

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»
КАФЕДРА ЕЛЕКТРОННОЇ ТЕХНІКИ

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних і практичних робіт та розрахункової роботи з дисциплін «Метрологічне забезпечення систем технічного захисту інформації», «Методи дослідження та аналізу інформаційних даних», «Електронні системи» для студентів денної та заочної форм навчання усіх спеціальностей

Луцьк – 2023

УДК 681.51(072)

M54

Методичні вказівки до виконання лабораторних і практичних робіт та розрахункової роботи з дисциплін «Метрологічне забезпечення систем технічного захисту інформації», «Методи дослідження та аналізу інформаційних даних», «Електронні системи» (для студентів денної та заочної форм навчання усіх спеціальностей) [Електронний ресурс] / уклад. : О.В. Вовна, Г.О. Шеїна. – Луцьк : ДонНТУ, 2023. – 68 с.

Методичні вказівки є лабораторним практикумом з дисциплін «Метрологічне забезпечення систем технічного захисту інформації», «Методи дослідження та аналізу інформаційних даних», «Електронні системи», у них викладено необхідні положення під час підготовки до лабораторних робіт і практичних занять, порядок виконання та завдання до лабораторних робіт, практичних занять та розрахункової роботи, зміст звіту та контрольні запитання.

Укладачі: Вовна О.В. – професор, д.т.н., зав. каф. електронної техніки.

Шеїна Г.О. – доцент, к.т.н., доцент каф. електронної техніки.

Рецензент Поцєпаєв В.В. – доцент, к.т.н., зав. каф. автоматики та телекомунікацій.

Відповідальний за випуск: Вовна О.В. – завідувач кафедри електронної техніки, д.т.н., професор.

Затверджено навчально-методичним відділом ДонНТУ,
протокол № 4 від 31.01.2023 р.

Розглянуто на засіданні кафедри електронної техніки,
протокол № 7 від 19.01.2023 р.

© ДонНТУ, 2023 рік

ЗМІСТ

ВСТУП	4
ЛАБОРАТОРНА РОБОТА № 1 – Спектральний аналіз	5
ЛАБОРАТОРНА РОБОТА № 2 – Дослідження ентропії	13
ЛАБОРАТОРНА РОБОТА № 3 – Непозиційне кодування. Дослідження коду Грея	24
ЛАБОРАТОРНА РОБОТА № 4 – Дослідження ефективних кодів. Код Шеннона – Фано	31
ЛАБОРАТОРНА РОБОТА № 5 – Дослідження ефективних кодів. Код Хаффмена	37
ЛАБОРАТОРНА РОБОТА № 6 – Дослідження завадостійких кодів. Код Хеммінга	57
Порядок виконання і оформлення розрахункової роботи	67
Список рекомендованої літератури	68

ВСТУП

Дисципліна метрологічне забезпечення систем технічного захисту інформації – розв’язує проблему захисту інформації шляхом її перетворення.

З цією метою в перших двох розділах методичних вказівок розглянуті базові питання теорії інформації, такі як кількісні оцінки інформації, характеристики каналів зв’язку, спектральний аналіз сигналів.

В наступному розділі методичних вказівок розглянутий код Грея, який є прикладом непозиційного кодування, коли положення символу не залежить від його значення.

Два розділи методичних вказівок присвячені ефективному кодуванню, яке забезпечує мінімального завантаження каналу зв’язку, так як символи первинного алфавіту, імовірність появи яких є високою, формують короткий код, а символи, які зустрічаються рідко – довгий код. Спочатку досліджувався принцип побудови коду Шенона-Фано й його кодового дерева. Наступним розглядався принцип побудови коду Хаффмена, його кодове дерево, матриця ідентифікації коду Хаффмена, принцип побудови коду Хаффмена на підставі матриці сум і матриці номерів елементів.

В останньому розділі методичних вказівок наводилися двійкові коди, що виявляють та виправляють помилки: код з перевіркою на парність; код з перевіркою на непарність; інверсний код; кореляційний код; код Хеммінга.

Кожний розділ завершується прикладами та контрольними питаннями, які допоможуть кращому засвоєнню матеріалу та набуттю практичних навичок у використанні методів забезпечення систем технічного захисту інформації.

ЛАБОРАТОРНА РОБОТА № 1

СПЕКТРАЛЬНИЙ АНАЛІЗ ПЕРІОДИЧНИХ СИНАЛІВ.

ПЕРЕТВОРЕННЯ ФУР'Є

Мета роботи: дослідити перетворення Фур'є для різних видів сигналів.

Теоретична частина

Будь який сигнал може бути розкладений на сукупність простих сигналів – спектр. Залежність амплітуд сукупності сигналів від частоти – це амплітудно-частотний спектр. Залежність початкових фаз сукупності сигналів від частоти – це фазово-частотний спектр.

Для розкладання сигналу на сукупність простих сигналів виконується за допомогою перетворень Фур'є.

Якщо функція $f(x)$ періодична ($T=2\pi$, $f(x+T)=f(x)$), кусочно-монотонна і обмежена на відрізку $[-\pi, \pi]$, то ряд Фур'є, побудований для цієї функції, визначається у всіх точках:

$$\begin{aligned} f(x) &= \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) = \\ &= \frac{a_0}{2} + a_1 \cos x + b_1 \sin x + a_2 \cos 2x + b_2 \sin 2x + \dots \end{aligned}$$

$$\text{де } a_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) dx, \quad a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(nx) dx, \quad b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(nx) dx.$$

Якщо функція парна $f(-x) = f(x)$, то

$$\int_{-\pi}^{\pi} f(x) dx = 2 \int_0^{\pi} f(x) dx, \quad a_n = \frac{2}{\pi} \int_0^{\pi} f(x) \cos(nx) dx, \quad b_n = 0.$$

Ряд Фур'є парної функції включає тільки косинуси:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx).$$

Якщо функція непарна $f(-x) = -f(x)$, то

$$\int_{-\pi}^{\pi} f(x) dx = 0, \quad a_0 = 0, \quad a_n = 0, \quad b_n = \frac{2}{\pi} \int_0^{\pi} f(x) \sin(nx) dx.$$

Ряд Фур'є непарної функції включає тільки синуси:

$$f(x) = \sum_{n=1}^{\infty} (b_n \sin nx).$$

Перетворення Фур'є розглянуто в практичній частині роботи.

Практична частина

Приклад 2.1. (Перетворення Фур'є).

Розкласти в ряд Фур'є функцію (рис. 1.1) на інтервалі $[-\pi; \pi]$:

$$f(x) = \begin{cases} 0, & -\pi \leq x < 0 \\ \pi - x, & 0 \leq x \leq \pi \end{cases}$$

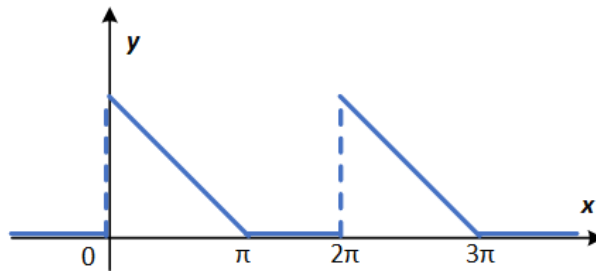


Рисунок 1.1 – Вихідна функція

Коефіцієнти ряду Фур'є:

$$\begin{aligned} a_0 &= \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) dx = \frac{1}{\pi} \left[\int_{-\pi}^0 0 dx + \int_0^{\pi} (\pi - x) dx \right] = \\ &= \frac{1}{\pi} \left[\pi x - \frac{x^2}{2} \right]_0^{\pi} = \frac{1}{\pi} \left[\pi^2 - \frac{\pi^2}{2} \right] = \frac{\pi}{2} \end{aligned}$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos nx dx = \frac{1}{\pi} \int_0^{\pi} (\pi - x) \cos nx dx =$$

$$\left\{ \begin{array}{l} \int u dv = uv - \int v du \quad u = \pi - x \quad dv = \cos nx dx \\ du = (\pi - x)' dx = -dx \quad v = \int \cos nx dx = \frac{1}{n} \int \cos nx d(nx) = \frac{1}{n} \sin nx \end{array} \right\}$$

$$= \frac{1}{\pi} \left[\left(\frac{\pi - x}{n} \sin nx \right) \Big|_0^{\pi} + \frac{1}{n} \int_0^{\pi} \sin nx dx \right] =$$

$$= \frac{1}{\pi} \left[\frac{\pi - \pi}{n} \sin n\pi - \frac{\pi - 0}{n} \sin n0 + \frac{1}{n^2} \int_0^{\pi} \sin nx dnx \right] =$$

$$= \frac{1}{\pi} \left[-\frac{1}{n^2} \cos nx \Big|_0^{\pi} \right] =$$

$$= \frac{1}{\pi} \left[-\frac{1}{n^2} \cos n\pi + \frac{1}{n^2} \cos n0 \right] = \frac{1}{\pi} \left[\frac{1}{n^2} - \frac{(-1)^n}{n^2} \right] = \frac{1}{\pi n^2} (1 - (-1)^n) =$$

$$= \begin{cases} 0, & \text{если } n = 2k \\ \frac{2}{\pi(2n-1)^2}, & \text{если } n = 2k-1. \end{cases}$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin nx dx = \frac{1}{\pi} \int_0^{\pi} (\pi - x) \sin nx dx =$$

$$\left\{ \begin{array}{l} \int u dv = uv - \int v du \quad u = \pi - x \quad dv = \sin nx dx \\ du = (\pi - x)' dx = -dx \quad v = \int \sin nx dx = -\frac{1}{n} \int \sin nx d(nx) = \frac{1}{n} \cos nx \end{array} \right\}$$

$$= \frac{1}{\pi} \left[\left(-\frac{(\pi - x)}{n} \cos nx \right) \Big|_0^{\pi} - \frac{1}{n} \int_0^{\pi} \cos nx dx \right] =$$

$$= \frac{1}{\pi} \left[0 + \frac{\pi}{n} \cos n0 - \frac{1}{n^2} \int_0^{\pi} \cos nx dnx \right] =$$

$$= \frac{1}{\pi} \left[\frac{\pi}{n} - \frac{1}{n^2} \sin nx \Big|_0^{\pi} \right] = \frac{1}{\pi} \left[\frac{\pi}{n} - \frac{1}{n^2} \sin n\pi + \frac{1}{n^2} \sin n0 \right] = \frac{1}{n}$$

Відповідь:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) = \frac{\pi}{4} + \sum_{n=1}^{\infty} \left(\frac{2}{\pi(2n-1)^2} \cos(2n-1)x + \frac{1}{n} \sin nx \right).$$

Приклад 2.2. (Перетворення Фур'є).

Розкласти в ряд Фур'є функцію $f(x) = e^x$, задав її на інтервалі $[0; \pi]$, продовжив її парним і непарним чином.

Якщо продовжити функцію $f(x) = e^x$ парним чином (рис. 1.2), то коефіцієнти ряду Фур'є:

$$a_0 = \frac{2}{\pi} \int_0^{\pi} e^x dx = \frac{1}{\pi} = \frac{2}{\pi} \cdot e^x \Big|_0^{\pi} = \frac{2}{\pi} (e^{\pi} - 1);$$

$$b_n = 0; \quad a_n = \frac{2}{\pi} \int_0^{\pi} e^x \cos nx dx.$$

Роздивимося наступні спрощення:

$$\begin{aligned} \int_0^{\pi} e^x \cos nx dx &= \\ \left\{ \begin{array}{l} u = e^x \quad dv = \cos nx dx \\ du = e^x dx \quad v = \frac{1}{n} \sin nx \end{array} \right\} &= e^x \cdot \frac{1}{n} \sin nx - \frac{1}{n} \cdot \int_0^{\pi} e^x \sin nx dx = \\ \left\{ \begin{array}{l} u = e^x \quad dv = \sin nx dx \\ du = e^x dx \quad v = -\frac{1}{n} \cos nx \end{array} \right\} &= e^x \cdot \frac{1}{n} \sin nx - \frac{1}{n} \cdot \left[e^x \cdot -\frac{1}{n} \cos nx + \frac{1}{n} \cdot \int_0^{\pi} e^x \cos nx dx \right] = \\ &= \frac{1}{n} e^x \sin nx + \frac{1}{n^2} e^x \cos nx - \frac{1}{n^2} \cdot \int_0^{\pi} e^x \cos nx dx \end{aligned}$$

$$\int_0^{\pi} e^x \cos nx dx = \frac{1}{n} e^x \sin nx + \frac{1}{n^2} e^x \cos nx - \frac{1}{n^2} \cdot \int_0^{\pi} e^x \cos nx dx;$$

$$\int_0^{\pi} e^x \cos nx dx = \frac{n^2}{n^2 + 1} \left(\frac{1}{n} e^x \sin nx + \frac{1}{n^2} e^x \cos nx \right);$$

$$\begin{aligned}
 a_n &= \frac{2}{\pi} \int_0^{\pi} e^x \cos nx \, dx = \frac{2}{\pi} \frac{n^2}{n^2+1} \left(\frac{1}{n} e^x \sin nx + \frac{1}{n^2} e^x \cos nx \right) \Bigg|_0^{\pi} = \\
 &= \frac{2}{\pi} \frac{n^2}{n^2+1} \left(\frac{1}{n} e^{\pi} \sin n\pi + \frac{1}{n^2} e^{\pi} \cos n\pi - \left(\frac{1}{n} e^0 \sin n0 + \frac{1}{n^2} e^0 \cos n0 \right) \right) = \\
 &= \frac{2}{\pi} \frac{n^2}{n^2+1} \left(\frac{1}{n^2} e^{\pi} (-1)^n - \frac{1}{n^2} \right) = \frac{2}{\pi(n^2+1)} (e^{\pi} (-1)^n - 1).
 \end{aligned}$$

В результаті:

$$f(x) = e^x = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) = \frac{1}{\pi} (e^{\pi} - 1) + \sum_{n=1}^{\infty} \left[\frac{2}{\pi(n^2+1)} (e^{\pi} (-1)^n - 1) \cos nx \right]$$

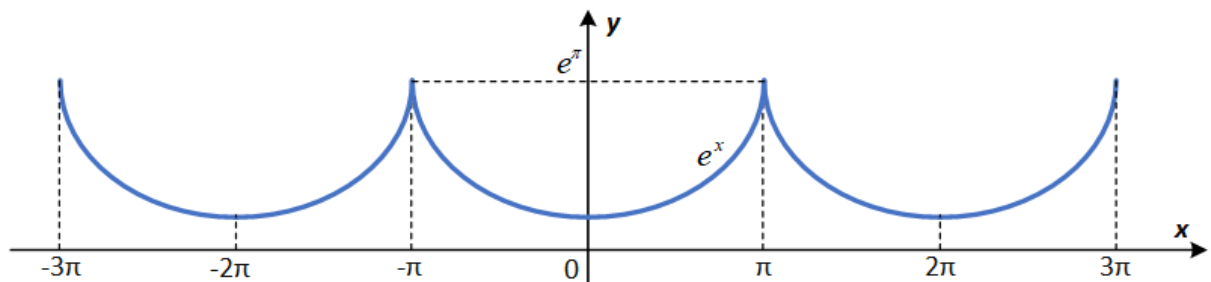


Рисунок 1.2 – Вихідний сигнал (парна функція)

Якщо продовжити функцію $f(x) = e^x$ непарним чином (рис. 1.3), то коефіцієнти ряду Фур'є: $a_0 = 0$; $a_n = 0$.

