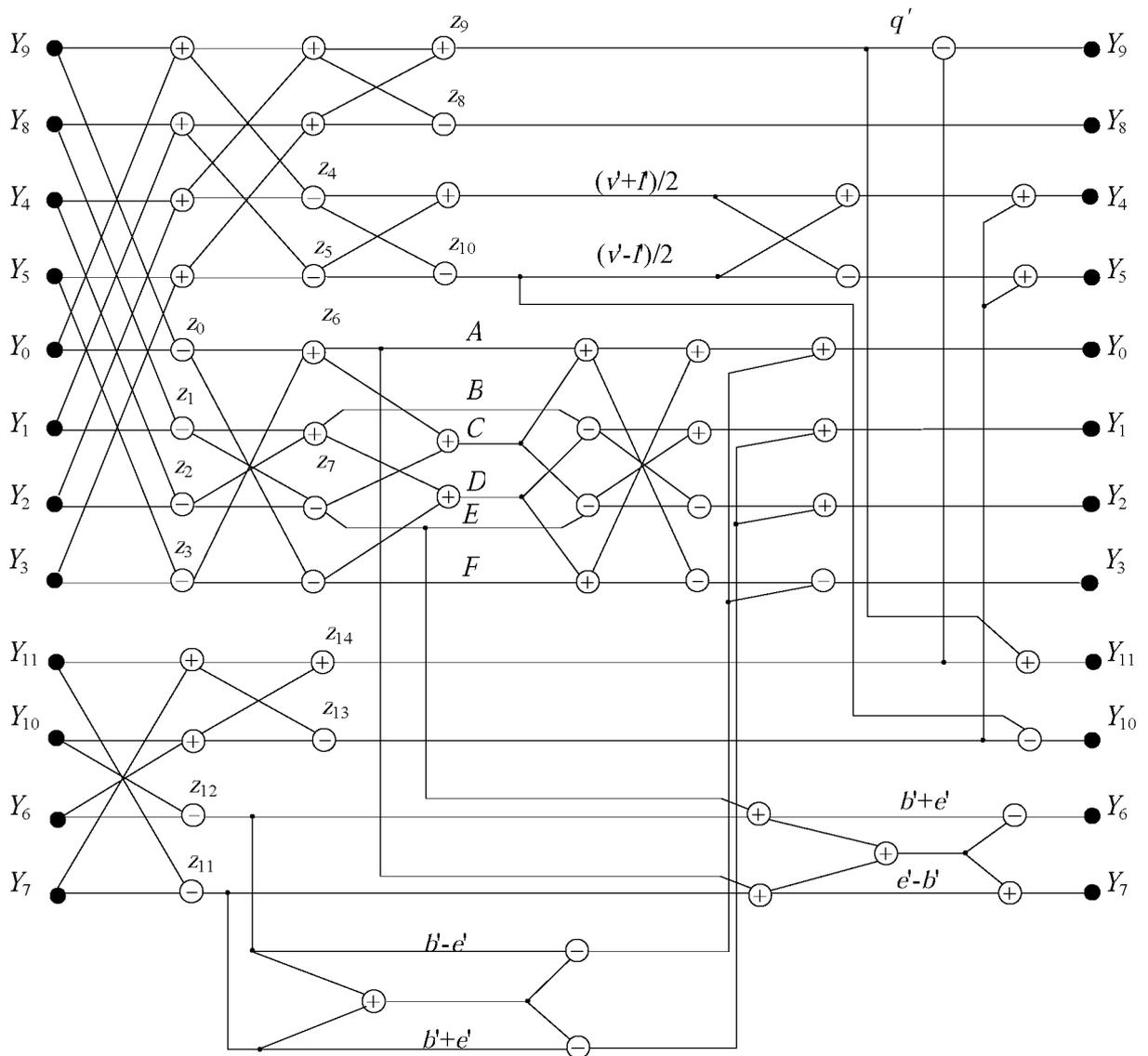
в) умножению переменного элемента алгебры  $A_2^{(2)}$  на постоянный элемент этой же алгебры;

г) дополнительным умножениям констант на переменные и вспомогательным сложениям.

**Следствие 3.3.** ДКП длины 12, реализуемое посредством умножения на матрицу  $T_{12}$  выполняется за 13 умножений и 55 сложений и эквивалентно умножению элементов в алгебре

$$\mathbb{B}_{12} \cong \mathcal{A}_3^{(4)} \oplus \mathbb{C} \oplus (\mathbb{R} \oplus \mathbb{R}) \oplus \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R}.$$

Схема алгоритма (направленный граф) изображена на рис.3.4.



$$A=(a'-d'-c'-u')/2; B=(a'+d+c'-u')/2; C=(c'+u')/2;$$

$$D=(c'-u')/2; E=(a'-d'+c'+u')/2; F=(a'+d'-c'+u')/2$$

Рис 3.4. Схема (направленный граф) алгоритма ДКП длины  $N = 12$

### 3.2.4. Алгоритм ДКП длины 15

С точностью до перестановки строк и столбцов матрица преобразования имеет

вид:

$$T_{15} = \left( \begin{array}{cccc|cccc|cc|cc|cc} a & d & f & g & -g & -f & -d & -a & b & e & -e & -b & c & -c & 0 \\ d & f & -g & -a & a & g & -f & -d & -e & b & -b & e & -c & c & 0 \\ f & -g & a & -d & d & -a & g & -f & -b & -e & e & b & c & -c & 0 \\ g & -a & -d & f & -f & d & s & -g & -e & b & -b & e & c & -c & 0 \\ \hline h & p & -s & -q & -q & -s & p & h & l & -r & -r & l & u & u & -1 \\ q & -h & -p & s & s & -p & -h & q & r & -l & -l & r & -u & -u & 1 \\ s & q & -h & -p & -p & -h & q & s & -l & r & r & -l & -u & -u & 1 \\ p & -s & -q & h & h & -q & -s & p & -r & l & l & -r & u & u & -1 \\ \hline b & -e & -b & -e & e & b & e & -b & e & -b & b & -e & 0 & 0 & 0 \\ e & b & -e & b & -b & e & -b & -e & -b & -e & e & b & 0 & 0 & 0 \\ \hline l & -r & l & -r & -r & l & -r & l & -r & l & l & -r & -1 & -1 & -1 \\ r & -l & r & -l & -l & r & -l & r & -l & r & r & -l & 1 & 1 & 1 \\ \hline c & -c & c & c & -c & -c & c & -c & 0 & 0 & 0 & 0 & -c & c & 0 \\ \hline u & u & u & u & u & u & u & u & -1 & -1 & -1 & -1 & u & u & -1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right),$$