Роздивимось наступні спрощення:

$$b_n = \frac{2}{\pi} \int_0^{\pi} e^x \sin nx \, dx$$

$$\int_0^{\pi} e^x \sin nx \, dx =$$

$$\left\{ \begin{array}{l} u = e^x \quad dv = \sin nx \, dx \\ du = e^x \, dx \quad v = -\frac{1}{n} \cos nx \end{array} \right\} = -e^x \cdot \frac{1}{n} \cos nx + \frac{1}{n} \cdot \int_0^{\pi} e^x \cos nx \, dx =$$

$$\left\{ \begin{array}{l} u = e^x \quad dv = \cos nx \, dx \\ du = e^x \, dx \quad v = \frac{1}{n} \sin nx \end{array} \right\} = -e^x \cdot \frac{1}{n} \cos nx + \frac{1}{n} \cdot \left[e^x \cdot \frac{1}{n} \sin nx - \frac{1}{n} \int_0^{\pi} e^x \sin nx \, dx \right] =$$

$$= -\frac{1}{n} e^x \cos nx + \frac{1}{n^2} e^x \sin nx - \frac{1}{n^2} \cdot \int_0^{\pi} e^x \sin nx \, dx$$

$$\int_0^{\pi} e^x \sin nx \, dx = \frac{n^2}{n^2 + 1} \left(-\frac{1}{n} e^x \cos nx + \frac{1}{n^2} e^x \sin nx \right);$$

$$b_n = \frac{2}{\pi} \int_0^{\pi} e^x \sin nx \, dx = \frac{2}{\pi} \left[\frac{n^2}{n^2 + 1} \left(-\frac{1}{n} e^x \cos nx + \frac{1}{n^2} e^x \sin nx \right) \right]_0^{\pi} =$$

$$= \frac{2}{\pi} \frac{n^2}{n^2 + 1} \left(-\frac{1}{n} e^{\pi} \cos n\pi + \frac{1}{n^2} e^{\pi} \sin n\pi - \left(-\frac{1}{n} e^0 \cos n0 + \frac{1}{n^2} e^0 \sin n0 \right) \right) =$$

$$= \frac{2}{\pi} \frac{n^2}{n^2 + 1} \left(-\frac{1}{n} e^{\pi} (-1)^n + \frac{1}{n} \right) = \frac{2n}{\pi(n^2 + 1)} (1 - e^{\pi} (-1)^n)$$

В результаті:

$$f(x) = e^x = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) = \sum_{n=1}^{\infty} \left[\frac{2n}{\pi(n^2 + 1)} (1 - e^{\pi} (-1)^n) \sin nx \right]$$

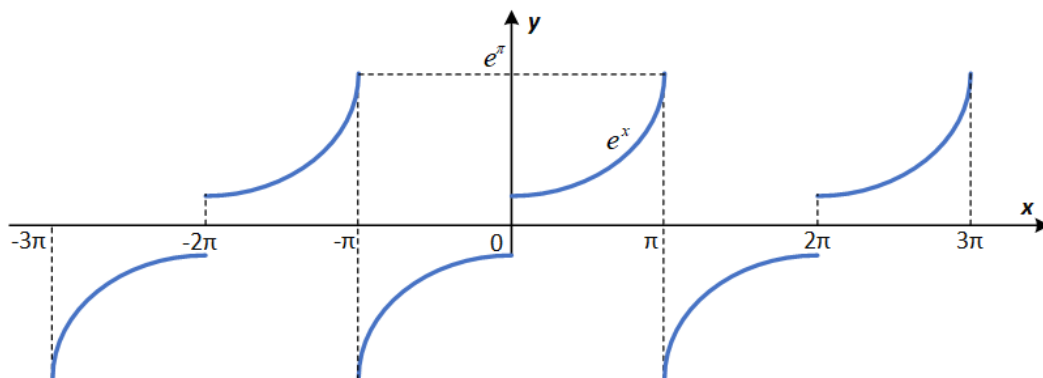
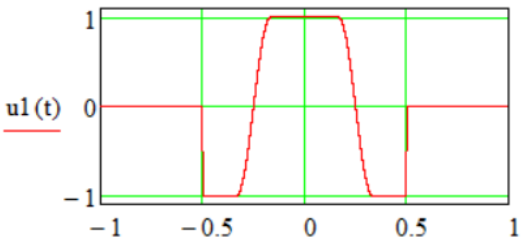
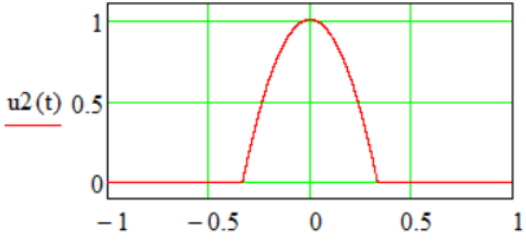
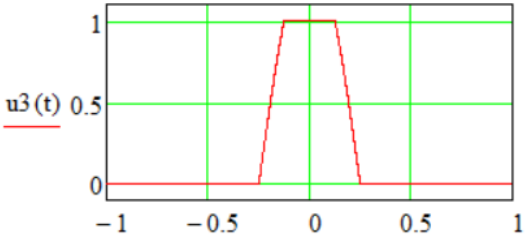
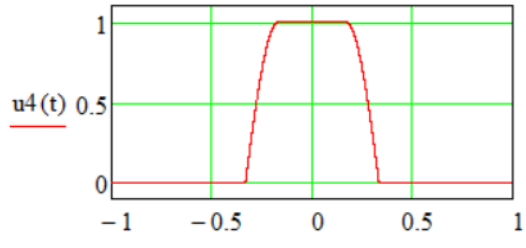
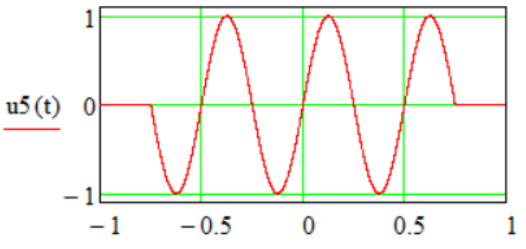


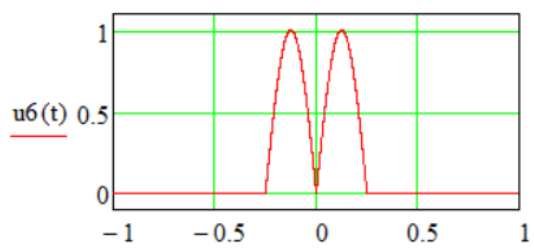
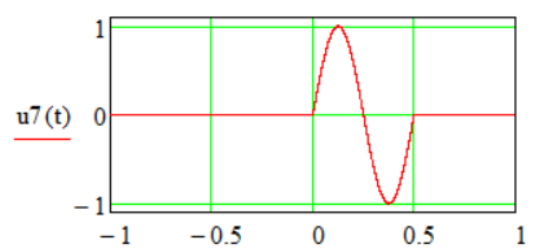
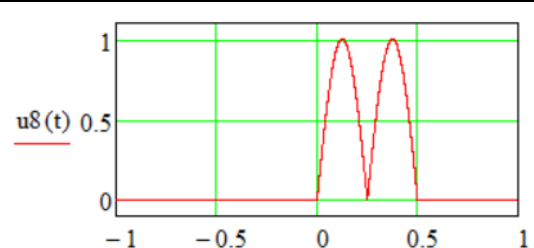
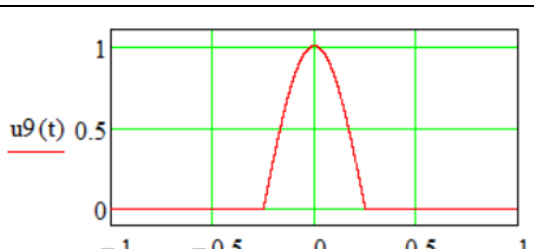
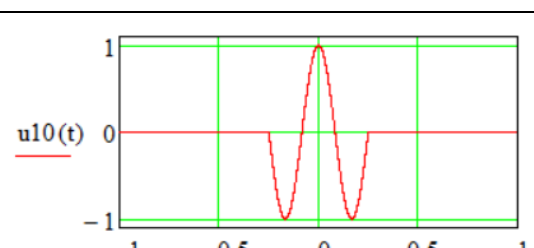
Рисунок 1.3 – Вихідний сигнал (непарна функція)

Індивідуальне завдання

Визначити коефіцієнти ряду Фур'є теоретично і за допомогою Mathcad.
Визначити комплексну форму Фур'є. Визначити амплітудно-частотний спектр і фазо-частотний спектр.

Таблиця 2.1 – Індивідуальне завдання до першої теми

варіант	функція	графік
1	$u_1(t) := \begin{cases} 1 & \text{if } t \leq \frac{1}{6} \\ -\cos(6 \cdot \pi \cdot t) & \text{if } \frac{1}{6} \leq t \leq \frac{1}{3} \\ (-1) & \text{if } \frac{1}{3} \leq t \leq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$	
2	$u_2(t) := \begin{cases} 1 - 9 \cdot t^2 & \text{if } t \leq \frac{1}{3} \\ 0 & \text{otherwise} \end{cases}$	
3	$u_3(t) := \begin{cases} 1 & \text{if } t \leq \frac{1}{8} \\ \sqrt{2} \cdot \cos(2 \cdot \pi \cdot t) & \text{if } \frac{1}{8} \leq t \leq \frac{1}{4} \\ 0 & \text{otherwise} \end{cases}$	
4	$u_4(t) := \begin{cases} 1 & \text{if } t \leq \frac{1}{6} \\ \sin(3 \cdot \pi \cdot t) & \text{if } \frac{1}{6} \leq t \leq \frac{1}{3} \\ 0 & \text{otherwise} \end{cases}$	
5	$u_5(t) := \begin{cases} \sin(4 \cdot \pi \cdot t) & \text{if } 0 \leq t \leq \frac{3}{4} \\ 0 & \text{otherwise} \end{cases}$	

варіант	функція	графік
6	$u_6(t) := \begin{cases} \sin(4 \cdot \pi \cdot t) & \text{if } 0 \leq t \leq \frac{1}{4} \\ 0 & \text{otherwise} \end{cases}$	
7	$u_7(t) := \begin{cases} \sin(4 \cdot \pi \cdot t) & \text{if } 0 \leq t \leq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$	
8	$u_8(t) := \begin{cases} \sin(4 \cdot \pi \cdot t) & \text{if } 0 \leq t \leq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$	
9	$u_9(t) := \begin{cases} \cos(2 \cdot \pi \cdot t) & \text{if } t \leq \frac{1}{4} \\ 0 & \text{otherwise} \end{cases}$	
10	$u_{10}(t) := \begin{cases} \cos(6 \cdot \pi \cdot t) & \text{if } t \leq \frac{1}{4} \\ 0 & \text{otherwise} \end{cases}$	

Питання для самоперевірки

1. Принцип перетворення Фур'є.
2. Амплітудно-частотний спектр.
3. Фазово-частотний спектр.

Література

[1, 4-7].

ЛАБОРАТОРНА РОБОТА № 2

ДОСЛІДЖЕННЯ ЕНТРОПІЇ

Мета роботи: дослідити кількісну оцінку інформації.

Теоретична частина

Якщо аналізувати інформацію на синтаксичному рівні, то це означає оцінювати зменшення невизначеності про об'єкт, який досліджується:

$$I = H_1 - H_2,$$

де H_1 – інформація до дослідження; H_2 – інформація після дослідження.

Логарифмічна міра Хартлі застосовується для визначення інформації, коли ймовірність виникнення символів однакова $p_1 = p_2 = \dots = p_i = \dots = p_m$:

$$I = \log_a H = \log_a m^n = n \cdot \log_a m.$$

де a – основа логарифму (визначає одиниці вимірювання), n – кількість символів в повідомленні; m – кінцева кількість символів, які мають рівну ймовірність появи:

$$p_i = \frac{1}{m}$$

Для символів, які мають неоднакові ймовірності $p_1 \neq p_2 \neq \dots \neq p_i \neq \dots \neq p_m$ використовується міра Шеннона:

$$I = -n \sum_{i=1}^m p_i \cdot \log_a p_i$$

Ентропія H – логарифмічна міра невизначеності стану джерела повідомлень, або іншими словами, середня кількість інформації на один символ:

$$H = - \sum_{i=1}^m p_i \cdot \log p_i$$

Розрізняють ентропію об'єднання для сумісної появи статистично залежних повідомлень:

$$H(A, B) = H(A) + H(B / A) = H(B) + H(A / B).$$

Імовірність сумісної появи символів

$$p(A, B) = p(B) p(A / B) = p(A) p(B / A).$$

За відсутністю статистичної залежності між подіями A та B умовні імовірності перетворюються на безумовні: $H(A, B) = H(A) + H(B)$.

При повній статистичній залежності між подіями A та B умовні ентропії рівні нулю: $H(A, B) = H(A) = H(B)$.

Умовна ентропія розраховується через умовні імовірності появи одних подій відносно інших:

$$H(b_j / a_i) = - \sum_j p(b_j / a_i) \log_2 p(b_j / a_i);$$

$$H(a_i / b_j) = - \sum_i p(a_i / b_j) \log_2 p(a_i / b_j).$$

Загальна умовна ентропія:

$$H(B / A) = - \sum_j p(a_i) H(b_j / a_i) = - \sum_i \sum_j p(a_i) p(b_j / a_i) \log_2 p(b_j / a_i).$$

У випадку нерівноймовірних та взаємопов'язаних символів:

$$H(B / A) = - \sum_i \sum_j p(a_i, b_j) \log_2 p(b_j / a_i).$$

Приклади визначення ентропії повідомлень наведені в практичній частині.

Практична частина

Приклад 2.1. (Безумовні ймовірності).

Для заданої імовірності символів $p(a_i, b_j)$ (табл. 2.1) знайти ймовірність об'єднання подій A і B .

Таблиця 2.1 – Вихідні дані для прикладу 2.1

A/B		$b_1 =$	$b_2 =$
		3	6
$a_1 =$	2	0,23	0,10
$a_2 =$	8	0,10	0,31
$a_3 =$	9	0,19	0,07

Безумовні імовірності $p(a_i)$ (табл. 2.2):

$$p(a_1) = p(a_1, b_1) + p(a_1, b_2) = 0,23 + 0,10 = 0,33$$

$$p(a_2) = p(a_2, b_1) + p(a_2, b_2) = 0,10 + 0,31 = 0,41$$

$$p(a_3) = p(a_3, b_1) + p(a_3, b_2) = 0,19 + 0,07 = 0,26$$

Імовірність об'єднання події A:

$$p(A) = \sum_i p(a_i) = p(a_1) + p(a_2) + p(a_3) = 0,33 + 0,41 + 0,26 = 1.$$

Таблиця 2.2 – Результати розрахунків

A/B		$b_1 =$	$b_2 =$		
		3	6		
$a_1 =$	2	0,23	0,10	$p(a_1) =$	0,33
$a_2 =$	8	0,10	0,31	$p(a_2) =$	0,41
$a_3 =$	9	0,19	0,07	$p(a_3) =$	0,26
		$p(b_1) =$	$p(b_2) =$		
		0,52	0,48		

Безумовні імовірності $p(b_j)$ (табл. 2.2):

$$p(b_1) = p(b_1, a_1) + p(b_1, a_2) + p(b_1, a_3) = 0,23 + 0,10 + 0,19 = 0,52$$

$$p(b_2) = p(b_2, a_1) + p(b_2, a_2) + p(b_2, a_3) = 0,10 + 0,31 + 0,07 = 0,48$$

Імовірність об'єднання події B:

$$p(B) = \sum_j p(b_j) = p(b_1) + p(b_2) = 0,52 + 0,48 = 1.$$

Приклад 2.2. (Безумовні й умовні імовірності).

Для заданої матриця символів $p(a_i, b_j)$ Знайти:

- 1) безумовні імовірності $p(a_i)$, $p(b_j)$;
- 2) умовні імовірності $p(a_i / b_j)$, $p(b_j / a_i)$.

Таблиця 2.3 – Вихідні данні для прикладу 2.2

A/B		$b_1 =$	$b_2 =$
		3	6
$a_1 =$	2	0,14	0,12
$a_2 =$	8	0,11	0,33
$a_3 =$	9	0,23	0,07

Безумовні імовірності $p(a_i)$, $p(b_j)$ знаходяться підсумовуванням за рядками і стовпчиками – табл. 2.4.

Таблиця 2.4 – Результати розрахунку безумовних ймовірностей

A/B		$b_1 =$	$b_2 =$		
		3	6		
$a_1 =$	2	0,14	0,12	$p(a_1) =$	0,26
$a_2 =$	8	0,11	0,33	$p(a_2) =$	0,44
$a_3 =$	9	0,23	0,07	$p(a_3) =$	0,30
		$p(b_1) =$	$p(b_2) =$		1,0
		0,48	0,52	1,0	

Умовні імовірності $p(a_i / b_j)$:

$$p(a_i / b_j) = \frac{p(a_i, b_j)}{p(b_j)}.$$

$$\left. \begin{aligned} p(a_1 / b_1) &= \frac{p(a_1, b_1)}{p(b_1)} = \frac{0,14}{0,48} = 0,29; \\ p(a_2 / b_1) &= \frac{p(a_2, b_1)}{p(b_1)} = \frac{0,11}{0,48} = 0,23; \\ p(a_3 / b_1) &= \frac{p(a_3, b_1)}{p(b_1)} = \frac{0,23}{0,48} = 0,48. \end{aligned} \right\} = 1$$

$$\left. \begin{aligned} p(a_1 / b_2) &= \frac{p(a_1, b_2)}{p(b_2)} = \frac{0,12}{0,52} = 0,23; \\ p(a_2 / b_2) &= \frac{p(a_2, b_2)}{p(b_2)} = \frac{0,33}{0,52} = 0,635; \\ p(a_3 / b_2) &= \frac{p(a_3, b_2)}{p(b_2)} = \frac{0,07}{0,52} = 0,135. \end{aligned} \right\} = 1$$

Умовні імовірності $p(b_j / a_i)$:

$$p(b_j / a_i) = \frac{p(b_j, a_i)}{p(a_i)}.$$

$$\left. \begin{aligned} p(b_1 / a_1) &= \frac{p(b_1, a_1)}{p(a_1)} = \frac{0,14}{0,26} = 0,54; \\ p(b_2 / a_1) &= \frac{p(b_2, a_1)}{p(a_1)} = \frac{0,12}{0,26} = 0,46. \end{aligned} \right\} = 1;$$

$$\left. \begin{aligned} p(b_1 / a_2) &= \frac{p(b_1, a_2)}{p(a_2)} = \frac{0,11}{0,44} = 0,25; \\ p(b_2 / a_2) &= \frac{p(b_2, a_2)}{p(a_2)} = \frac{0,33}{0,44} = 0,75. \end{aligned} \right\} = 1;$$

$$\left. \begin{aligned} p(b_1 / a_3) &= \frac{p(b_1, a_3)}{p(a_3)} = \frac{0,23}{0,3} = 0,77; \\ p(b_2 / a_3) &= \frac{p(b_2, a_3)}{p(a_3)} = \frac{0,07}{0,3} = 0,23. \end{aligned} \right\} = 1.$$

Таблиця 2.5 – Результати розрахунку умовних ймовірностей

A/B		$b_1 =$	$b_2 =$				
		3	6				
$a_1 =$	2	0,14	0,12	$p(a_1) =$	0,26	0,29	0,23
$a_2 =$	8	0,11	0,33	$p(a_2) =$	0,44	0,23	0,635
$a_3 =$	9	0,23	0,07	$p(a_3) =$	0,30	0,48	0,135
		$p(b_1) =$	$p(b_2) =$		1,0	1,0	1,0
		0,48	0,52	1,0			
$p(a_1 / b_j)$		0,54	0,46	1,0			
$p(a_2 / b_j)$		0,25	0,75	1,0			
$p(a_3 / b_j)$		0,77	0,23	1,0			

Приклад 2.3. (Ентропії об'єднання).

Визначити ентропії $H(A)$, $H(B)$, $H(A/B)$, $H(B/A)$ для матриці ймовірностей об'єднання.

Таблиця 2.6 – Вихідні данні для прикладу 2.3

A/B		$b_1 =$	$b_2 =$	$b_3 =$
		...	...	...
$a_1 =$	...	0,4	0,1	0
$a_2 =$	...	0	0,2	0,1
$a_3 =$	...	0	0	0,2

Безумовна ймовірність (табл. 2.7):

$$\left. \begin{aligned} p(a_1) &= p(a_1, b_1) + p(a_1, b_2) + p(a_1, b_3) = 0,4 + 0,1 + 0 = 0,5 \\ p(a_2) &= p(a_2, b_1) + p(a_2, b_2) + p(a_2, b_3) = 0 + 0,2 + 0,1 = 0,3 \\ p(a_3) &= p(a_3, b_1) + p(a_3, b_2) + p(a_3, b_3) = 0 + 0 + 0,2 = 0,2 \end{aligned} \right\} = 1$$

$$\left. \begin{aligned} p(b_1) &= p(b_1, a_1) + p(b_1, a_2) + p(b_1, a_3) = 0,4 + 0 + 0 = 0,4 \\ p(b_2) &= p(b_2, a_1) + p(b_2, a_2) + p(b_2, a_3) = 0,1 + 0,2 + 0 = 0,3 \\ p(b_3) &= p(b_3, a_1) + p(b_3, a_2) + p(b_3, a_3) = 0 + 0,1 + 0,2 = 0,3 \end{aligned} \right\} = 1$$

Таблиця 2.7 – Результати розрахунку безумовної ймовірності

A/B		$b_1 =$	$b_2 =$	$b_3 =$		
$a_1 =$		0,4	0,1	0	$p(a_1) =$	0,5
$a_2 =$		0	0,2	0,1	$p(a_2) =$	0,3
$a_3 =$		0	0	0,2	$p(a_3) =$	0,2
		$p(b_1) =$	$p(b_2) =$	$p(b_3) =$		1
		0,4	0,3	0,3	1	

Ентропія джерела:

$$H(A) = -[p(a_1) \cdot \log_2 p(a_1) + p(a_2) \cdot \log_2 p(a_2) + p(a_3) \cdot \log_2 p(a_3)] =$$

$$-[0,5 \cdot \log_2 0,5 + 0,3 \cdot \log_2 0,3 + 0,2 \cdot \log_2 0,2] = 1,485 \text{ біт / символ}$$

Ентропія приймача:

$$H(B) = -[p(b_1) \cdot \log_2 p(b_1) + p(b_2) \cdot \log_2 p(b_2) + p(b_3) \cdot \log_2 p(b_3)] =$$

$$-[0,4 \cdot \log_2 0,4 + 0,3 \cdot \log_2 0,3 + 0,3 \cdot \log_2 0,3] = 1,571 \text{ біт / символ}$$

Умовні імовірності (табл. 2.8):

$$\left. \begin{aligned} p(b_1 / a_1) &= \frac{p(b_1, a_1)}{p(a_1)} = \frac{0,4}{0,5} = 0,8 \\ p(b_2 / a_1) &= \frac{p(b_2, a_1)}{p(a_1)} = \frac{0,1}{0,5} = 0,2 \\ p(b_3 / a_1) &= \frac{p(b_3, a_1)}{p(a_1)} = \frac{0}{0,5} = 0 \end{aligned} \right\} = 1$$

Таблиця 2.8 – Результати розрахунку умовної ймовірності

A/B		$b_1 =$	$b_2 =$	$b_3 =$						
		..	..	..			$p(b_1 / a_i)$	$p(b_2 / a_i)$	$p(b_3 / a_i)$	
$a_1 =$	..	0,4	0,1	0	$p(a_1) =$	0,5	0,8	0,2	0	1
$a_2 =$	..	0	0,2	0,1	$p(a_2) =$	0,3	0	0,67	0,33	1
$a_3 =$	..	0	0	0,2	$p(a_3) =$	0,2	0	0	1	1
$p(a_1 / b_j) =$ $p(a_2 / b_j) =$ $p(a_3 / b_j)$		$p(b_1) =$	$p(b_2) =$	$p(b_3) =$		1				
		0,4	0,3	0,3	1					
		1	0,33	0						
		0	0,67	0,33						
		0	0	0,67						
		1	1	1						

$$\left. \begin{aligned} p(b_1 / a_2) &= \frac{p(b_1, a_2)}{p(a_2)} = \frac{0}{0,3} = 0 \\ p(b_2 / a_2) &= \frac{p(b_2, a_2)}{p(a_2)} = \frac{0,2}{0,3} = 0,67 \\ p(b_3 / a_2) &= \frac{p(b_3, a_2)}{p(a_2)} = \frac{0,1}{0,3} = 0,33 \end{aligned} \right\} = 1$$

$$\left. \begin{aligned} p(b_1 / a_3) &= \frac{p(b_1, a_3)}{p(a_3)} = \frac{0}{0,2} = 0 \\ p(b_2 / a_3) &= \frac{p(b_2, a_3)}{p(a_3)} = \frac{0}{0,2} = 0 \\ p(b_3 / a_3) &= \frac{p(b_3, a_3)}{p(a_3)} = \frac{0,2}{0,2} = 1 \end{aligned} \right\} = 1$$

Умовна ентропія:

$$\begin{aligned} H(B / A) &= -\sum_j p(a_i) H(b_j / a_i) = -\sum_i \sum_j p(a_i) p(b_j / a_i) \log_2 p(b_j / a_i) = \\ &= -p(a_1) [p(b_1 / a_1) \cdot \log_2 (b_1 / a_1) + p(b_2 / a_1) \cdot \log_2 (b_2 / a_1) + p(b_3 / a_1) \cdot \log_2 (b_3 / a_1)] - \\ &- p(a_2) [p(b_1 / a_2) \cdot \log_2 (b_1 / a_2) + p(b_2 / a_2) \cdot \log_2 (b_2 / a_2) + p(b_3 / a_2) \cdot \log_2 (b_3 / a_2)] - \\ &- p(a_3) [p(b_1 / a_3) \cdot \log_2 (b_1 / a_3) + p(b_2 / a_3) \cdot \log_2 (b_2 / a_3) + p(b_3 / a_3) \cdot \log_2 (b_3 / a_3)] = \\ &= -0,5 \cdot \left[0,8 \cdot \log_2 (0,8) + 0,2 \cdot \log_2 0,2 + \underbrace{0 \cdot \log_2 0}_{0 \cdot \infty} \right] - \\ &- 0,3 \cdot \left[\underbrace{0 \cdot \log_2 0}_{0 \cdot \infty} + 0,67 \cdot \log_2 0,67 + 0,33 \cdot \log_2 0,33 \right] - \\ &- 0,2 \cdot \left[\underbrace{0 \cdot \log_2 0}_{0 \cdot \infty} + \underbrace{0 \cdot \log_2 0}_{0 \cdot \infty} + \underbrace{1 \cdot \log_2 1}_{1 \cdot 0} \right] = 0,635 \text{ біт / символ} \end{aligned}$$

Умовні імовірності (табл. 2.8):

$$\left. \begin{aligned} p(a_1 / b_1) &= \frac{p(a_1, b_1)}{p(b_1)} = \frac{0,4}{0,4} = 1 \\ p(a_2 / b_1) &= \frac{p(a_2, b_1)}{p(b_1)} = \frac{0}{0,4} = 0 \\ p(a_3 / b_1) &= \frac{p(a_3, b_1)}{p(b_1)} = \frac{0}{0,4} = 0 \end{aligned} \right\} = 1$$

$$\left. \begin{aligned} p(a_1 / b_2) &= \frac{p(a_1, b_2)}{p(b_2)} = \frac{0,1}{0,3} = 0,33 \\ p(a_2 / b_2) &= \frac{p(a_2, b_2)}{p(b_2)} = \frac{0,2}{0,3} = 0,67 \\ p(a_3 / b_2) &= \frac{p(a_3, b_2)}{p(b_2)} = \frac{0}{0,3} = 0 \end{aligned} \right\} = 1$$

$$\left. \begin{aligned} p(a_1 / b_3) &= \frac{p(a_1, b_3)}{p(b_3)} = \frac{0}{0,3} = 0 \\ p(a_2 / b_3) &= \frac{p(a_2, b_3)}{p(b_3)} = \frac{0,1}{0,3} = 0,33 \\ p(a_3 / b_3) &= \frac{p(a_3, b_3)}{p(b_3)} = \frac{0,2}{0,3} = 0,67 \end{aligned} \right\} = 1$$

$$\begin{aligned} H(A/B) &= -\sum_j p(b_j) H(a_i / b_j) = -\sum_i \sum_j p(b_j) p(a_i / b_j) \log_2 p(a_i / b_j) = \\ &= -p(b_1) [p(a_1 / b_1) \cdot \log_2 (a_1 / b_1) + p(a_2 / b_1) \cdot \log_2 (a_2 / b_1) + p(a_3 / b_1) \cdot \log_2 (a_3 / b_1)] - \\ &\quad - p(b_2) [p(a_1 / b_2) \cdot \log_2 (a_1 / b_2) + p(a_2 / b_2) \cdot \log_2 (a_2 / b_2) + p(a_3 / b_2) \cdot \log_2 (a_3 / b_2)] - \\ &\quad - p(b_3) [p(a_1 / b_3) \cdot \log_2 (a_1 / b_3) + p(a_2 / b_3) \cdot \log_2 (a_2 / b_3) + p(a_3 / b_3) \cdot \log_2 (a_3 / b_3)] = \\ &= -0,4 \cdot \left[\underbrace{1 \cdot \log_2 1}_{1:0} + \underbrace{0 \cdot \log_2 0}_{0:\infty} + \underbrace{0 \cdot \log_2 0}_{0:\infty} \right] - \\ &\quad - 0,3 \cdot \left[0,33 \cdot \log_2 (0,33) + 0,67 \cdot \log_2 0,67 + \underbrace{0 \cdot \log_2 0}_{0:\infty} \right] - \\ &\quad - 0,3 \cdot \left[\underbrace{0 \cdot \log_2 0}_{0:\infty} + 0,33 \cdot \log_2 0,33 + 0,67 \cdot \log_2 0,67 \right] = \\ &= 0,549 \text{ бит / символ} \end{aligned}$$

Питання для самоперевірки

1. Міра Хартлі.
2. Міра Шеннона.
3. Поняття ентропії.

Література

[1, 4-7].

ЛАБОРАТОРНА РОБОТА № 3

НЕПОЗИЦІЙНЕ КОДУВАННЯ. ДОСЛІДЖЕННЯ КОДУ ГРЕЯ

Мета роботи: дослідити теоретичні принципи формування коду Грея.

Теоретична частина

Як добре відоме, система числення відображає правила зображення чисел цифровими символами. Позиційна система числення – значення символу залежить від його позиції в ряді символів. Наприклад, в десятковій системі, 11_{10} – один десяток і одна одиниця. В двійковій системі, 11_2 – це коефіцієнти при двійці у першому і нульовому розрядах: $11_2 = 1 \cdot 2^1 + 1 \cdot 2^0 = 3_{10}$. Непозиційна система числення – положення символу не залежить від його значення. Код Грея є прикладом непозиційного кодування.

3.1. Двійковий код Грея – спосіб переліку всіх 2^n бітових рядків з n біт, таким чином, що на наступному переліку змінюється лише один біт. Зміна детермінована і регулярна.

Генерація кодів Грея (Grey code):

1. Задається послідовність коду: n біт.

2. На першому етапі $k=1$.

3. Задається додаткова послідовність a , розміром $n+1$, яка на першому етапі заповнюється нулями:

$$[a_1^k \quad a_2^k \quad \dots \quad a_i^k \quad \dots \quad a_{n+1}^k] = [0 \quad 0 \quad \dots \quad 0 \quad \dots \quad 0].$$

4. Вводиться додаткова змінна $j=1$.

5. Перші n елементів коду Грея визначаються у реверсивному порядку:

$$G = [g_1 \quad g_2 \quad \dots \quad g_i \quad \dots \quad g_n]$$

$$g_1 = a_n^k; \quad g_2 = a_{n-1}^k; \quad \dots; \quad g_n = a_1^k$$

6. На наступному $(k+1)$ етапі встановлюється: $a_{n+1}^{k+1} = 1 - a_{n+1}^k$.

Якщо $a_{n+1}^{k+1} = 1$, то $j=1$, $a_j^{k+1} = 1 - a_j^k$.

Якщо $a_{n+1}^{k+1} = 0$, то $j = i$, i – номер елементу (першого за порядком) з умовою $a_i^k = 1$, $a_{j+1}^{k+1} = 1 - a_{j+1}^k$.

7. Наступні n елементів коду Грея визначаються так само у реверсивному порядку:

$$G = [g_1 \ g_2 \ \dots \ g_i \ \dots \ g_{n-1}]$$

$$g_1 = a_n^{k+1}; \quad g_2 = a_{n-1}^{k+1}; \quad \dots; \quad g_n = a_1^{k+1}$$

8. Критерій закінчення генерації кодів Грея: $j = n$.

Принцип генерація кодів Грея проілюстровано в прикладі 3.1 практичної частини роботи.

3.2. Код Грея – рефлексний (відбитий) двійковий код. Друга половина значень у коді Грея еквівалентна першій половині, тільки у зворотному порядку, за винятком старшого біта, який просто інвертується.

Алгоритм кодів Грея, як для рефлексного двійкового коду.

1. $G_0 = []$ – пустий рядок.

2. $G_n = 0G_{n-1}, 1G_{n-1}^R$,

де $0G_{n-1}$ – рядки із попередньої послідовності доповнені нулем попереду; $1G_{n-1}^R$ – рядки із попередньої послідовності, взяті у зворотному порядку і доповнені одиницею попереду, n – кількість бітів.

Вище наведений принцип генерація кодів Грея проілюстровано в прикладі 3.2 практичної частини роботи.

3.3. З іншого боку отримати код Грея можна з двійкового коду за співвідношеннями:

$$g_n = a_n; \quad g_{n-1} = a_n \oplus a_{n-1}; \quad \dots \quad g_1 = a_2 \oplus a_1,$$

де $a_n, a_{n-1}, \dots, a_2, a_1$ – розряди двійкового коду; $g_n, g_{n-1}, \dots, g_2, g_1$ – розряди коду Грея; n – кількість розрядів.

Зворотне перетворення:

$$a_n = g_n; \quad a_{n-1} = a_n \oplus g_{n-1}; \quad \dots \quad a_1 = a_2 \oplus g_1.$$

Вище наведений принцип генерація кодів Грея проілюстровано в прикладі 3.3 практичної частини роботи.

Практична частина

Приклад 3.1. (Код Грея).

Побудувати код Грея для значення біт $n = 3$.

На першому етапі $k=1$ задається додаткова послідовність a , розміром $n+1=4$, яка на першому етапі заповнюється нулями:

$$k=1; \quad a = [0 \quad 0 \quad 0 \quad 0].$$

Вводиться додаткова змінна j .

Перші n елементів коду Грея визначаються у реверсивному порядку:

$$g_1 = a_3 = 0; \quad g_2 = a_2 = 0; \quad g_3 = a_1 = 0.$$

$$G_1 = g_1 \quad g_2 \quad g_3$$

$$G_1 = 0 \quad 0 \quad 0$$

На наступному $k=2$ етапі останній елемент додаткової послідовності встановлюється за правилом: $a_{n+1}^{k+1} = 1 - a_{n+1}^k$. Отримуємо: $a_4 = 1 - 0 = 1$.