где

$$a = \cos\left(\frac{\pi}{30}\right), \quad d = \cos\left(\frac{7\pi}{30}\right), \quad f = \cos\left(\frac{11\pi}{30}\right), \quad g = \cos\left(\frac{13\pi}{30}\right),$$

$$h = \cos\left(\frac{\pi}{15}\right), \quad p = \cos\left(\frac{7\pi}{15}\right), \quad s = \cos\left(\frac{4\pi}{15}\right), \quad q = \cos\left(\frac{2\pi}{15}\right),$$

$$b = \cos\left(\frac{\pi}{10}\right), \quad e = \cos\left(\frac{3\pi}{10}\right), \quad l = \cos\left(\frac{\pi}{5}\right), \quad r = \cos\left(\frac{2\pi}{5}\right),$$

$$c = \cos\left(\frac{\pi}{6}\right), \quad u = \cos\left(\frac{\pi}{3}\right).$$

**Лемма 3.5.** Умножение матрицы ДКП длины 15 на входной вектор эквивалентно двум умножениям переменного элемента алгебры  $\mathcal{A}_2^{(4)}$  на постоянный элемент этой же алгебры; умножению переменного элемента алгебры  $\mathcal{A}_1^{(2)}$  на постоянный элемент этой же алгебры; умножению переменного элемента алгебры

$\mathcal{A}_2^{(2)}$  на постоянный элемент этой же алгебры; дополнительным умножениям констант на переменные и вспомогательным сложениям  $u$ , следовательно, выполняется за 24 умножения и 83 сложения и эквивалентно умножению элементов в алгебре

$$\mathbb{B}_{15} \cong \mathcal{A}_2^{(4)} \oplus \mathcal{A}_2^{(4)} \oplus \mathbb{C} \oplus (\mathbb{R} \oplus \mathbb{R}) \oplus \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R}.$$

Схема алгоритма ДКП длины 15 изображена на рис.3.5.

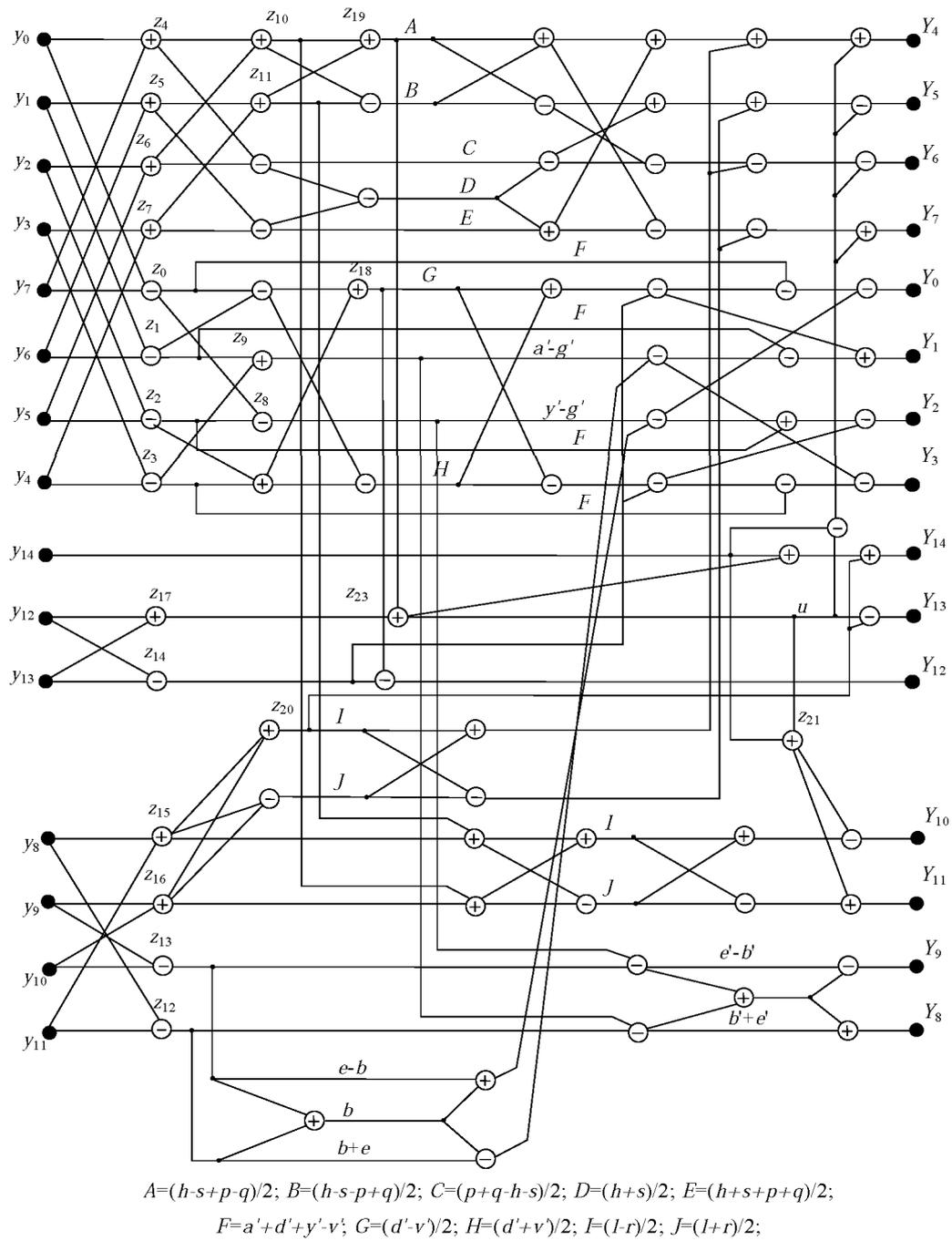


Рис. 3.5. Схема (направленный граф) алгоритма ДКП длины  $N = 15$

### 3.2.5. Некоторые экспериментальные результаты

В качестве основы для сравнительного анализа вычислительной сложности синтезированных алгоритмов был использован алгоритм работы [15] синтезированный для ДКП произвольных длин, оценки сложности которого при  $N = 2^k$  совпадают с оценками сложности лучших из известных алгоритмов ДКП [9, 20, 29].

В таблице 3.4 приведено количество операций необходимых для вычисления ДКП предложенным алгоритмом и известным способом. На рис.3.6 приводится зависимость удельной мультипликативной сложности алгоритмов от длины преобразования.

Таблица 3.4. Количество операций необходимых для вычисления ДКП

$N$	Предложенные алгоритмы		Алгоритм работы [15]	
	умножений	сложений	умножений	сложений
8	10	32	12	29
9	8	44	11	44
10	9	43	15	36
12	13	51	20	43
15	21	82	35	89



Рис. 3.6. Удельная мультипликативная сложность алгоритмов ДКП

На рис. 3.7 показана относительная характеристика  $\tau_N/\tau_8$ , где  $\tau_8$  - это время обработки изображения блоками  $8 \times 8$  ( $\tau_N/\tau_8 = 1$  при  $N=8$ ), размер изображения -  $1024 \times 1024$  пиксела. Экспериментальное исследование зависимости времени  $\tau_N$

обработки изображения блочным ДКП от размера квадратного блока  $N$  позволяет утверждать, что предложенные алгоритмы ДКП гарантируют скорость обработки изображения близкую к скорости обработки лучшим из известных алгоритмов ДКП длины 8. Время обработки практически не возрастает с ростом  $N$ .

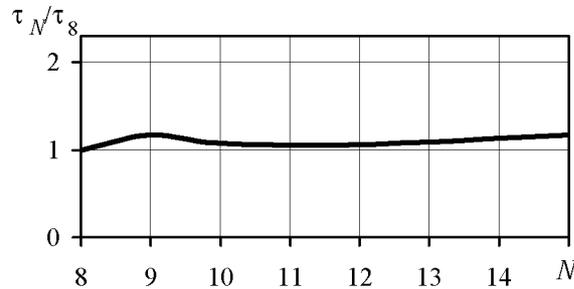


Рис. 3.7.. Относительное время обработки изображения  $1024 \times 1024$  пиксела блочным ДКП

В таблице 3.5. Приводятся сравнительные характеристики лучших с точки зрения вычислительной сложности алгоритмов ДКП длины  $N = 8$ .

Таблица 3.5 Сравнительные характеристики лучших алгоритмов ДКП длины  $N = 8$

	Умножений	Сложений
Автор(ы)		
Chen, Smith, Fralick, 1977 [10]	16	26
Lee, 1984, [23]	12	29
Нои, 1987, [19]	12	29
Loeffler, Ligtenberg, Moschytz, 1989, [24]	11	29
Чернов, Чичева, [12]	10	32

### 3.3. Алгоритмы ДКП, реализуемые в системах счисления с "косинусными" основаниями

Рассмотрим популярную формулу элементарной тригонометрии:

$$2 \cos\left(\frac{\varphi}{2}\right) = \sqrt{2 + 2 \cos \varphi}$$

с естественными ограничениями на величину угла  $\varphi$ . Далее последовательно получаем

$$2 \cos\left(\frac{\varphi}{4}\right) = \sqrt{2 + \sqrt{2 + 2 \cos \varphi}}, \quad 2 \cos\left(\frac{\varphi}{8}\right) = \sqrt{2 + \sqrt{2 + \sqrt{2 + 2 \cos \varphi}}}$$

и так далее. Если значение  $\cos \varphi$  является квадратичной иррациональностью, то, например, для  $\varphi = \frac{\pi}{4}$  имеем

$$\xi = 2 \cos\left(\frac{\pi}{16}\right) = \sqrt{2 + \sqrt{2 + \sqrt{2}}}.$$

Легко проверить последовательным возведением в квадрат, что число  $\xi$  является корнем алгебраического уравнения

$$\xi^8 - 8\xi^6 + 20\xi^4 - 16\xi^2 + 2 = 0,$$

то есть целым алгебраическим числом. Непосредственно устанавливается справедливость равенств

$$\begin{aligned} 2 \cos\left(\frac{2\pi}{16}\right) &= \sqrt{2 + \sqrt{2}} & 2 \cos\left(\frac{5\pi}{16}\right) &= \sqrt{2 - \sqrt{2} - \sqrt{2}} \\ 2 \cos\left(\frac{3\pi}{16}\right) &= \sqrt{2 + \sqrt{2 - \sqrt{2}}} & 2 \cos\left(\frac{6\pi}{16}\right) &= \sqrt{2 - \sqrt{2}} \\ 2 \cos\left(\frac{4\pi}{16}\right) &= \sqrt{2} & 2 \cos\left(\frac{7\pi}{16}\right) &= \sqrt{2 - \sqrt{2 + \sqrt{2}}} \end{aligned}.$$

Из этих равенств следует, что числа  $2 \cos\left(\frac{k\pi}{16}\right)$ ,  $(k = 0, 1, \dots, 7)$  представимы в "системе счисления с основанием  $\xi$ ":

$$2 \cos\left(\frac{k\pi}{16}\right) = \sum_{j=0}^7 a_j^{(k)} \xi^j$$

с целыми коэффициентами – "цифрами"  $a_j^{(k)}$ . Следовательно, умножение на  $2 \cos\left(\frac{k\pi}{16}\right)$  чисел, представленных в системе счисления с основанием  $\xi$ , сводится к умножению полиномов с целыми коэффициентами.

В таблице 3.6. приведены значения цифр  $a_j^{(k)}$ . (Верхний индекс опущен:  $a_j^{(k)} = a_j$ ).