Так як останній елемент $a_4 = 1$, то $j=1$, $a_j^{k+1} = 1 - a_j^k$. Отримуємо:
 $a_1 = 1 - 0 = 1$.

$$\text{Для } k=2; \quad a = [1 \quad 0 \quad 0 \quad 1].$$

$$g_1 = a_3 = 0; \quad g_2 = a_2 = 0; \quad g_3 = a_1 = 1.$$

$$G_2 = 0 \quad 0 \quad 1$$

Наступний етап $k=3$: $a_4 = 1 - 1 = 0$. Так як $a_4 = 0$, то $j=i=1$, i – номер елемента (першого за порядком) з умовою $a_i^k = 1$, $a_{j+1}^{k+1} = 1 - a_{j+1}^k$. Отримуємо:
 $a_2 = 1 - 0 = 1$.

$$\text{Для } k=3: \quad a = [1 \quad 1 \quad 0 \quad 0].$$

$$g_1 = a_3 = 0; \quad g_2 = a_2 = 1; \quad g_3 = a_1 = 1.$$

$$G_3 = 0 \quad 1 \quad 1$$

Наступний етап $k=4$: $a_4 = 1 - 0 = 1$. Так як $a_4=1$, то $j=1$, $a_1 = 1 - 1 = 0$.

$$a = [0 \ 1 \ 0 \ 1].$$

$$g_1 = a_3 = 0; \ g_2 = a_2 = 1; \ g_3 = a_1 = 0.$$

$$G_4 = 0 \ 1 \ 0$$

Наступний етап $k=5$: $a_4 = 1 - 1 = 0$. Так як $a_4=0$, то $j=i=2$, $a_3 = 1 - 0 = 1$.

$$a = [0 \ 1 \ 1 \ 0].$$

$$g_1 = a_3 = 1; \ g_2 = a_2 = 1; \ g_3 = a_1 = 0.$$

$$G_5 = 1 \ 1 \ 0$$

Наступний етап $k=6$: $a_4 = 1 - 0 = 1$. Так як $a_4=1$, то $j=1$, $a_1 = 1 - 0 = 1$.

$$a = [1 \ 1 \ 1 \ 1].$$

$$g_1 = a_3 = 1; \ g_2 = a_2 = 1; \ g_3 = a_1 = 1.$$

$$G_6 = 1 \ 1 \ 1$$

Наступний етап $k=7$: $a_4 = 1 - 1 = 0$. Так як $a_4=0$, то $j=i=1$, $a_2 = 1 - 1 = 0$.

$$a = [1 \ 0 \ 1 \ 0].$$

$$g_1 = a_3 = 1; \ g_2 = a_2 = 0; \ g_3 = a_1 = 1.$$

$$G_7 = 1 \ 0 \ 1$$

Наступний етап $k=8$: $a_4 = 1 - 0 = 1$. Так як $a_4=1$, то $j=1$, $a_1 = 1 - 1 = 0$.

$$a = [0 \ 0 \ 1 \ 1].$$

$$g_1 = a_3 = 1; \ g_2 = a_2 = 0; \ g_3 = a_1 = 0.$$

$$G_8 = 1 \ 0 \ 0$$

Наступний етап $k=9$: $a_4 = 1 - 1 = 0$. Так як $a_4=0$, то $j=i=3$.

Отримано, що $j = n = 3$. Виконується критерій закінчення генерації кодів

Грея.

Для $n=3$ код Грея:

$$G = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Перелік всіх комбінація за визначенням дорівнює $2^n = 2^3 = 8$.

Приклад 3.2. (рефлексний код Грея).

Побудувати код Грея, як рефлексний код, для значення біт $n = 3$.

Якщо $n=0$, то $G_0 = [\]$.

Якщо $n=1$, то

$$G_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Якщо $n=2$ (рис. 3.1, а), то

$$G_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Якщо $n=3$ (рис. 3.1, б), то

$$G_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

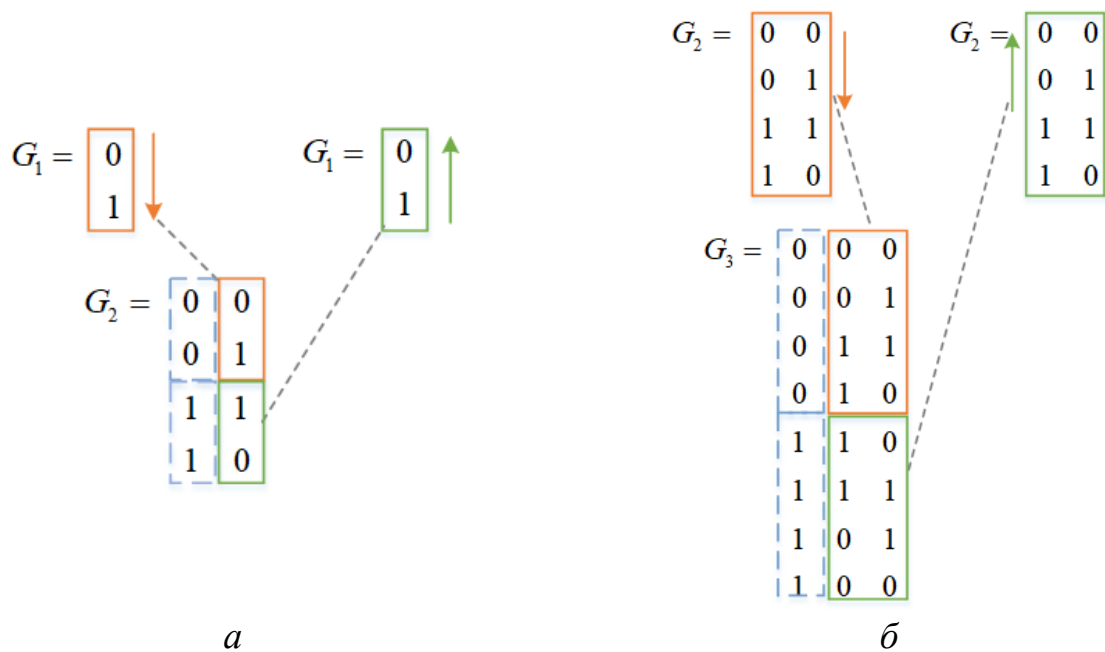


Рисунок 3.1 – Принцип визначення рефлексного коду Грея

Приклад 3.2. (генерація коду Грея з двійкового коду).

Згенерувати код Грея з двійкового коду для чисел від 0 до 7 в десятковому коді.

Спочатку визначимо в двійковій системі, $0_{10}=000_2$, $1_{10}=001_2$, $2_{10}=010_2$, $3_{10}=011_2$, $4_{10}=100_2$, $5_{10}=101_2$, $6_{10}=110_2$, $7_{10}=111_2$.

Для генерації коду Грея:

000_2 : $g_3 = 0$; $g_2 = a_3 \oplus a_2 = 0 \oplus 0 = 0$; $g_1 = a_2 \oplus a_1 = 0 \oplus 0 = 0$; $G: 0 \ 0 \ 0$;

001_2 : $g_3 = 0$; $g_2 = a_3 \oplus a_2 = 0 \oplus 0 = 0$; $g_1 = a_2 \oplus a_1 = 0 \oplus 1 = 1$; $G: 0 \ 0 \ 1$;

010_2 : $g_3 = 0$; $g_2 = a_3 \oplus a_2 = 0 \oplus 1 = 1$; $g_1 = a_2 \oplus a_1 = 1 \oplus 0 = 1$; $G: 0 \ 1 \ 1$;

011_2 : $g_3 = 0$; $g_2 = a_3 \oplus a_2 = 0 \oplus 1 = 1$; $g_1 = a_2 \oplus a_1 = 1 \oplus 1 = 0$; $G: 0 \ 1 \ 0$;

100_2 : $g_3 = 1$; $g_2 = a_3 \oplus a_2 = 1 \oplus 0 = 1$; $g_1 = a_2 \oplus a_1 = 0 \oplus 0 = 0$; $G: 1 \ 1 \ 0$;

101_2 : $g_3 = 1$; $g_2 = a_3 \oplus a_2 = 1 \oplus 0 = 1$; $g_1 = a_2 \oplus a_1 = 0 \oplus 1 = 1$; $G: 1 \ 1 \ 1$;

110_2 : $g_3 = 1$; $g_2 = a_3 \oplus a_2 = 1 \oplus 1 = 0$; $g_1 = a_2 \oplus a_1 = 1 \oplus 0 = 1$; $G: 1 \ 0 \ 1$;

111_2 : $g_3 = 1$; $g_2 = a_3 \oplus a_2 = 1 \oplus 1 = 0$; $g_1 = a_2 \oplus a_1 = 1 \oplus 1 = 0$; $G: 1 \ 0 \ 0$.

Індивідуальне завдання

Визначити код Грея для кількості бітів $n=4$.

Питання для самоперевірки

1. Принцип побудови коду Грея.
2. Принцип побудови реверсивного коду.
3. Принцип побудови коду Грея з двійкового коду.
4. Принцип побудови двійкового коду з коду Грея.

Література

[1, 3].

ЛАБОРАТОРНА РОБОТА № 4

ДОСЛІДЖЕННЯ ЕФЕКТИВНИХ КОДІВ. КОД ШЕННОНА – ФАНО

Мета роботи: дослідити теоретичні принципи формування коду Шеннона – Фано.

Теоретична частина

Задача ефективного кодування – забезпечення мінімального завантаження каналу зв'язку. Принцип ефективного кодування: символи первинного алфавіту, імовірність появи яких є високою, формують короткий код, а символи, які зустрічаються рідко – довгий код. Тому зменшується час використання каналу зв'язку. Код Шеннона – Фано є ефективним кодом.

Алгоритм побудови коду Шеннона-Фано має наступні кроки:

1. Сортують N символів первинного алфавіту за величиною ймовірності їх появи у порядку зменшення.
2. Задається кількість символів q , що використовується у коді.
3. Розділяють N символів первинного алфавіту на q груп за можливістю з рівною ймовірністю виникнення.
4. Символам окремих груп надаються певні символи алфавіту q .
5. Наступним кроком, кожен групу розділяють на q груп за можливістю з рівною ймовірністю виникнення і символам окремих груп надаються певні символи алфавіту q .
6. Розділ на групи і присвоєння символів алфавіту q продовжується доки в групі не залишиться один символ.

Приведений принцип отримання коду Шеннона-Фано проілюстровано в прикладі 4.1 практичної частини роботи.

Практична частина

Приклад 4.1. (Код Шеннона-Фано).

Задано шістнадцять символів ($N=16$) з різною ймовірністю виникнення. Вони приведені у табл. 4.1 у порядку зменшення ймовірності їх виникнення.

Побудувати код Шеннона-Фано з використанням кодового алфавіту $q=2$ з символами 1 і 0.

Таблиця 4.1 – Вхідні дані

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}
0,22	0,10	0,10	0,08	0,07	0,07	0,06	0,06	0,05	0,05	0,04	0,03	0,02	0,02	0,02	0,01

Згідно з алгоритмом розділимо 16 символів на дві групи (табл. 4.2): перша група включає символи первинного алфавіту $x_1 - x_4$ з сумарною ймовірністю 0,5, а друга – символи $x_5 - x_{16}$ з сумарною ймовірністю 0,5. У перший розряд символів першої групи ($x_1 - x_4$) вписується символ «0» алфавіту q , а у перший розряд символів другої групи ($x_5 - x_{16}$) – символ «1».

Наступним кроком групу символів $x_1 - x_4$ розділяємо на дві групи: перша – символ x_1 з ймовірністю 0,22 (у другий розряд вписується символ «0»); друга – символи $x_2 - x_4$ з сумарною ймовірністю 0,28 (у другий розряд вписується символ «1»). Отримали, що для символу x_1 код Шеннона – Фано: 00.

Групу символів $x_2 - x_4$ розділяємо на дві групи: перша – символ x_2 з ймовірністю 0,1 (у третій розряд вписується символ «0»); друга – символи $x_3 - x_4$ з сумарною ймовірністю 0,18 (у третій розряд вписується символ «1»). Для символу x_2 код Шеннона – Фано: 010.

Групу символів $x_3 - x_4$ розділяємо на дві групи: перша – символ x_3 з ймовірністю 0,1 (у четвертий розряд вписується символ «0»); друга – символ x_4 з ймовірністю 0,08 (у четвертий розряд вписується символ «1»). Для символу x_3 код Шеннона – Фано: 0110; для символу x_4 : 0111.

Групу символів $x_5 - x_{16}$ розділяємо на дві групи: перша – символи $x_5 - x_8$ з сумарною ймовірністю 0,26 (у другий розряд вписується символ «0»); друга – символи $x_9 - x_{16}$ з сумарною ймовірністю 0,24 (у другий розряд вписується символ «1»).

Групу символів $x_5 - x_8$ розділяємо на дві групи: перша – символи $x_5 - x_6$ з сумарною ймовірністю 0,14 (у третій розряд вписується символ «0»); друга – символи $x_7 - x_8$ з сумарною ймовірністю 0,12 (у третій розряд вписується символ «1»).

Групу символів $x_5 - x_6$ розділяємо на дві групи: перша – символ x_5 з ймовірністю 0,07 (у четвертий розряд вписується символ «0»); друга – символ x_6 з ймовірністю 0,07 (у четвертий розряд вписується символ «1»). Для символу x_5 код Шеннона – Фано: 1000; для символу x_6 : 1001.

Групу символів $x_7 - x_8$ розділяємо на дві групи: перша – символ x_7 з ймовірністю 0,06 (у четвертий розряд вписується символ «0»); друга – символ x_8 з ймовірністю 0,06 (у четвертий розряд вписується символ «1»). Для символу x_7 код Шеннона – Фано: 1010; для символу x_8 : 1011.

Групу символів $x_9 - x_{16}$ розділяємо на дві групи: перша – символи $x_9 - x_{10}$ з сумарною ймовірністю 0,1 (у третій розряд вписується символ «0»); друга – символи $x_{11} - x_{16}$ з сумарною ймовірністю 0,14 (у третій розряд вписується символ «1»).

Групу символів $x_9 - x_{10}$ розділяємо на дві групи: перша – символ x_9 з ймовірністю 0,05 (у четвертий розряд вписується символ «0»); друга – символ x_{10} з ймовірністю 0,05 (у четвертий розряд вписується символ «1»). Для символу x_9 код Шеннона – Фано: 1100; для символу x_{10} : 1101.

Групу символів $x_{11} - x_{16}$ розділяємо на дві групи: перша – символи $x_{11} - x_{12}$ з сумарною ймовірністю 0,07 (у четвертий розряд вписується символ «0»); друга – символи $x_{13} - x_{16}$ з сумарною ймовірністю 0,07 (у четвертий розряд вписується символ «1»).

Групу символів $x_{11} - x_{12}$ розділяємо на дві групи: перша – символ x_{11} з ймовірністю 0,04 (у п'ятий розряд вписується символ «0»); друга – символ x_{12} з ймовірністю 0,03 (у п'ятий розряд вписується символ «1»). Для символу x_{11} код Шеннона – Фано: 11100; для символу x_{12} : 11101.

Групу символів $x_{13} - x_{16}$ розділяємо на дві групи: перша – символи $x_{13} - x_{14}$ з сумарною ймовірністю 0,04 (у п'ятий розряд вписується символ «0»);

друга – символи $x_{15} - x_{16}$ з сумарною ймовірністю 0,03 (у п'ятий розряд вписується символ «1»).

Групу символів $x_{13} - x_{14}$ розділяємо на дві групи: перша – символ x_{13} з ймовірністю 0,02 (у шостий розряд вписується символ «0»); друга – символ x_{14} з ймовірністю 0,02 (у шостий розряд вписується символ «1»). Для символу x_{13} код Шеннона – Фано: 111100; для символу x_{14} : 111101.

Групу символів $x_{15} - x_{16}$ розділяємо на дві групи: перша – символ x_{15} з ймовірністю 0,02 (у шостий розряд вписується символ «0»); друга – символ x_{16} з ймовірністю 0,01 (у шостий розряд вписується символ «1»). Для символу x_{15} код Шеннона – Фано: 111110; для символу x_{16} : 111111.

Розділ на групи, присвоєння символів алфавіту q показано на дереві графу коду Шеннона-Фано – рис. 4.1.

Таблиця 4.2 – Визначення коду Шеннона-Фано для прикладу 4.1

Символ	Імовірність	Розбиття на групи						Код Шеннона-Фано
		1 ітерація	2 ітерація	3 ітерація	4 ітерація	5 ітерація	6 ітерація	
x_1	0,22	0	0	-	-	-	-	00
x_2	0,10	0	1	0	-	-	-	010
x_3	0,10	0	1	1	0	-	-	0110
x_4	0,08	0	1	1	1	-	-	0111
x_5	0,07	1	0	0	0	-	-	1000
x_6	0,07	1	0	0	1	-	-	1001
x_7	0,06	1	0	1	0	-	-	1010
x_8	0,06	1	0	1	1	-	-	1011
x_9	0,05	1	1	0	0	-	-	1100
x_{10}	0,05	1	1	0	1	-	-	1101
x_{11}	0,04	1	1	1	0	0	-	11100
x_{12}	0,03	1	1	1	0	1	-	11101
x_{13}	0,02	1	1	1	1	0	0	111100
x_{14}	0,02	1	1	1	1	0	1	111101
x_{15}	0,02	1	1	1	1	1	0	111110
x_{16}	0,01	1	1	1	1	1	1	111111

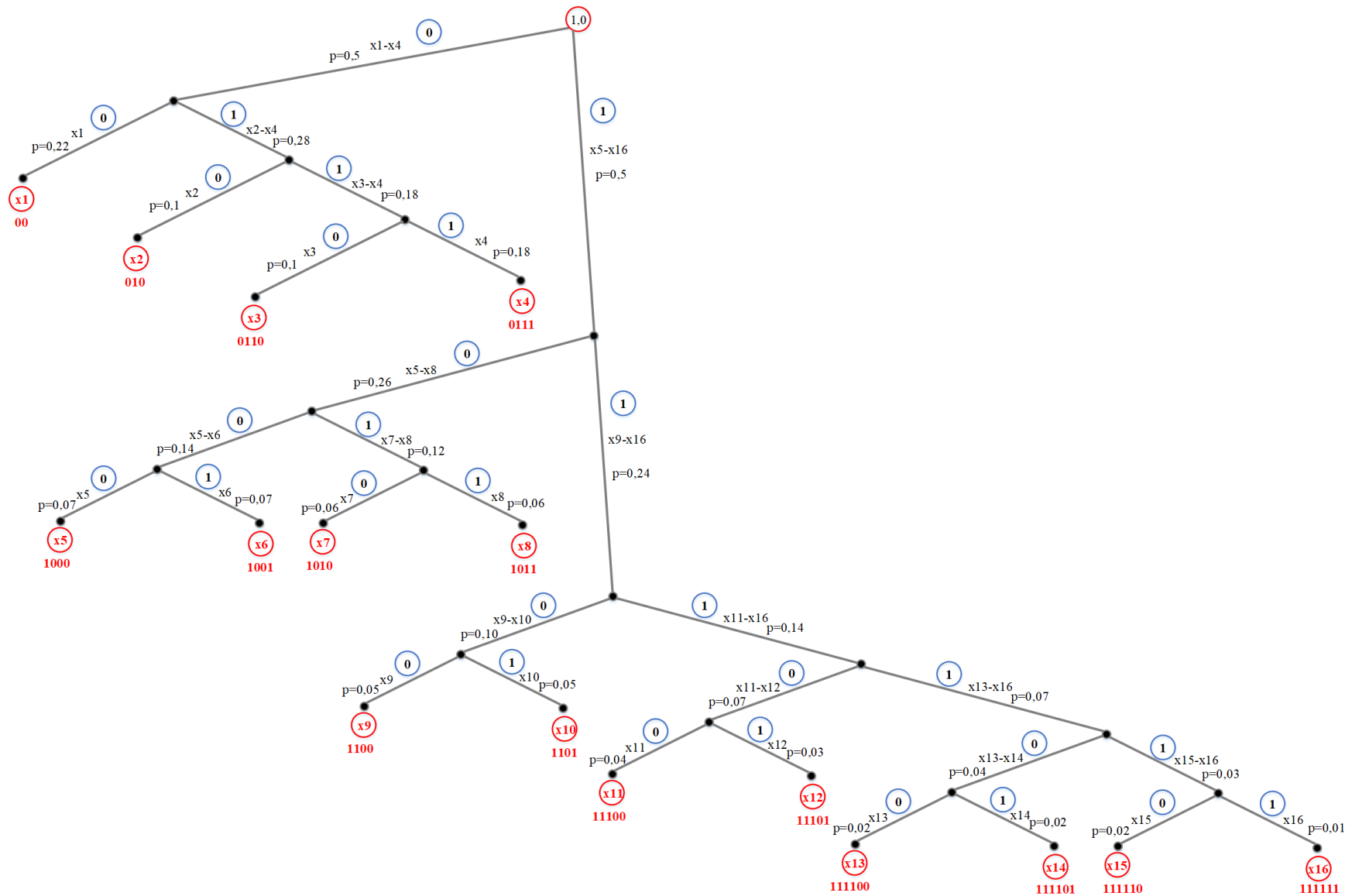


Рисунок 4.1 – Дерево графа коду Шеннона-Фано

Індивідуальне завдання

Інформаційні символи (табл. 4.3) закодувати кодом Шеннона – Фано відповідно до варіанту.