Таблица 3.6

	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$
$2 \cos\left(\frac{0\pi}{16}\right)$	2	0	0	0	0	0	0	0
$2 \cos\left(\frac{1\pi}{16}\right)$	0	1	0	0	0	0	0	0
$2 \cos\left(\frac{2\pi}{16}\right)$	-2	0	1	0	0	0	0	0
$2 \cos\left(\frac{3\pi}{16}\right)$	0	-3	0	1	0	0	0	0
$2 \cos\left(\frac{4\pi}{16}\right)$	2	0	-4	0	1	0	0	0
$2 \cos\left(\frac{5\pi}{16}\right)$	0	5	0	-5	0	1	0	0
$2 \cos\left(\frac{6\pi}{16}\right)$	-2	0	9	0	-6	0	1	0
$2 \cos\left(\frac{7\pi}{16}\right)$	0	-7	0	14	0	-7	0	1

Из таблицы видно, что целые числа  $a_j^{(k)}$  имеют относительно маленький динамический диапазон, "малобитовы". Следовательно, умножение данных, представленных в системе счисления с основанием  $\xi$  может быть реализовано посредством только сложений.

Отсюда следует алгоритм вычисления ДКП длины  $N = 8$ , эффективность которого определяется, в основном, эффективностью аппаратной или программной реализации алгоритма перевода чисел из системы счисления с основанием  $\xi$  в "общепринятую" систему счисления.

#### 4. ПРИМЕНЕНИЕ ДКП В ЗАДАЧАХ ОБРАБОТКИ ИЗОБРАЖЕНИЙ НА ПРИМЕРЕ КОМПРЕССИИ ИЗОБРАЖЕНИЙ

В настоящее время чрезвычайно широкое распространение получил метод блочного кодирования изображений, который базируется на использовании обобщенного спектрального представления сигнала. Эффективность метода обуславливается тем, что спектральные компоненты изображения (трансформанты) могут быть статистически более независимы, чем сами отсчеты сигнала, и, следовательно, нести ту же информацию при меньшем объеме передаваемых цифровых данных.

Выбор преобразования в этом методе обусловлен следующими требованиями: преобразование должно быть обратимым, основная информация о сигнале должна быть сосредоточена по возможности в меньшем числе трансформант для обеспечения эффекта сжатия, прямое и обратное преобразование должны легко вычисляться. Идеальным по второму требованию является преобразование Хотеллинга (дискретная версия Карунена-Лоэва), у которого наиболее быстро убывают дисперсии трансформант. Однако на практике оно не используется, так как его базис жестко привязан к автоковариационной функции сигнала и его необходимо строить заново для каждого нового класса сигналов. Кроме того, это преобразование не обладает быстрым алгоритмом. Поэтому, как правило, используются известные алгоритмы спектральных преобразований, несколько проигрывающие в качестве, но выигрывающие в скорости. Наиболее близким по своим характеристикам к преобразованию Карунена-Лоэва для широкого класса изображений является дискретное косинусное преобразование (1.1).

Схема метода представлена на рис.4.1. Входное изображение разбивается на квадратные блоки размером  $M \times N$  отсчетов, в каждом блоке выполняется двумерное дискретное косинусное преобразование. Затем производится отбор существенных трансформант и их квантование. Совокупность отобранных и квантованных трансформант для всех блоков составляет содержание сжатых данных.

Ниже описываются некоторые экспериментальные результаты по кодированию изображений с помощью ДКП блоками  $N \times N$  при различных значениях  $N$ .

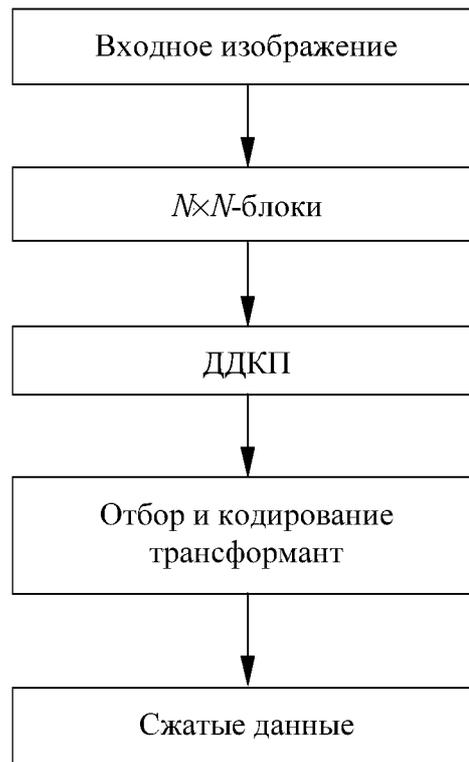


Рис.4.1. Схема метода кодирования с преобразованием.

Входное изображение подвергалось кодированию с заданными коэффициентом сжатия  $K_c$ , размером блока  $N$  и типом преобразования  $Type$ , затем восстанавливалось, и по разности исходного и восстановленного изображений определялась среднеквадратичная ошибка (ошибка сжатия), внесенная в данные сквозной процедурой кодирования/декодирования. Ниже для краткости будем называть ее ошибкой сжатия (см. рис.4.2).

На рис.4.3, 4.4, 4.5 приведены тестовые изображения "Портрет", "Аэрофотосъемка", "Глазное дно" и результаты их сжатия/восстановления. Приведенные зависимости позволяют сделать вывод о том, что в принятом методе компрессии применение ДКП устойчиво дает хорошее качество сжатия. Расширение набора длин, для которых существуют эффективные алгоритмы ДКП, позволяет выбрать размер блока, при котором ошибка сжатия при заданном коэффициенте сжатия будет наименьшей. Использование блоков нестандартных размеров ( $N \neq 8$ ) позволяет уменьшить ошибку сжатия в 1,3 - 1,5 раза. Полученные результаты подтверждаются и визуальным качеством восстановленных изображений.

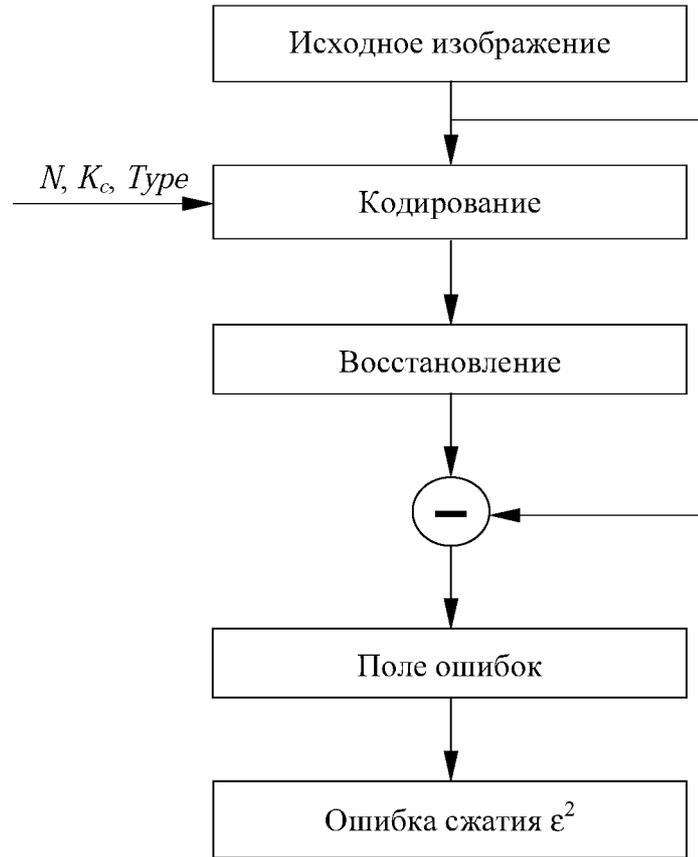


Рис.4.2. Схема эксперимента по кодированию изображений.

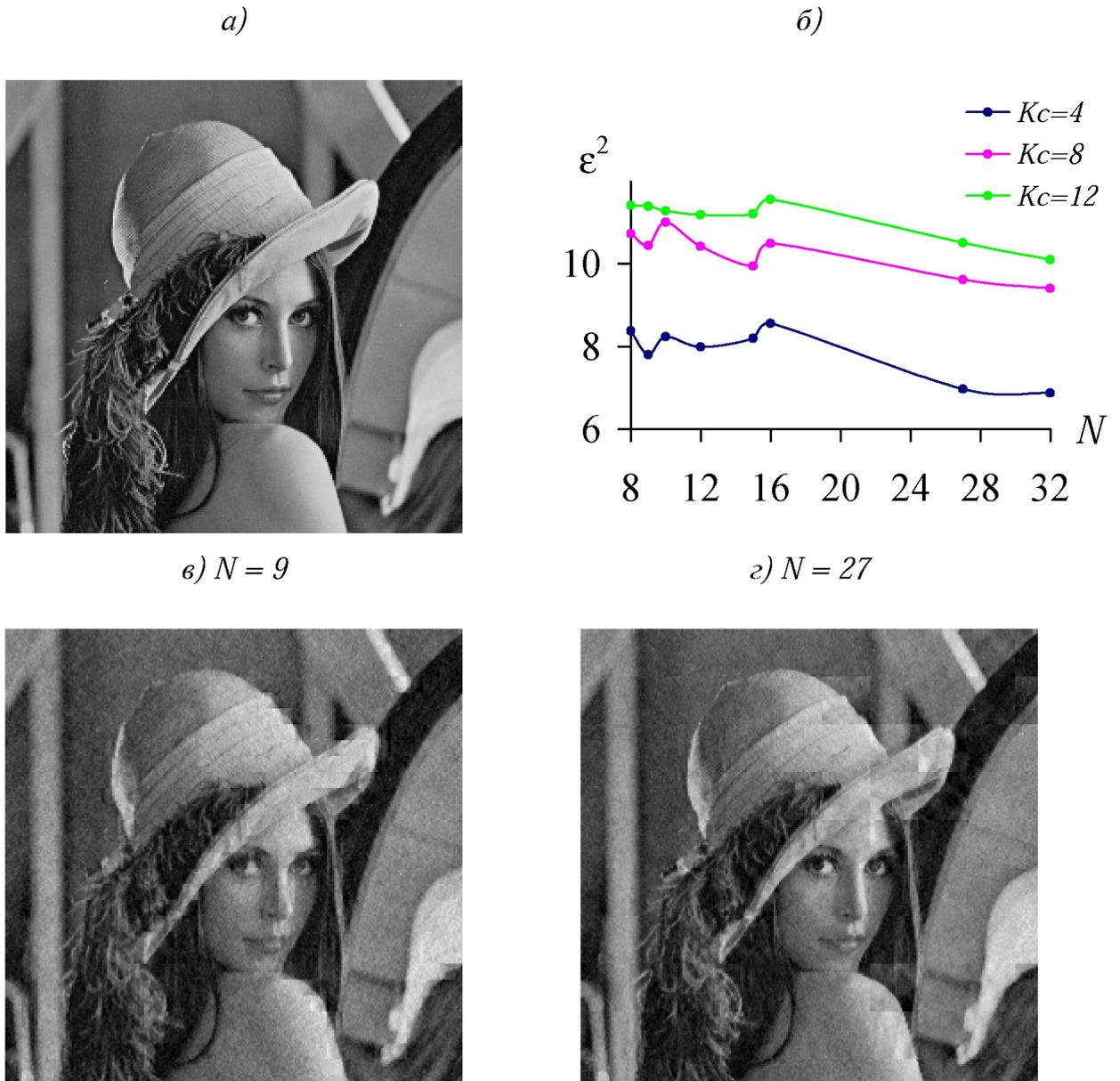


Рис.4.3. Результаты кодирования изображения "Портрет".

а) Исходное изображение.

б) Зависимость ошибки сжатия от размера блока.

в) Восстановленное изображения после кодирования при коэффициенте сжатия  $K_c=6$  размере блока  $N=9$ .

г) Восстановленное изображения после кодирования при коэффициенте сжатия  $K_c=6$  размере блока  $N=27$ .