Таблиця 4.3 – Вихідні дані до індивідуального завдання

№ ва- ріанту	x ₁	x ₂	x ₃	x ₄	x ₅	x ₆	x ₇	x ₈
1	0,33	0,21	0,11	0,09	0,08	0,07	0,06	0,05
2	0,28	0,19	0,17	0,13	0,09	0,05	0,05	0,04
3	0,34	0,22	0,14	0,1	0,08	0,05	0,04	0,03
4	0,27	0,2	0,16	0,15	0,08	0,07	0,05	0,02
5	0,32	0,23	0,15	0,09	0,08	0,06	0,06	0,01
6	0,29	0,18	0,17	0,12	0,09	0,08	0,02	0,05
7	0,31	0,24	0,15	0,11	0,08	0,04	0,03	0,04
8	0,26	0,17	0,17	0,16	0,11	0,06	0,04	0,03
9	0,35	0,25	0,1	0,1	0,07	0,06	0,05	0,02
10	0,25	0,16	0,14	0,13	0,12	0,09	0,07	0,04

Питання для самоперевірки

1. Принцип побудови коду Шеннона-Фано.
2. Принцип побудови графу кодового дерева Шеннона-Фано.

Література

[1, 3].

ЛАБОРАТОРНА РОБОТА № 5

ДОСЛІДЖЕННЯ ЕФЕКТИВНИХ КОДІВ. КОД ХАФФМЕНА

Мета роботи: дослідити теоретичні принципи формування коду Хаффмена.

Теоретична частина

5.1. Задача ефективного кодування – забезпечення мінімального завантаження каналу зв'язку. Принцип ефективного кодування: символи первинного алфавіту, імовірність появи яких є високою, формують короткий код, а символи, які зустрічаються рідко, – довгий код. Це дозволяє зменшити час використання каналу зв'язку. Прикладом ефективного коду є код Хаффмена.

Алгоритм побудови коду Хаффмена:

1. Сортують N символів первинного алфавіту за величиною ймовірності їх появи у порядку зменшення.
2. Задається кількість символів q , що використовується у коді.
3. Символи з найменшою ймовірністю, кількістю N_0 ($2 \leq N_0 \leq q$), об'єднуються в новий символ, ймовірність появи якого дорівнює сумі ймовірностей символів, які входять до нього. Символам, що об'єднуються, присвоюється значення символів кодового алфавіту q .
4. Отриману нову множину символів кількістю $N - N_0 + 1$ знову сортують у порядку зменшення ймовірностей їх появи.
5. Починаючи з 3 пункту алгоритм ітераційно повторюється до тих пір, доки не отримуємо символ з ймовірністю 1.
6. Будується кодове дерево. Верхня точка (корінь дерева) має ймовірність $p=1$. Від неї відходять гілки, які відповідають кодовому алфавіту q і мають задану величину ймовірності p_i . Таке розгалуження виконується до тих пір, доки не буде сформований останній символ послідовності N (вершини дерева).

7. Код символу – це послідовність символів алфавіту q на шляху від кореня до вершини дерева, яка відповідає цьому символу.

Приведений принцип отримання коду Хаффмена проілюстровано в прикладі 5.1 і прикладі 5.2 практичної частини роботи.

5.2. Описаний вище ручний спосіб отримання коду Хаффмена є простим та наочним, але погано формалізованим та неефективним з технічної точки зору для реалізації в електронних системах обробки інформації.

Роздивимось алгоритм побудови матриці ідентифікації за ієрархічним деревом коду Хаффмена:

1. Для ребер дерева коду Хаффмена вказуються значення ваги, яка приймає значення кодового алфавіту q : 1 або 0. Значення ймовірностей p_i для ребер дерева не вказуються.

2. Нумеруються всі вершини дерева коду Хаффмена. Нумерація починається з кінцевих вершин ієрархічного дерева коду Хаффмена: символам первинного алфавіту із номером i , де $i=1 \div N$, відповідають вершини з номером i . Внутрішні вершини нумеруються наступними: нумерація починається з $N+1$, де N – кількість символів, які кодуються. Коли всі K внутрішніх вершин дерева будуть пронумеровані, кореню дерева надається останній номер, якій буде складати $n=N+K+1$.

3. Нумеруються всі ребра дерева коду Хаффмена. Ребра дерева кількістю m нумерують від найбільш віддалених зовнішніх вершин до кореня дерева, тобто знизу до гори.

4. Кореню дерева буде відповідати максимальне значення номера вершини n і максимальні значення номерів ребер $m, m-1, \dots$, які до нього підходять.

5. Формується прямокутна матриця ідентифікації I для ієрархічного дерева коду Хаффмена із кількістю стовпчиків n та кількістю рядків m , де n – кількість вершин у дереві, а m – кількість ребер.

6. Якщо існує зв'язок між вершинами із номерами i та j через ребро із номером l , тоді у рядку l у стовпчиках i та j записується значення ваги ребра,

«0» або «1». Якщо ребро l безпосередньо не пов'язано із вершиною k , то у стовпчик k рядка l ставиться значення «-1».

За сформованою матрицею ідентифікації I можна знайти розряди кодів Хаффмена для символів $x_1 - x_N$ за наступним ітераційним алгоритмом.

1. В матриці ідентифікації I у стовпчику i визначається елемент матриці, який не дорівнює «-1» для рядку l . Ця операція проводиться для всіх значень $i \leq N$.

2. Отримане значення знайденого елемента матриці $I(l, i)$ вписується у перший розряд коду символу x_i .

3. У визначеному рядку l , для якого $I(l, i) \neq \text{«-1»}$, визначається інший елемент матриці ідентифікації із номером стовбця k , для якого $I(l, i) = I(l, k)$, $k > N$. Такий елемент буде лише один (всі ребра пов'язані лише із двома вершинами дерева).

4. У стовбцю k визначається інший елемент матриці ідентифікації $I(h, k)$, який не дорівнює «-1».

5. У рядку h шукаємо інший елемент матриці ідентифікації $I(h, r)$ такий, що $I(h, k) = I(h, r)$ за умови $r > N$.

6. Якщо $r > k$, вважаємо, що $k = r$, вписуємо значення знайденого елемента матриці ідентифікації $I(h, k)$ до наступного розряду коду символу x_i та переходимо до пункту 4.

7. Якщо $r < k$, збільшуємо h на одиницю та переходимо до пункту 4.

8. Якщо номер стовбця $k = n$ – завершення формування коду символу x_i та збільшення значення i на 1.

9. Якщо номер початкового стовбця $i < N$ – перехід до пункту 1 алгоритму, у протилежному випадку – закінчення формування коду.

Таким чином визначається єдиний шлях від кінцевої вершини до кореня дерева, а не шляхи між кінцевими вершинами.

Приведений принцип отримання матриці ідентифікації за ієрархічним деревом коду Хаффмена проілюстровано в прикладі 5.3 практичної частини роботи.

5.3. Суттєвий недолік наведеного вище алгоритму створення матриці ідентифікації кодом Хаффмена полягає у тому, що для побудови матриці ідентифікації необхідно знати структуру ієрархічного дерева коду Хаффмена, яка тісно пов'язана із ймовірностями появи символів.

Тобто, алгоритм аналізу матриці ідентифікації пристосований лише для формування кодів символів на основі відомої структури ієрархічного дерева, а методика побудови дерева у цьому алгоритмі не передбачена.

Розглянемо наступний алгоритм для побудови коду Хаффмена. Він буде оснований на формуванні матриці сум M_{Σ} ймовірностей ґрунтуючись на впорядкованому векторі ймовірностей появи елементів.

Перший ряд матриці сум M_{Σ} – це початковий впорядкований вектор ймовірностей появи всіх символів. Наступні рядки матриці сум – це ітерації сумування елементів. Якщо елемент матриці входить в суму на певному етапі ітерації, то на наступних етапах він вважається рівним нулю. Коли сумуються два сусідні елементи із матриці сум, елементу, який стоїть праворуч, присвоюється значення 0, а лівому елементу – значення 1. Останній ряд матриці сум – це одиниця в першій колонці та нулі в інших колонках.

Під час підсумовування ймовірностей елементів змінюється їх порядок. Необхідно контролювати номери елементи, які підсумовувалися. З цією метою вводиться матриця номерів елементів S_{el} , які сумуються. В цій матриці вказуються лише номери двох сусідніх елементів, які сумуються на кожній ітерації, при цьому у першому рядку стоїть елемент із вагою 1, а у другому – елемент із вагою 0.

Матриця номерів елементів, які сумуються, дозволяє відобразити складну структуру ієрархічного дерева коду Хаффмена.

Приведений принцип отримання матриці сум коду Хаффмена і матриці номерів елементів, які сумуються, проілюстровано в прикладі 5.4 практичної частини роботи.

Практична частина

Приклад 5.1. (Код Хаффмена).

Задано шістнадцять символів ($N=16$) з різною імовірністю виникнення. Вони приведені у табл. 5.1 у порядку зменшення ймовірності їх виникнення.

Побудувати код Хаффмена з використанням кодового алфавіту $q=2$ з символами 1 і 0.

Таблиця 5.1 – Вхідні дані

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}
0,22	0,10	0,10	0,08	0,07	0,07	0,06	0,06	0,05	0,05	0,04	0,03	0,02	0,02	0,02	0,01

Згідно з алгоритмом об'єднуємо символи x_{15} і x_{16} . Символам, що об'єднуються, присвоюється значення символу кодового алфавіту q : $x_{15}=1$; $x_{16}=0$. Їх сумарна ймовірність складає 0,03. Ймовірність появи нового об'єданого символу вводиться в додатковий перший стовбець (табл. 5.2), але рядок визначається шляхом сортування у порядку зменшення ймовірностей їх появи. Звідси отримуємо 13 рядок для нового символу $x_{15}-x_{16}$.

Зазначимо, що для другої ітерації ми будемо аналізувати перший додатковий стовбець. Об'єднуємо символи x_{13} і x_{14} , які розташовані в (14 і 15 рядках). Символам присвоюється значення символів кодового алфавіту q : $x_{13}=1$; $x_{14}=0$. Їх сумарна ймовірність складає 0,04. Ця ймовірність нового об'єданого символу $x_{13}-x_{14}$ вводиться в додатковий другий стовбець, у 12 рядок.

Подальший аналіз другого стовбця – це третя ітерація. Об'єднуємо символ x_{12} і новий символ $x_{15}-x_{16}$ першої ітерації. Символам присвоюється значення символів кодового алфавіту q : $x_{12}=1$; $x_{15}-x_{16}=0$. Їх сумарна ймовірність складає 0,06. Новий об'єднаний символ $x_{12}-x_{15}-x_{16}$ вводиться в додатковий третій стовбець, у 9 рядок.

Далі, четверта ітерація – об'єднуємо символ x_{11} і новий символ x_{13} - x_{14} другої ітерації. Присвоюємо значення: $x_{11}=1$; x_{13} - $x_{14}=0$. Їх сумарна ймовірність складає 0,08. Новий об'єднаний символ x_{11} - x_{13} - x_{14} вводиться в додатковий четвертий стовбець у 5 рядок.

П'ята ітерація – об'єднуємо символи x_9 і x_{10} . Присвоюємо значення: $x_9=1$; $x_{10}=0$. Їх сумарна ймовірність складає 0,10. Новий об'єднаний символ x_9 - x_{10} вводиться в додатковий п'ятий стовбець у 4 рядок.

Шоста ітерація – об'єднуємо символ x_8 і новий символ x_{12} - x_{15} - x_{16} третьої ітерації. Присвоюємо значення: $x_8=1$; x_{12} - x_{15} - $x_{16}=0$. Їх сумарна ймовірність складає 0,12. Новий об'єднаний символ x_8 - x_{12} - x_{15} - x_{16} вводиться в додатковий шостий стовбець у 2 рядок.

Сьома ітерація – об'єднуємо символи x_6 і x_7 . Присвоюємо значення: $x_6=1$; $x_7=0$. Їх сумарна ймовірність складає 0,13. Новий об'єднаний символ x_6 - x_7 вводиться в додатковий сьомий стовбець у 2 рядок.

Восьма ітерація – об'єднуємо новий символ x_{11} - x_{13} - x_{14} четвертої ітерації і символ x_5 . Присвоюємо значення: x_{11} - x_{13} - $x_{14}=1$; $x_5=0$. Їх сумарна ймовірність складає 0,15. Новий об'єднаний символ x_{11} - x_{13} - x_{14} - x_5 вводиться в додатковий восьмий стовбець у 2 рядок.

Дев'ята ітерація – об'єднуємо новий символ x_9 - x_{10} п'ятої ітерації і символ x_4 . Присвоюємо значення: x_9 - $x_{10}=1$; $x_4=0$. Їх сумарна ймовірність складає 0,18. Новий об'єднаний символ x_9 - x_{10} - x_4 вводиться в додатковий дев'ятий стовбець у 2 рядок.

Десята ітерація – об'єднуємо символи x_2 і x_3 . Присвоюємо значення: $x_2=1$; $x_3=0$. Їх сумарна ймовірність складає 0,20. Новий об'єднаний символ x_2 - x_3 вводиться в додатковий десятий стовбець у 2 рядок.

Таблиця 5.2 – Визначення коду Хаффмена для прикладу 5.1

[illegible]

Одинадцята ітерація – об'єднуємо новий символ x_6-x_7 з сьомої ітерації і новий символ $x_8-x_{12}-x_{15}-x_{16}$ з шостої ітерації. Присвоюємо значення: $x_6-x_7=1$; $x_8-x_{12}-x_{15}-x_{16}=0$. Їх сумарна ймовірність складає 0,25. Новий об'єднаний символ $x_6-x_7-x_8-x_{12}-x_{15}-x_{16}$ вводиться в одинадцятий стовбець у 1 рядок.

Дванадцята ітерація – об'єднуємо новий символ $x_9-x_{10}-x_4$ з дев'ятої ітерації і новий символ $x_{11}-x_{13}-x_{14}-x_5$ з восьмої ітерації. Присвоюємо значення: $x_9-x_{10}-x_4=1$; $x_{11}-x_{13}-x_{14}-x_5=0$. Їх сумарна ймовірність складає 0,33. Новий об'єднаний символ $x_9-x_{10}-x_4-x_{11}-x_{13}-x_{14}-x_5$ вводиться в дванадцятий стовбець у 1 рядок.

Тринадцята ітерація – об'єднуємо символ x_1 і новий символ x_2-x_3 з десятої ітерації. Присвоюємо значення: $x_1=1$; $x_2-x_3=0$. Їх сумарна ймовірність складає 0,42. Новий об'єднаний символ $x_1-x_2-x_3$ вводиться в тринадцятий стовбець у 1 рядок.

Чотирнадцята ітерація – об'єднуємо новий символ $x_9-x_{10}-x_4-x_{11}-x_{13}-x_{14}-x_5$ з дванадцятої ітерації і новий символ $x_6-x_7-x_8-x_{12}-x_{15}-x_{16}$ з одинадцятої ітерації. Присвоюємо значення: $x_9-x_{10}-x_4-x_{11}-x_{13}-x_{14}-x_5=1$; $x_6-x_7-x_8-x_{12}-x_{15}-x_{16}=0$. Їх сумарна ймовірність складає 0,58. Новий об'єднаний символ $x_9-x_{10}-x_4-x_{11}-x_{13}-x_{14}-x_5-x_6-x_7-x_8-x_{12}-x_{15}-x_{16}$ вводиться в чотирнадцятий стовбець у 1 рядок.

У п'ятнадцятому стовбці буде сформований останній символ $x_9-x_{10}-x_4-x_{11}-x_{13}-x_{14}-x_5-x_6-x_7-x_8-x_{12}-x_{15}-x_{16}-x_1-x_2-x_3$. Він буде об'єднувати всі символи, тому його ймовірність буде дорівнювати 1.

Цей алгоритм дозволяє знайти код кожного вихідного символу, який визначений в табл. 5.2.

Далі будуємо кодове дерево (рис. 5.1): верхня точка (корінь дерева) має ймовірність $p=1$; від неї відходять гілки, які відповідають кодовому алфавіту q (1 або 0); таке розгалуження виконується до всіх шістнадцяти символів послідовності N (вершини дерева).

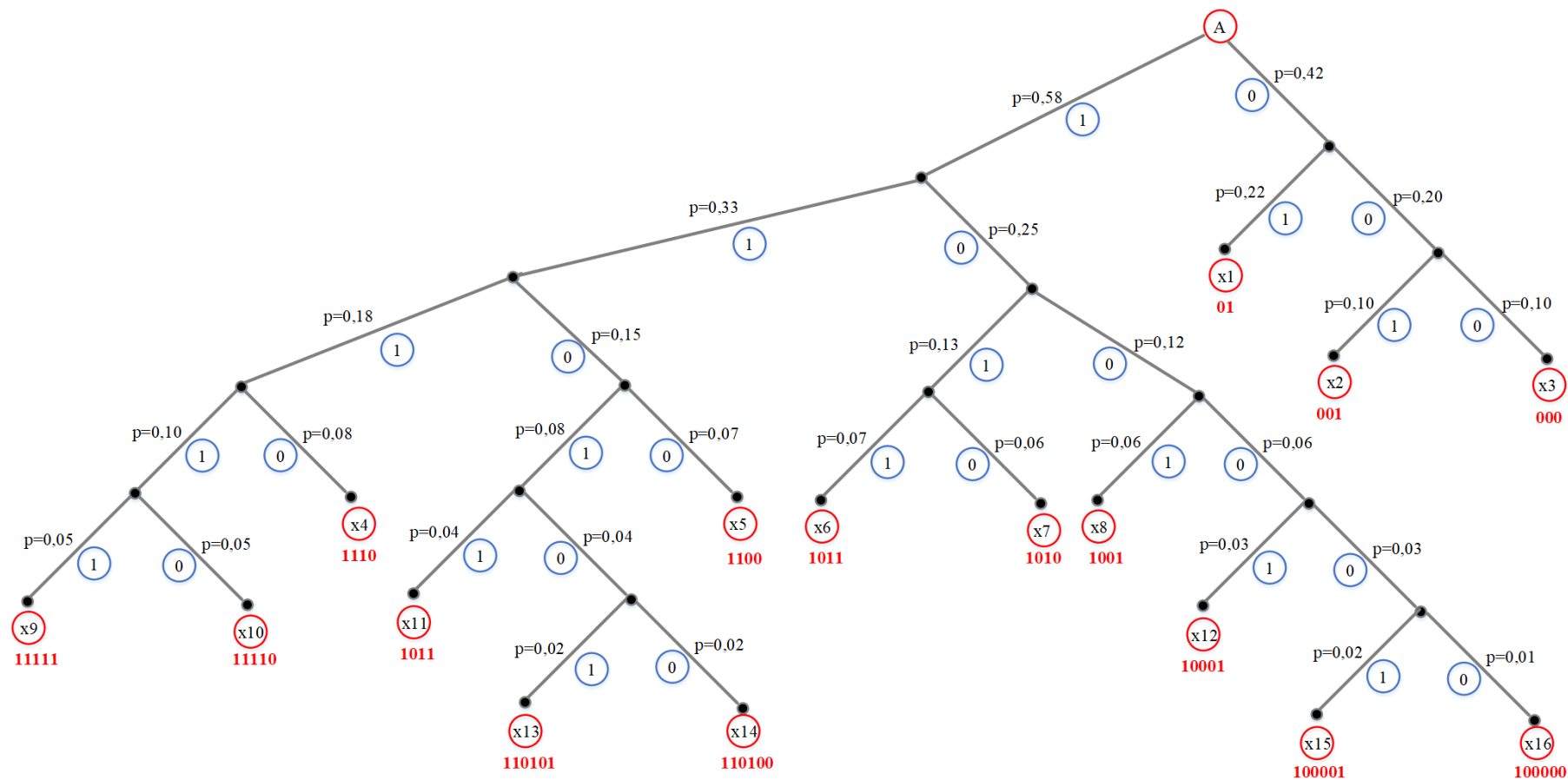


Рисунок 5.1 – Кодовое дерево Хаффмена для прикладу 5.1

Приклад 5.2. (Код Хаффмена).

Задано вісім символів ($N=8$) з різною імовірністю виникнення. Вони приведені у табл. 5.3 у порядку зменшення ймовірності їх виникнення.

Побудувати код Хаффмена з використанням кодового алфавіту $q=2$ з символами 1 і 0.

Таблиця 5.3 – Вхідні дані до прикладу 5.2

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
0,22	0,20	0,16	0,16	0,10	0,10	0,04	0,02

Таблиця 5.4 – Визначення коду Хаффмена для прикладу 5.2

Символ	Імовірність	Допоміжні етапи							Код
		1	2	3	4	5	6	7	
x_1	0,22	0,22	0,22	0,26	0,32	0,42	0,58	1,0	01
x_2	0,20	0,20	0,20	0,22	0,26	0,32	0,42	-	00
x_3	0,16	0,16	0,16	0,20	0,22	0,26	-	-	111
x_4	0,16	0,16	0,16	0,16	0,20	-	-	-	110
x_5	0,10	0,10	0,16	0,16	-	-	-	-	100
x_6	0,10	0,10	0,10	-	-	-	-	-	1011
x_7	0,04	0,06	-	-	-	-	-	-	10101
x_8	0,02	-	-	-	-	-	-	-	10100

Приклад 5.3. (Код Хаффмена матричний метод).

Визначити матрицю ідентифікації для прикладу 5.1 за ієрархічним деревом коду Хаффмена (рис. 5.1).

Для ребер дерева коду Хаффмена вказуємо значення ваги: 1 або 0 (рис. 5.2, значення вказані в окружностях з синім контуром). Значення ймовірностей p_i для ребер дерева з графу прибираються.

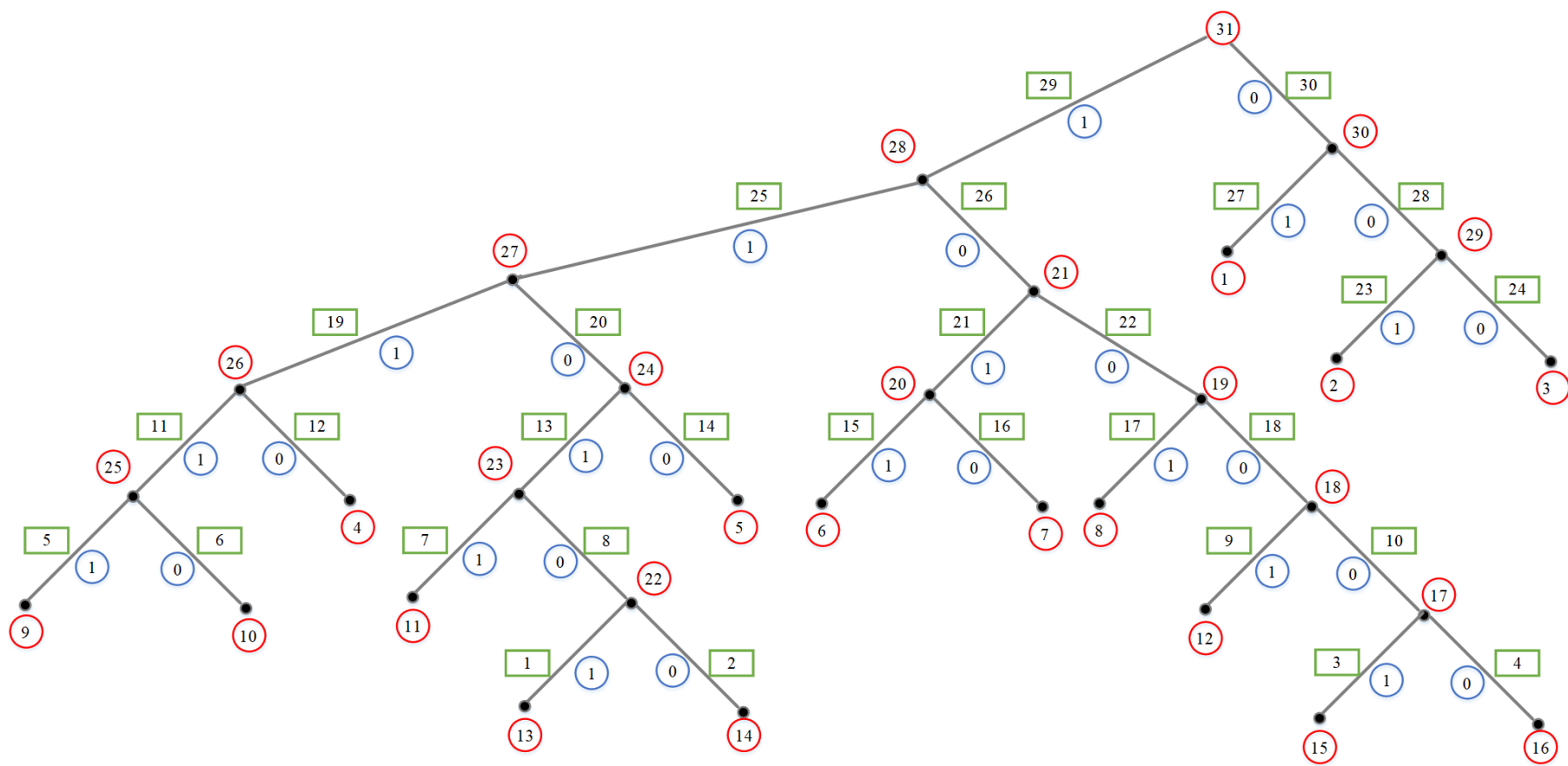


Рисунок 5.2 – Граф для прикладу 5.3

Нумеруються вершини дерева коду Хаффмена: символам, що кодуються, присвоюються номери від 1 до 16 ($N=16$) (рис. 5.2, значення вказані в окружностях з червоним контуром). Внутрішні вершини нумеруються наступними: їх нумерація відбувається з 17 до 30 ($K=14$). Кореню дерева надається останній номер, який буде складати $n=N+K+1=16+14+1=31$.

Як бачимо з рис.5.2, кількість ребер дерева коду Хаффмена $m = 30$. Вони нумерувалися від найбільш віддалених зовнішніх вершин до кореня дерева (рис. 5.2, значення вказані в квадратах із зеленим контуром).

Якщо подивитися на отриманий граф, то кореню дерева відповідає максимальне значення номера вершини $n=31$ і до нього підходять два ребра з максимальними значеннями 29 і 30.

Далі формується прямокутна матриця ідентифікації для ієрархічного дерева коду Хаффмена із кількістю стовпчиків $n=31$ та кількістю рядків $m=30$.

В отриманій матриці – табл. 5.5 – першому ребру відповідає перший рядок. Перше ребро має вагу «1» і зв'язує 13 і 22 вузол. Тому покажемо, що в першому рядку в 13 і 22 стовбцях підставляємо значення «1». В інших стовбцях цього рядку ставиться значення «-1» оскільки перше ребро не зв'язано з іншими вершинами.

Друге ребро відповідає другому рядку матриці. Друге ребро має вагу «0» і зв'язує 14 і 22 вузол. Вкажемо тепер в другому рядку в 14 і 22 стовбцях значення «0», а в інших стовбцях цього рядку – значення «-1».

Третє ребро відповідає третьому рядку матриці. Третє ребро має вагу «1» і зв'язує 15 і 17 вузол. Вкажемо тепер в третьому рядку в 15 і 17 стовбцях значення «1», а в інших стовбцях цього рядку – значення «-1».

Зрозуміло, що таким же чином заповнюються інші рядки.

Остаточний результат – в табл. 5.2.

Таблиця 5.5 – Матриця ідентифікації коду Хаффмена

		<i>n</i> – вершини																																
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31		
m - ребра	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		
	2	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1		
	3	-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		
	4	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1		
	5	-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		
	6	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1		
	7	-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		
	8	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1		
	9	-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		
	10	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1		
	11	-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		
	12	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	
	13	-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		
	14	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	
	15	-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	
	16	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	
	17	-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	
	18	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	
	19	-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	
	20	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	0	-1	-1	-1	-1	
	21	-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	
	22	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	
	23	-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	
	24	-1	-1	0	-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	0	-1	-1
	25	-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
	26	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1
	27	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	
	28	-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	0	0	-1	
	29	-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	
	30	-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	0	0	

За сформованою матрицею ідентифікації I можна знайти розряди кодів Хаффмена для всіх вхідних символів.

Наприклад, для x_{16} – це шістнадцятий стовбець. В шістнадцятому стовбці (табл. 5.6) значення не дорівнює «-1» для елемента четвертого рядка. Це значення «0». Воно вписується у перший розряд символу x_{16} .

У четвертому рядку визначаємо, що для елемента в сімнадцятому стовбці виконується умова $I(4,16)=(4,17)=0$. У сімнадцятому стовбці елемент в десятому рядку матриці ідентифікації не дорівнює «-1»: $I(10,17)=\langle 0 \rangle$. Це значення вписується у другий розряд символу x_{16} .

У десятому рядку визначаємо, що для елемента в вісімнадцятому стовбці виконується умова $I(10,17)=(10,18)=0$. У вісімнадцятому стовбці елемент у вісімнадцятому рядку матриці ідентифікації не дорівнює «-1»: $I(18,18)=\langle 0 \rangle$. Це значення вписується у третій розряд символу x_{16} .

У вісімнадцятому рядку визначаємо, що для елемента в дев'ятнадцятому стовбці виконується умова $I(18,18)=(18,19)=0$. У дев'ятнадцятому стовбці елемент у двадцять другого рядку матриці ідентифікації не дорівнює «-1»: $I(22,19)=\langle 0 \rangle$. Це значення вписується у четвертий розряд символу x_{16} .

У двадцять другому рядку визначаємо, що для елемента в двадцять першому стовбці виконується умова $I(22,19)=(22,21)=0$. У двадцять першому стовбці елемент у двадцять шостому рядку матриці ідентифікації не дорівнює «-1»: $I(26,21)=\langle 0 \rangle$. Це значення вписується у п'ятий розряд символу x_{16} .

У двадцять шостому рядку визначаємо, що для елемента в двадцять восьмому стовбці виконується умова $I(26,21)=(26,28)=0$. У двадцять восьмому стовбці елемент у двадцять дев'ятому рядку матриці ідентифікації не дорівнює «-1»: $I(29,28)=\langle 1 \rangle$. Це значення вписується у шостий розряд символу x_{16} .

У двадцять дев'ятому рядку визначаємо, що для елемента в тридцять першому стовбці виконується умова $I(29,28)=(29,31)=1$.

Так як номер стовбця дорівнює $n=31$, то завершуємо формування коду символу x_{16} .

Таблиця 5.6 – Визначення коду Хаффмена за таблицею ідентифікації

		<i>n</i> – вершины																															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
<i>m</i> - ребра	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	
	2	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
	3	-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
	4	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
	5	-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
	6	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1
	7	-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
	8	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1
	9	-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
	10	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
	11	-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
	12	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1
	13	-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
	14	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1
	15	-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
	16	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
	17	-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
	18	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
	19	-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
	20	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	0	-1	-1	-1	-1
	21	-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
	22	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
	23	-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
	24	-1	-1	0	-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	0	-1	-1
	25	-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
	26	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	0	-1	-1
	27	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
	28	-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	0	0
	29	-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
	30	-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	0	0

Приклад 5.4. (матриця сум коду Хаффмена).

Для вихідних даних прикладу 5.2 визначити матрицю сум M_Σ ; матриця номерів елементів S_{el} , які сумуються; і побудувати код Хаффмена цим способом.