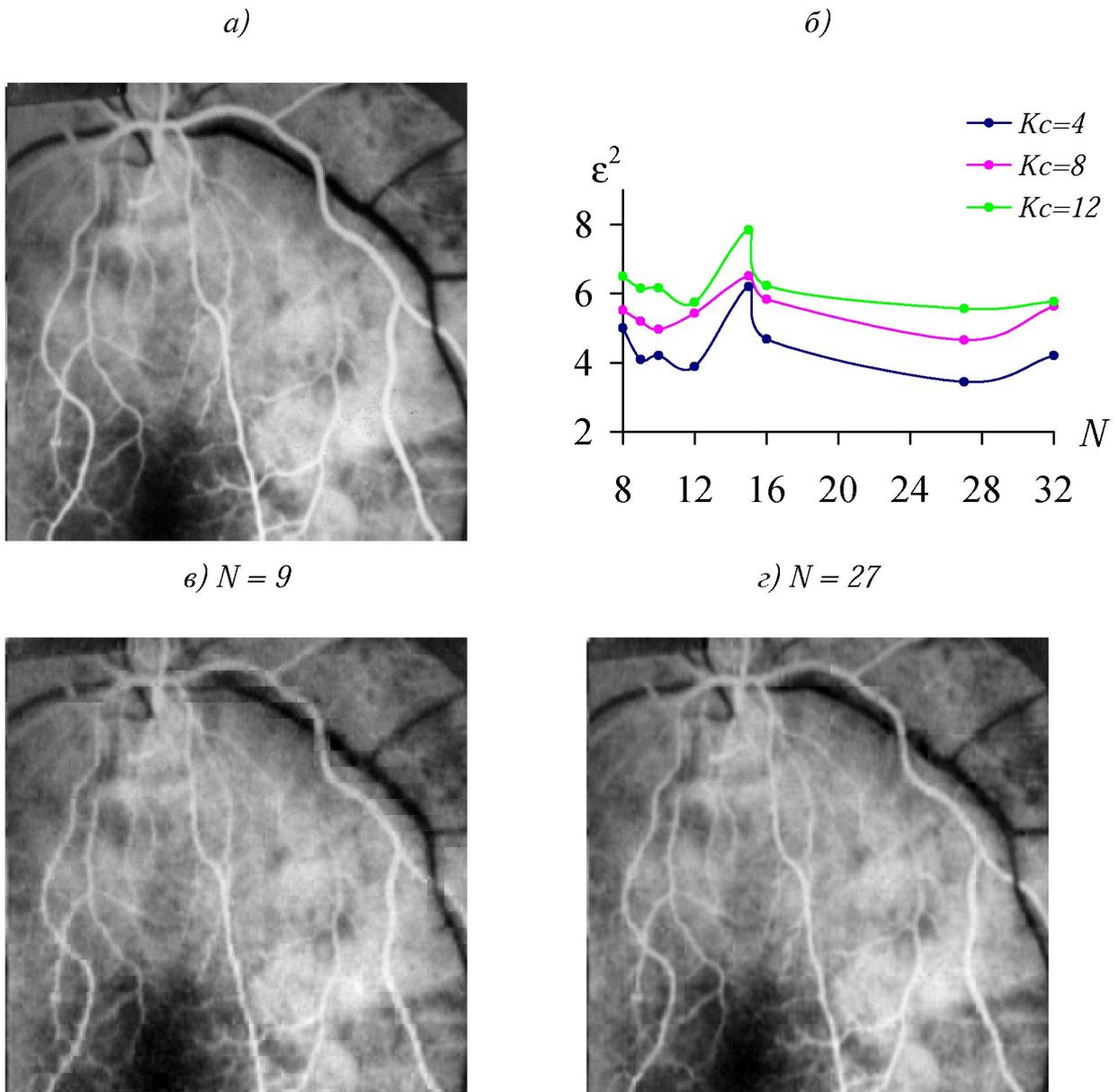


Рис.4.4. Результаты кодирования изображения "Глазное дно".

а) Исходное изображение.

б) Зависимость ошибки сжатия от размера блока.

в) Восстановленное изображения после кодирования при коэффициенте сжатия  $K_c=10$  размере блока  $N=9$ .

г) Восстановленное изображения после кодирования при коэффициенте сжатия  $K_c=10$  размере блока  $N=27$ .