У першому рядку (нульова ітерація) матриці сум вказуються ймовірності виникнення символів в упорядкованому порядку M_Σ :

$$M_\Sigma = \begin{pmatrix} 0,22 & 0,2 & 0,16 & 0,16 & 0,10 & 0,10 & 0,04 & 0,02 \\ 0,22 & 0,2 & 0,16 & 0,16 & 0,10 & 0,10 & 0,06 & 0 \\ 0,22 & 0,2 & 0,16 & 0,16 & 0,16 & 0,10 & 0 & 0 \\ 0,26 & 0,22 & 0,2 & 0,16 & 0,16 & 0 & 0 & 0 \\ 0,32 & 0,26 & 0,22 & 0,2 & 0 & 0 & 0 & 0 \\ 0,42 & 0,32 & 0,26 & 0 & 0 & 0 & 0 & 0 \\ 0,58 & 0,42 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix};$$

$$M_\Sigma = \begin{array}{c|cccccccc} & x1 & x2 & x3 & x4 & x5 & x6 & x7 & x8 \\ \hline \begin{array}{c} 0,22 \\ 0,22 \\ 0,22 \\ 0,26 \\ 0,32 \\ 0,42 \\ 0,58 \\ 1 \end{array} & \begin{array}{c} 0,2 \\ 0,2 \\ 0,2 \\ 0,22 \\ 0,26 \\ 0,32 \\ 0,42 \\ 0 \end{array} & \begin{array}{c} 0,16 \\ 0,16 \\ 0,16 \\ 0,2 \\ 0,22 \\ 0,26 \\ 0 \\ 0 \end{array} & \begin{array}{c} 0,16 \\ 0,16 \\ 0,16 \\ 0,16 \\ 0,2 \\ 0 \\ 0 \\ 0 \end{array} & \begin{array}{c} 0,10 \\ 0,10 \\ 0,16 \\ 0,16 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} & \begin{array}{c} 0,10 \\ 0,10 \\ 0,10 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} & \begin{array}{c} 0,04 \\ 0,06 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} & \begin{array}{c} 0,02 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \end{array}$$

Так як $M_\Sigma(1,7)$ і $M_\Sigma(1,8)$ мають найменші значення ймовірностей, то на першій ітерації вони об'єднуються: $M_\Sigma(2,7)=M_\Sigma(1,7)+M_\Sigma(1,8)=0,06$. Вага $M_\Sigma(1,7)$ буде складати 1, а вага $M_\Sigma(1,8)$ – 0. Відповідно до цих даних в матрицю номерів елементів S_{el} , які сумуються, в перший стовбець в перший рядок вноситься номер стовбця 7 з вагою одиниця, а в другий рядок – номер стовбця 8 з вагою нуль:

$$S_{el} = \begin{pmatrix} 7 & 6 & 6 & 3 & 1 & 3 & 3 \\ 8 & 7 & 5 & 4 & 2 & 6 & 1 \end{pmatrix}.$$

На другій ітерації нові елементи матриця сум:

$M_{\Sigma}(3,5)=M_{\Sigma}(2,6)+M_{\Sigma}(2,7)=0,10+0,06=0,16$; а нові елементи матриці номерів:
 $S_{ел}(2,1)=6$; $S_{ел}(2,2)=7$.

Третя ітерації:

$M_{\Sigma}(4,1)=M_{\Sigma}(3,5)+M_{\Sigma}(3,6)=0,16+0,10=0,26$; а $S_{ел}(3,1)=6$; $S_{ел}(3,2)=5$.

Четверта ітерації:

$M_{\Sigma}(5,1)=M_{\Sigma}(4,4)+M_{\Sigma}(4,5)=0,16+0,16=0,32$; а $S_{ел}(4,1)=3$; $S_{ел}(4,2)=4$.

П'ята ітерації:

$M_{\Sigma}(6,1)=M_{\Sigma}(5,3)+M_{\Sigma}(5,4)=0,22+0,2=0,42$; а $S_{ел}(5,1)=1$; $S_{ел}(5,2)=2$.

Шоста ітерації:

$M_{\Sigma}(7,1)=M_{\Sigma}(6,2)+M_{\Sigma}(6,3)=0,32+0,26=0,58$; а $S_{ел}(6,1)=3$; $S_{ел}(6,2)=6$.

Сьома ітерації:

$M_{\Sigma}(8,1)=M_{\Sigma}(7,1)+M_{\Sigma}(7,2)=0,58+0,42=1$; а $S_{ел}(7,1)=3$; $S_{ел}(7,2)=1$.

Останній ряд матриці сум, як передбачалося, – це 1 в першій колонці та нулі в інших колонках. Далі, для визначення коду Хаффмена працюємо з матрицею номерів елементів, які сумуються. В перший розряд символу x_7 вписується значення ваги 1, а перший розряд символу x_8 – значення 0:

$$S_{ел} = \begin{pmatrix} \boxed{1} & 6 & 6 & 3 & 1 & 3 & 3 \\ 7 & 6 & 6 & 3 & 1 & 3 & 3 \\ 8 & 7 & 5 & 4 & 2 & 6 & 1 \\ \boxed{0} & & & & & & \end{pmatrix}$$

В перший розряд символу x_6 вписується значення ваги 1, а другий розряд символів x_7 і x_8 – значення 0:

$$S_{ел} = \begin{pmatrix} 01 & \boxed{1} & 6 & 6 & 3 & 1 & 3 & 3 \\ 7 & 6 & 6 & 3 & 1 & 3 & 3 \\ 8 & 7 & 5 & 4 & 2 & 6 & 1 \\ 00 & \boxed{0} & & & & & \end{pmatrix}$$

Значення одиниці вписується в другий розряд символу x_6 , в третій розряд символів x_7 і x_8 , а значення нуль – в перший розряд символу x_5 :

$$S_{el} = \begin{pmatrix} 7 & 6 & 6 & 3 & 1 & 3 & 3 \\ 8 & 7 & 5 & 4 & 2 & 6 & 1 \end{pmatrix}$$

101 11 100

Diagram illustrating the assignment of values to the first row of the matrix S_{el} . The values 1 and 0 are shown in blue, indicating the placement of the unit and zero respectively. Red arrows indicate the flow of information from the input symbols to the matrix elements.

В перший розряд символу x_3 вписується значення ваги 1, перший розряд символів x_4 – значення 0, перший розряд символу x_1 – значення 1, перший розряд символу x_2 – значення 0:

$$S_{el} = \begin{pmatrix} 7 & 6 & 6 & 3 & 1 & 3 & 3 \\ 8 & 7 & 5 & 4 & 2 & 6 & 1 \end{pmatrix}$$

101 11 100 0

Diagram illustrating the assignment of values to the second row of the matrix S_{el} . The values 1 and 0 are shown in blue, indicating the placement of the unit and zero respectively. Red arrows indicate the flow of information from the input symbols to the matrix elements.

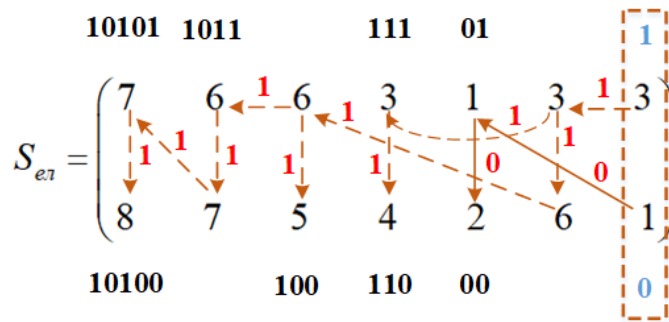
Значення одиниці вписується в другий розряд символів x_3 і x_4 , а значення нуль – в другий розряд символу x_5 , в третій розряд символу x_6 , в четвертий розряд символів x_7 і x_8 :

$$S_{el} = \begin{pmatrix} 7 & 6 & 6 & 3 & 1 & 3 & 3 \\ 8 & 7 & 5 & 4 & 2 & 6 & 1 \end{pmatrix}$$

0101 011 11 1 0100 00 10 0

Diagram illustrating the assignment of values to the third row of the matrix S_{el} . The values 1 and 0 are shown in blue, indicating the placement of the unit and zero respectively. Red arrows indicate the flow of information from the input symbols to the matrix elements.

Значення одиниці вписується в третій розряд символів x_3 і x_4 , в третій розряд символу x_5 , в четвертий розряд символу x_6 , в п'ятий розряд символів x_7 і x_8 , а значення нуль – в другий розряд символів x_1 і x_2 :



Коди символів методом Хаффмена отримані за допомогою матриці сум M_Σ і матриці номерів елементів S_{el} :

$$S_{el} = \begin{pmatrix} 7 & 6 & 6 & 3 & 1 & 3 & 3 \\ 8 & 7 & 5 & 4 & 2 & 6 & 1 \end{pmatrix}$$

10101 1011 111 01
10100 100 110 00

Отримані результатів співпадають з результатами прикладу 5.2

Індивідуальне завдання

Інформаційні символи (табл. 4.3), які досліджувалися в індивідуальному завданні попередньої теми, закодувати кодом Хаффмена (табл. 5.7) з використанням трьох алгоритмів:

- 1) побудова коду Хаффмена і визначення кодового дерева;
- 2) граф кодового дерева Хаффмена і матриця ідентифікації;
- 3) побудова коду Хаффмена на підставі матриці сум M_Σ і матриці номерів елементів S_{el} .

Таблиця 5.7 – Результати виконання індивідуального завдання

інформаційний символ	код Хаффмена
...	

Питання для самоперевірки

1. Принцип побудови коду Хаффмена і визначення кодового дерева.
2. Принцип побудови графу кодового дерева Хаффмена і матриці ідентифікації.
3. Принцип побудови коду Хаффмена на підставі матриці сум M_Σ і матриці номерів елементів $S_{\text{ел}}$.

Література

[1, 3].

ЛАБОРАТОРНА РОБОТА № 6

ДОСЛІДЖЕННЯ ЗАВАДОСТІЙКИХ КОДІВ. КОД ХЕММІНГА

Мета роботи: дослідити теоретичні принципи формування коду Хеммінга.

Теоретична частина

Розглянемо двійкові коди, що виявляють та виправляють помилки.

Код з перевіркою на парність (parity-check code).

Область застосування: виявлення однократної помилки чи помилок непарної кратності.

Принцип побудови коду: до інформаційних бітів, які передаються, додається один контрольний біт, значення якого обирається таким, щоб сума всіх бітів за модулем 2 дорівнювала 0.

Інформаційний сигнал в первинному коді містить комбінацію із k -елементів. Відправлений сигнал доповнюється одним елементом – бітом парності (контрольним бітом) r_0 , який визначається сумою k -бітів за модулем 2:

$$r_0 = \sum_{i=1}^k \oplus a_i. \text{ У відправленому сигналі кількість одиниць повинна бути парною.}$$

В отриманому сигналі буде n -символів: $n=k+1$. На наступному етапі для виявлення помилки визначається кодовий синдром $s_1 = \sum_{i=1}^{n-1} a_i \oplus r_0$, де a_i , r_0 – прийняті інформаційні та контрольний елементи.

Помилки в коді прийнятого сигналу немає, якщо $s_1=0$; і навпаки, помилка є, якщо $s_1=1$.

Узагальнена схема кодера і декодера коду з перевіркою на парність приведена на рис. 6.1.

Недолік: код не дозволяє виявити парну кількість помилок.

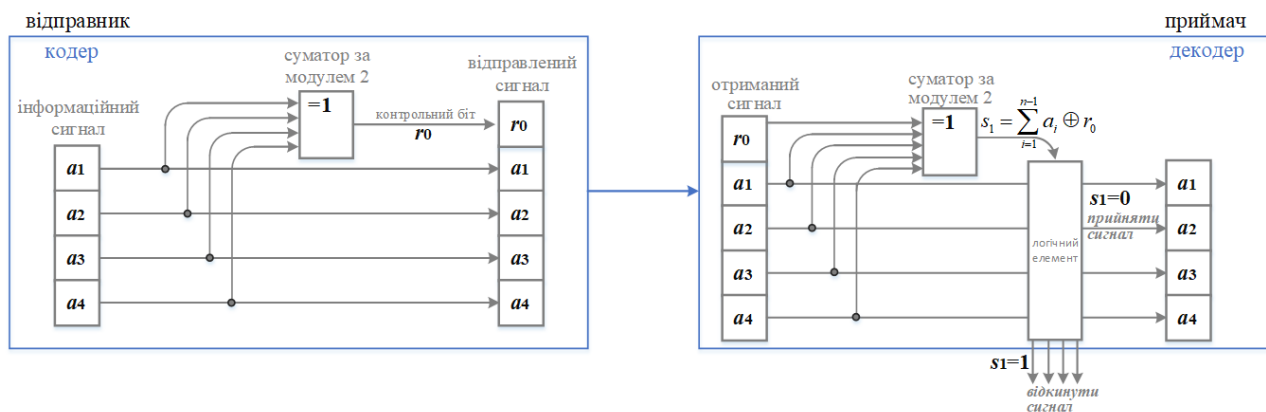


Рисунок 6.1 – Структурна схема кодера і декодера коду з перевіркою на парність

Принцип побудови коду з перевіркою на парність проілюстровано в прикладі 6.1 практичної частини роботи.

Код з перевіркою на непарність.

Відмінність від попереднього способу кодування полягає у тому, що кожна кодова комбінація повинна мати непарне число одиниць. Додатковий перевірючий елемент r_0 (контрольний біт) формується, виходячи з числа одиниць у первинній кодовій комбінації: при парному числі одиниць він дорівнює одиниці, при непарному – нулю. В прийнятому сигналі для виявлення помилки в кодовій комбінації виконується перевірка на непарність.

Код включає $n-1$ інформаційних елементів a_i та один перевірючий елемент r_0 , має довжину n , і спроможність виявити помилки, як і код з перевіркою на парність.

Роздивимось наступними коди з повторенням (repetition code).

Код з простим повторенням (з повторенням без інверсії).

Код містить k інформаційних та $r=k$ перевірючих елементів.

Принцип побудови коду: кожний біт послідовності, яка передається, повторюється двічі: $r_i = a_i$ ($a_1, r_1, a_2, r_2, \dots, a_i, r_i, \dots, a_k, r_k$). В отриманому повідомленні помилки виявляються шляхом порівняння інформаційних a_i та перевірючих r_i елементів одного розряду.

Помилки в коді прийнятого сигналу немає, якщо інформаційний a_i й перевірочний r_i елементи одного розряду збігаються ($a_i=r_i$); помилка є, якщо елементи a_i і r_i не збігаються ($a_i \neq r_i$).

Недолік: код не дозволяє виявити помилки, які повторюються в інформаційному і перевірочному елементах одного розряду.

Принцип побудови коду з простим повторенням проілюстровано в прикладі 6.2 практичної частини роботи.

Код зі зворотнім повторенням (інверсний код, код Бауера).

Код містить k інформаційних та k перевірочних елементів. Значення перевірочних k елементів у коді залежать від значення суми за модулем 2 всіх інформаційних k елементів.

Принцип побудови коду. Якщо сума за модулем 2 інформаційних елементів дорівнює нулю $\sum_{i=1}^k \oplus a_i = 0$, тобто число одиниць у первинному коду комбінації є парним, то перевірочні елементи просто повторюють інформаційні: $r_i = a_i$, $i=1 \dots k$ ($a_1, a_2, \dots a_i, \dots a_k, r_1, r_2, \dots r_i, \dots r_k$). Якщо сума за модулем 2 інформаційних елементів дорівнює одиниці $\sum_{i=1}^k \oplus a_i = 1$, тобто число одиниць у первинному коду комбінації є непарним, то перевірочні елементи повторюють інформаційні в інвертованому вигляді (у зворотному коді): $r_i = a_i \oplus 1$, $i=1 \dots k$ ($a_1, a_2, \dots a_i, \dots a_k, r_1, r_2, \dots r_i, \dots r_k$).

Виявлення декодером помилок в коді прийнятого сигналу проводиться наступним чином:

- послідовність, що складається з $2k$ елементів, розділяється на дві послідовності з k елементів;
- у перших k елементах підсумовуються кількість одиниць;
- якщо кількість одиниць серед перших k елементів прийнятої комбінації *парна*, то друга частина з k елементів приймається без змін: $r_i = a_i$, $i=1 \dots k$.
- якщо кількість одиниць серед перших k елементів прийнятої комбінації *непарна*, то решта k елементів приймається у реверсі (інвертуються): $r_i = a_i \oplus 1$.

- обидві частини кодової комбінації поелементно порівнюються (перший елемент з першим, другий – з другим і т. д.);

- при наявності хоча б одного незбігу вся послідовність елементів бракується.

Така побудова коду дозволяє виявити дуже багато варіантів спотворення елементів.

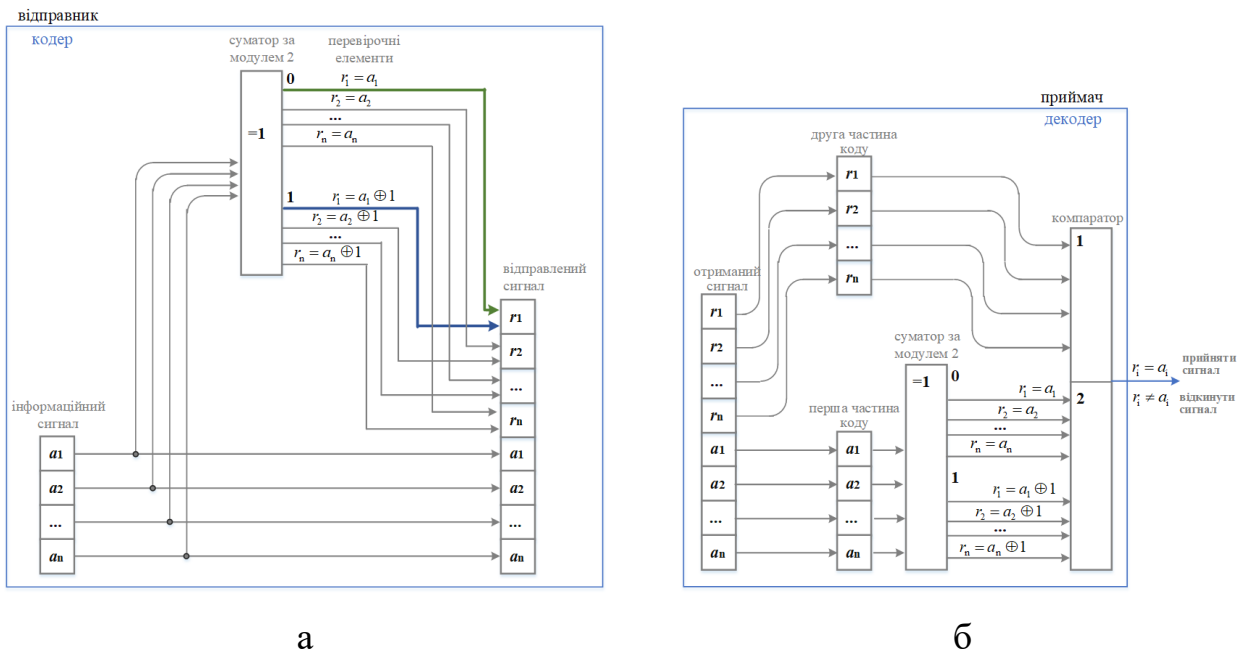


Рисунок 6.2 – Структурна схема кодера (а) і декодера (б) інверсного коду

Принцип побудови коду з простим повторенням проілюстровано в прикладі 6.3 практичної частини роботи.