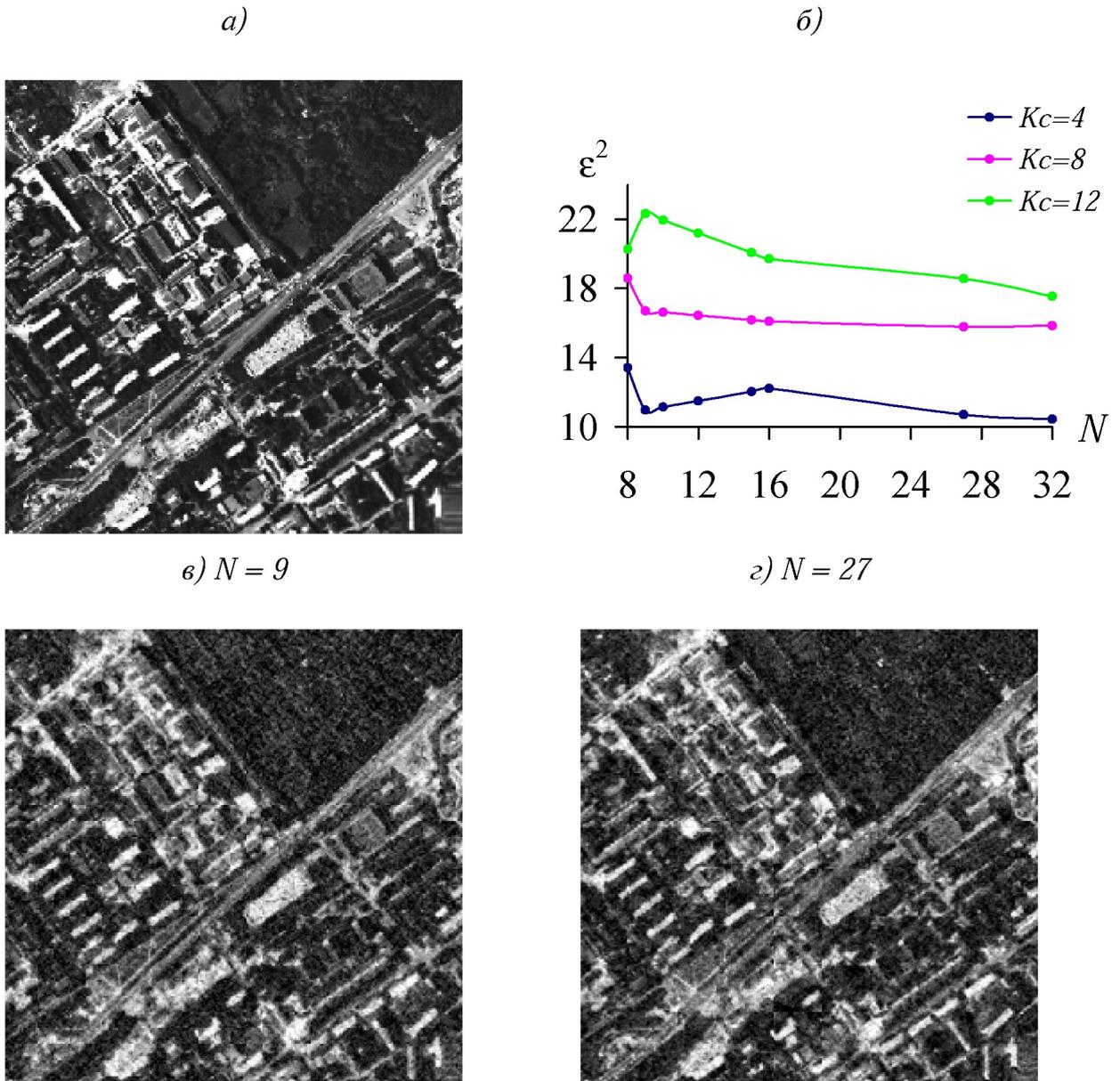


Рис. 4.5. Результаты кодирования изображения "Аэрофотосъемка".

а) Исходное изображение.

б) Зависимость ошибки сжатия от размера блока.

в) Восстановленное изображения после кодирования при коэффициенте сжатия  $K_c=8$  размере блока  $N=9$ .

г) Восстановленное изображения после кодирования при коэффициенте сжатия  $K_c=8$  размере блока  $N=27$ .

## 5. ЗАДАЧИ, УПРАЖНЕНИЯ, КОНТРОЛЬНЫЕ ВОПРОСЫ

### 5.1. Задачи, упражнения

5.1.1. По аналогии с п.1.3.6 постройте алгоритм одномерного ДПФ с декомпозицией по основанию 5. Как будут выглядеть в этом случае  $\gamma$ -коды? Выведите оценки сложности синтезированного алгоритма.

5.1.2. Синтезируйте соответствующий алгоритм ДКП длины  $N = 5^k$ . Покажите сведение ДКП к ДПФ. Выведите оценки вычислительной сложности.

5.1.3. Покажите применение алгоритма из п.2.2. для  $N=16$ .

5.1.4. Рассчитайте количество арифметических операций для обработки изображения размером  $2048 \times 2048$  пикселей блочным ДКП. Используйте алгоритмы раздела 3, в предположении, что для двумерного блока используется построчно-столбцовый метод.

### 5.2. Контрольные вопросы

5.2.1. Какие методы декомпозиции одномерного ДПФ вы знаете? Какой из них наиболее эффективен в вычислительном отношении? Какой наиболее просто реализовать?

5.2.2. В чем преимущество алгоритмов ДКП из пп.2.2, 2.3 по отношению к традиционному способу вычисления ДКП п.2.1?

5.2.3. С какой целью разрабатываются специальные алгоритмы ДКП коротких длин?

5.2.4. Назовите основные принципы построения алгоритмов ДКП коротких длин.

5.2.5. В каких задачах обработки изображений используется ДКП? Какие свойства ДКП определяют его применение в этих задачах?

## ЛИТЕРАТУРА

1. Ахмед Н., Рао К. Р. Ортогональные преобразования при обработке цифровых сигналов. Пер. с англ. - М.: Связь, 1980. - 248 с.
2. Блейхут Р. Быстрые алгоритмы цифровой обработки сигналов. Пер. с англ. М.: Мир, 1989. - 448 с.
3. Власенко В. А., Лаппа Ю. М., Ярославский Л. П. Методы синтеза быстрых алгоритмов свертки и спектрального анализа сигналов. - М.: Наука, 1990. - 160 с.
4. Нуссбаумер Г. Быстрое преобразование Фурье и алгоритмы вычисления сверток. Пер. с англ. - М.: Радио и связь, 1985.- 248 с.
5. Чернов В. М. Алгоритмы двумерных дискретных ортогональных преобразований, реализуемые в кодах Гамильтона-Эйзенштейна. //Проблемы передачи информации, том 31, № 3, с.38-46, 1995
6. Чернов В. М. Арифметические аспекты задачи синтеза быстрых алгоритмов ДОП типа Винограда. //Труды Пятого Международного Семинара "Распределенная обработка информации" - Новосибирск, 1995. - с. 296-300.
7. Чернов В. М. *Быстрый алгоритм дискретного косинусного преобразования нечетной длины.* //Автомат. и вычисл. техн., № 3, 1994. - с.62-70
8. Chan S.-C., Ho K.-L. Fast algorithms for computing the discrete cosine transform. //IEEE Trans. on Circuits and Systems, Vol. 39, No 3, pp.185-190, 1992
9. Chan-Wan Y.-H., Siu C. On the realization of discrete cosine transform using the distributed arithmetic. //IEEE Transactions on Circuits and Systems - I: Fundamental Theory and Applications, Vol. 39, No 9, pp. 705-712, 1992
10. Chen W.H., Smith C.H., Fralick S.C. A fast computational algorithm for discrete cosine transform. IEEE Trans. on Comm., 1977, pp.1004-1009.
11. Chernov V. M. *Fast algorithms of discrete orthogonal transforms for data represented in cyclotomic fields.* //Pattern Recognition and Image Analysis, Vol.3, No 4, pp. 455-458, 1993
12. Chichyeva M. A., Chernov V. M. "One-step" short-length DCT algorithms with data representation in the direct sum of associative algebras. //Proceedings CAIP'97, Springer, LNCS 1296, pp.590-596, 1997

13. De Natale F. G. B., Desoli G. S., Giusto D. D., Vernazza G. Adaptive DCT for image-data compression. //Eur. Trans. Telecommun. and Relat. Technol., Vol. 3, No 4, pp. 359-366, 1992
14. Duhamel P. Implementation of split-radix FFT algorithms for complex, real, and real-symmetric data. //IEEE Trans. Accoust., Speech, and Signal Process., 1984, Vol.32, No 4, pp.750-761
15. Duhamel & Guillemot Polynomial Transform Computation of the 2-D DCT. //ICASSP '90 p.1515.
16. Ersoy O. K., Chen C. H. Transform-coding of images with reduced complexity. //Computer Vision, Graphics, and Image Processing, No 42, pp.19-31, 1988
17. Good I. J. *The interaction algorithm and practical Fourier analysis.* //J. Royal Statist. Soc., Ser. B., No 20, pp. 361-375, 1958
18. Heideman M. T. Computation of an odd-length DCT from a real-valued DFT of the same length. //IEEE Trans. Signal Process., Vol. 40, No 1, pp.54-61, 1992
19. Hou H.S. A fast recursive algorithm for computing the discrete cosine transform. IEEE Trans. ASSP, 1987, pp. 1455-1461
20. Hou H. S., Tretter D. K. Interesting properties of the discrete cosine transform. //J. Visual Commun. and Image Represent., Vol. 3, No 1, pp. 73-83, 1992
21. Jeong J., Jo J. M. Adaptive Huffman coding of 2-D DCT coefficients for image sequence compression. //Signal Processing: Image Communication, Vol. 7, Issue 1, pp. 1-11, 1995
22. Krupiczka A. Interblock variance as a segmentation criterion in image coding. //Machine Graphics and Vision, Vol. 5, Nos 1/2, pp. 229-235, 1996
23. Lee B.G. A new algorithm to compute the discrete cosine transform. IEEE Trans. ASSP, 1984, pp. 1243-1245
24. Loeffler, Ligtenberg & Moschytz, Practical Fast 1-D DCT Algorithms with 11 Multiplications. //ICASSP '89 p. 988.
25. Rao K. R., Yip P. Discrete Cosine Transform. Academic Press, San Diego, 1990.
26. Rozinaj G., Marček A. Approximation of DCT without multiplication in JPEG. //Proc. of the Third IEEE International Conference on Electronics, Circuits, and Systems, Vol.2, pp.704-707, 1996

27. Schafer R.D. An introduction to nonassociative algebras. London: Academic Press, 1966
28. Sikora T. Low complexity shape-adaptive DCT for coding of arbitrarily shaped image segments. //Signal Processing: Image Communication, Special Issue on Coding Techniques for Very Low Bit-rate Video, Vol. 7, Issue 4-6, pp. 381-395, 1995
29. Suheiro N., Hatori M. Fast algorithms for the DFT and other sinusoidal transforms. //IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP-34, No 6, pp. 642-644, 1986
30. Thomas L. H. Using a computer to solve problems in physics, in applications and of digital computer. Boston, Mass.: Ginn and Co., 1963
31. Wallace G. K. The JPEG still picture compression standard. //Communications of the ACM, Vol. 34, No 4, pp. 31-44, 1991.

## СОДЕРЖАНИЕ

1. Предварительные сведения из теории быстрых алгоритмов дискретных ортогональных преобразований.....	1
1.1. Постановка задачи, основные идеи .....	3
1.2. Сложность операции умножения в конечномерных алгебрах .....	6
1.3. Основные схемы редукции вспомогательного дискретного преобразования Фурье (ДПФ).....	10
1.3.1. Декомпозиция Кули-Тьюки "по основанию 2" .....	10
1.3.2. Декомпозиция Кули-Тьюки "по основанию 4" .....	11
1.3.3. Декомпозиция Кули-Тьюки с расщеплением основания (сплит-радикс алгоритм) .....	12
1.3.4. Декомпозиция ДПФ Гуда-Томаса.....	13
1.3.5. Декомпозиция Кули-Тьюки "по основанию $p$ " .....	14
1.3.6. Алгоритм одномерного ДПФ при длине преобразования $N=3^k$ .....	16
2. Алгоритмы одномерного ДКП.....	18
2.1. Традиционный способ вычисления ДКП.....	18
2.2 Современный алгоритм ДКП четной длины.....	18
2.3. Сведение ДКП нечетной длины к ДПФ вещественной последовательности той же длины.....	22
2.4. Пример. Алгоритм дискретного косинусного преобразования длиной $N=3^k$ .....	23
3. Алгоритмы ДКП коротких длин .....	25
3.1. Алгебраические принципы синтеза алгоритмов ДКП коротких длин.....	25
3.1.1. Дискретное косинусное преобразование длины 8 .....	25
3.1.2. Связь между алгебраическими свойствами значений базисных функций ДКП и структурой матрицы преобразования .....	28
3.2. Примеры алгоритмов ДКП с минимальной арифметической сложностью .....	31
3.2.1. Алгоритм ДКП длины 9 .....	31
3.2.2. Алгоритм ДКП длины 10 .....	34
3.2.3. Алгоритм ДКП длины 12 .....	36
3.2.4. Алгоритм ДКП длины 15 .....	38
3.2.5. Некоторые экспериментальные результаты .....	40

3.3. Алгоритмы ДКП, реализуемые в системах счисления с "косинусными" основаниями.....	41
4. Применение ДКП в задачах обработки изображений на примере компрессии изображений .....	44
5. Задачи, упражнения, контрольные вопросы .....	50
5.1. Задачи, упражнения.....	50
5.2. Контрольные вопросы.....	50
Литература.....	51