Кореляційний код (манчестерський код).

Код не містить перевірочних елементів. Це спосіб кодування сигналів у каналі зв'язку, якому відповідає не завадостійке, а натуральне кодування. З іншої точки зору – це не формування сигналу, а принцип кодування інформації.

Принцип побудови кореляційного манчестерського коду. Формується кодова послідовність, у якій кожному біту числа, яке кодується, відповідає 2 біти кодової послідовності. Символу «1» відповідає послідовність двох бітів

«10», а символу «0» – послідовність «01». Послідовності символів «11» та «00» є забороненими.

Виявлення декодером помилок в коді прийнятого сигналу проводиться наступним чином: у двох сусідніх елементах одного такту фіксується перехід «0®1» або «1®0». Якщо виявляється послідовність «11» або «00», то фіксується помилка. Якщо в середині такту сигнал змінив полярність, то помилка відсутня; якщо ні – то фіксується помилка.

Така побудова коду дозволяє виявити помилки у кожній парі елементів одного такту, але не в змозі виявити двократні помилки в одному такті. Виявлення помилок передавання інформації забезпечується за рахунок перевантаження ресурсів каналу зв'язку.

Принцип побудови коду з простим повторенням проілюстровано в прикладі 6.4 практичної частини роботи.

Коди Хеммінга.

В коді Хеммінга використовуються перевірочні й інформаційні символи. Позиції перевірочних символів визначаються степеню 2^i , $i = 1, 2, \dots, n$, тобто їх позиції – це 1, 2, 4, 8, 16, 32, Інформаційні символи будуть займати інші вільні позиції 3, 5, 6, 7, 9, 11, 12,

Значення перевірочних символів визначається синдромом, який перевіряє на парність символи на певних позиціях.

Перший синдром визначається перевіркою на парність інформаційних символів, які мають позицію, номер якої в двійковій системі включає «1» на першій позиції, тобто це 1_{10} (1_2), 3_{10} (11_2), 5_{10} (101_2), 7_{10} (111_2), 9_{10} (1001_2), 11_{10} (1011_2), і так далі: $S_1 = n_1 \oplus n_3 \oplus n_5 \oplus n_7 \oplus n_9 \oplus n_{11} \oplus \dots =$.

Другий синдром визначається перевіркою на парність інформаційних символів, які мають позицію, номер якої в двійковій системі включає «1» на другій позиції, тобто це 2_{10} (10_2), 3_{10} (11_2), 6_{10} (110_2), 7_{10} (111_2), 10_{10} (1010_2), 11_{10} (1011_2), і так далі: $S_2 = n_2 \oplus n_3 \oplus n_6 \oplus n_7 \oplus n_{10} \oplus n_{11} \oplus \dots =$.

Третій синдром визначається перевіркою на парність інформаційних символів, які мають позицію, номер якої в двійковій системі включає «1» на

третій позиції, тобто це 4_{10} (100_2), 5_{10} (101_2), 6_{10} (110_2), 7_{10} (111_2), 12_{10} (1100_2), 13_{10} (1101_2), 14_{10} (1101_2), 15_{10} (1111_2), 20_{10} (10100_2), і так далі:
 $S_3 = n_4 \oplus n_5 \oplus n_6 \oplus n_7 \oplus n_{12} \oplus n_{13} \oplus n_{14} \oplus n_{15} \oplus n_{20} \oplus \dots = .$

Можна сформулювати наступне правило: перевірючий символ під номером i ($i=2^n$) перевіряє на парність всі наступні i біт (через кожні i біт), починаючи з позиції i .

Принцип побудови коду з простим повторенням проілюстровано в прикладах 6.5-6.7 практичної частини роботи.

Практична частина

Приклад 6.1. (Код з перевіркою на парність).

Для інформаційного сигналу 1011 контрольний біт буде визначатися, як $r_0 = 1 \oplus 0 \oplus 1 \oplus 1 = 1$. Зрозуміло, що відправлений сигнал буде визначено, як 10111. Приймач перевіряє кодовий синдром: $s_1 = \sum_{i=1}^{n-1} a_i \oplus r_0 = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 0$.

Так як $s_1 = 0$, то робиться висновок, що помилки в коді прийнятого сигналу немає.

Приклад 6.2. (Код з простим повторенням).

Для інформаційного сигналу 1011 відправлений сигнал буде визначено, як 11001111.

Приймач перевіряє рівність інформаційного a_i й перевірючого r_i елементи: $a_i = r_i$. Так як $a_4 = r_4 = 1$, $a_3 = r_3 = 0$, $a_2 = r_2 = 1$, $a_1 = r_1 = 1$, то робиться висновок, що помилки в коді прийнятого сигналу немає.

Якщо б хоча б для однієї позиції $a_i \neq r_i$, то робиться висновок, про помилку.

Приклад 6.3. (Код з інверсним повторенням).

Код містить 4 інформаційних елементів: 1011. Їх сума за модулем 2 дорівнює одиниці $1 \oplus 0 \oplus 1 \oplus 1 = 1$, тобто число одиниць у первинному коду

комбінації є непарним. Тому значення 4 перевірочних елементів повторюють інформаційні елементи в інвертованому вигляді (у зворотному коді): $r_i = a_i \oplus 1$, $r_4 = 1 \oplus 1 = 0$, $r_3 = 0 \oplus 1 = 1$, $r_2 = 1 \oplus 1 = 0$, $r_1 = 1 \oplus 1 = 0$.

Для інформаційного сигналу 1011 відправлений сигнал буде визначено, як 10110100.

Приймач розділяє прийнятий сигнал 10110100 на дві послідовності з 4 елементів: 1011 і 0100. У перших 4 елементах (1011) підсумовуються кількість одиниць: $1 \oplus 0 \oplus 1 \oplus 1 = 1$. Так як кількість одиниць серед перших 4 елементів прийнятої комбінації непарна, то решта з 4 елементів приймається у реверсі (інвертуються): $r_i = a_i \oplus 1$, $r_4 = 1 \oplus 1 = 0$, $r_3 = 0 \oplus 1 = 1$, $r_2 = 1 \oplus 1 = 0$, $r_1 = 1 \oplus 1 = 0$. Далі приймач порівнює обидві частини кодової комбінації поелементно. Так як кількість одиниць непарна, то порівнюються інвертовані значення: $r_4 = a_4 \oplus 1 = 0$, $r_3 = a_3 \oplus 1 = 1$, $r_2 = a_2 \oplus 1 = 0$, $r_1 = a_1 \oplus 1 = 0$. Всі рівності виконуються, тому робиться висновок, що помилки в коді прийнятого сигналу немає.

Якщо б була наявність хоча б одного незбігу $r_i \neq a_i \oplus 1$, то вся послідовність елементів бракувалася і робився би висновок про помилку.

Приклад 6.4. (Кореляційний манчестерський код).

Сигналу 1011 буде відповідати кореляційний манчестерський код: 10 01 10 10. Приймач перевіряє послідовність бітів в одному такті. Всі переходи виконуються: «1Ⓡ0», «0Ⓡ1», «1Ⓡ0», «1Ⓡ0», тому робиться висновок, що помилки в коді прийнятого сигналу немає.

Якщо б була наявність хоча б одного не переходу, то робився би висновок про наявність помилки і було б визначено положення елементу з помилкою.

Приклад 6.5. (Код Хеммінга).

Сигнал 0101 має чотири інформаційні символи (розряди). Відповідно до кількості інформаційних символів кількість перевірочних символів буде дорівнювати трьом: позиції 1 (2^0); 2 (2^1); 4 (2^2). Для наочності розрахунок зведено до табл. 6.1.

Таблиця 6.1

№ позиції		1	2	3	4	5	6	7
Символ		0	1	0	0	1	0	1
Синдром:	S_1	...	-	0	-	1	-	1
	S_2	-	...	0	-	-	0	1
	S_3	-	-	-	...	1	0	1

Відповідно до табл. 6.1: $S_1 = n_3 \oplus n_5 \oplus n_7 = 0 \oplus 1 \oplus 1 = 0$;

$S_2 = n_3 \oplus n_6 \oplus n_7 = 0 \oplus 0 \oplus 1 = 1$; $S_3 = n_5 \oplus n_6 \oplus n_7 = 1 \oplus 0 \oplus 1 = 0$.

Отже для коду Хеммінга сигналу 0101 відповідає відправлений сигнал 0100101.

Приклад 6.6. (Код Хеммінга).

Сигнал 11001001 має вісім інформаційних символів, тому кількість перевірочних символів буде дорівнювати чотирьом: позиції 1, 2, 4, 8. Розрахунок зведений до табл. 6.2.

Таблиця 6.2

№ позиції		1	2	3	4	5	6	7	8	9	10	11	12
Символ		1	1	1	0	1	0	0	0	1	0	0	1
Синдром:	S_1	...	-	1	-	1	-	0	-	1	-	0	-
	S_2	-	...	1	-	-	0	0	-	-	0	0	-
	S_3	-	-	-	...	1	0	0	-	-	-	-	1
	S_4	-	-	-	-	-	-	-	...	1	0	0	1

Відповідно до табл. 6.2:

$S_1 = n_1 \oplus n_3 \oplus n_5 \oplus n_7 \oplus n_9 \oplus n_{11} = 1 \oplus 1 \oplus 0 \oplus 1 \oplus 0 = 1$;

$$S_2 = n_2 \oplus n_3 \oplus n_6 \oplus n_7 \oplus n_{10} \oplus n_{11} = 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 1;$$

$$S_3 = n_4 \oplus n_5 \oplus n_6 \oplus n_7 \oplus n_{12} = 1 \oplus 0 \oplus 0 \oplus 1 = 0;$$

$$S_4 = n_8 \oplus n_9 \oplus n_{10} \oplus n_{11} \oplus n_{12} = 1 \oplus 0 \oplus 0 \oplus 1 = 0.$$

Отже для коду Хеммінга сигналу 11001001 відповідає відправлений сигнал 111010001001.

Приклад 6.7. (Код Хеммінга).

Отримано сигнал 110010001001 (табл. 6.3).

Виявити та визначити похибку.

Таблиця 6.3

№ позиції		1	2	3	4	5	6	7	8	9	10	11	12
Символ		1	1	0	0	1	0	0	0	1	0	0	1
Синдром:	S_1	1	-	0	-	1	-	0	-	1	-	0	-
	S_2	-	1	0	-	-	0	0	-	-	0	0	-
	S_3	-	-	-	0	1	0	0	-	-	-	-	1
	S_4	-	-	-	-	-	-	-	0	1	0	0	1

Відповідно до табл. 6.3:

$$S_1 = n_1 \oplus n_3 \oplus n_5 \oplus n_7 \oplus n_9 \oplus n_{11} = 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 \oplus 0 = 1;$$

$$S_2 = n_2 \oplus n_3 \oplus n_6 \oplus n_7 \oplus n_{10} \oplus n_{11} = 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 1;$$

$$S_3 = n_4 \oplus n_5 \oplus n_6 \oplus n_7 \oplus n_{12} = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0;$$

$$S_4 = n_8 \oplus n_9 \oplus n_{10} \oplus n_{11} \oplus n_{12} = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0.$$

Зауважимо, що отримані синдроми визначають позицію помилки в двійковій системі: 0011. Використовуючи отримані коефіцієнти визначаємо, що помилка знаходиться на третій позиції: $0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 3$.

Виправимо отриманий сигнал і отримуємо: 111010001001.

Можна зробити повторну перевірку:

$$S_1 = n_1 \oplus n_3 \oplus n_5 \oplus n_7 \oplus n_9 \oplus n_{11} = 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 0 = 0;$$

$$S_2 = n_2 \oplus n_3 \oplus n_6 \oplus n_7 \oplus n_{10} \oplus n_{11} = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 0;$$

$$S_3 = n_4 \oplus n_5 \oplus n_6 \oplus n_7 \oplus n_{12} = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0;$$

$$S_8 = n_8 \oplus n_9 \oplus n_{10} \oplus n_{11} \oplus n_{12} = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0.$$

Ми отримали номер нульової позиції, або іншими словами відсутність помилки. Отже для коду Хеммінга інформаційний сигнал визначається як 111010001001.

Індивідуальне завдання

Інформаційні символи (табл. 4.3), які досліджувалися в індивідуальному завданні попередньої теми, закодувати наступними кодами (табл. 6.4): код з перевіркою на парність, інверсний код; кореляційний код; код Хеммінга.

Таблиця 6.4 – Результати виконання індивідуального завдання

інформаційний символ	код з перевіркою на парність	інверсний код	кореляційний код	код Хеммінга
...				

Питання для самоперевірки

1. Принцип побудови коду з перевіркою на парність.
2. Принцип побудови інверсного коду.
3. Принцип побудови коду Хеммінга.

Література

[1, 3].

ПОРЯДОК ВИКОНАННЯ ТА ОФОРМЛЕННЯ РОЗРАХУНКОВОЇ РОБОТИ

Завдання до виконання розрахункової роботи з дисциплін «Метрологічне забезпечення систем технічного захисту інформації», «Методи дослідження та аналізу інформаційних даних», «Електронні системи» обирається відповідно з номером варіанту (номер у журналі групи) і затверджується керівником проекту.

Зміст розрахункової роботи наступний:

- 1). Побудова коду Грея для кількості бітів $n=4$ (індивідуальне завдання роботи 3).
- 2). Побудова коду Шенона-Фано і графу кодового дерева Шенона-Фано (індивідуальне завдання роботи 4, табл. 4.3).
- 3). Код Хаффмена.
 - 3.1). Побудова коду Хаффмена і кодового дерева Хаффмена.
 - 3.2). Побудова графу кодового дерева Хаффмена і матриці ідентифікації.
 - 3.3). Побудова коду Хаффмена на підставі матриці сум M_{Σ} і матриці номерів елементів $S_{\text{ел}}$ (індивідуальне завдання роботи 5).
- 4). Побудова коду з перевіркою на парність, інверсного коду; кореляційного коду і коду Хеммінга (індивідуальне завдання роботи 6).

Текст розрахункової роботи набирають на комп'ютері: тип шрифту - Times New Roman; розмір шрифту - 14; міжрядковий інтервал - 1,5; поля: зліва, зверху і знизу - 20 мм, справа - 10 мм; абзацний відступ повинен бути однаковим упродовж усього тексту і дорівнювати —1,25 см.

Критерії оцінювання розрахункової роботи з дисциплін виглядають наступним чином: 20 б, якщо завдання виконано у повному обсязі; 10 б, якщо завдання виконано у повному обсязі із зауваженнями; 0 б, якщо завдання виконано не у повному обсязі, є серйозні помилки. Під час семестрового контролю враховуються результати виконання розрахункової роботи відповідно до розподілу балів.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Денбновецький, С. В. Кодування сигналів в електронних системах. Частина 3. Способи кодування сигналів. Том 1. Натуральні, ефективні та лінійні коди [Електронний ресурс] : навч. посіб. для студ. спеціальності 171 «Електроніка», освітньої програми «Електронні прилади та пристрої» / С. В. Денбновецький, І. В. Мельник, Л. Д. Писаренко ; КПІ ім. Ігоря Сікорського. – Київ : КПІ ім. Ігоря Сікорського, 2021. – 470 с. – Режим доступу: <https://ela.kpi.ua/handle/123456789/40102>
2. Денбновецький, С.В. Електронні системи : навч. посіб. / С.В. Денбновецький, О.В. Лецишин. – К. : НТУУ „КПІ”, 2011. – 288 с.
3. Майданюк, В. П. Кодування та захист інформації : навч. посіб. / В.П. Майданюк. – Вінниця : ВНТУ, 2009. – 164 с.
4. Кузьмин, И.В. Основы теории информации и кодирования / И.В. Кузьмин, В.А. Кедрус. – К. : Вища шк., 1977. – 238 с.
5. Пеннин, П.И. Радиотехнические системы передачи информации / П.И. Пеннин, Л.И. Филиппов. – М. : Радио и связь, 1984. – 256 с.
6. Цымбыл, В.П. Основы теории информации и кодирования / В.П. Цымбыл. – К. : Вища шк., 1992. – 294 с.
7. Дмитриев, В.И. Прикладная теория информации / В.И. Дмитриев. – М. : Высш. шк., 1989. – 320 с.