Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

Навчально-науковий інститут комп'ютерних наук та інформаційних технологій
(повне найменування інституту, назва факультету (відділення))

Кафедра інформаційних систем та комп'ютерного моделювання.
(повна назва кафедри (предметної, циклової комісії))

Пояснювальна записка

до дипломної роботи

перший (бакалаврський)
(рівень вищої освіти)

на тему: Програмно-алгоритмічне забезпечення генерації ортогональних кодів

Виконав: студент 2 курсу групи ІСТСз-21
спеціальності
126 "Інформаційні системи та технології"
(шифр і назва напряму підготовки, спеціальності)

Лях В. В.
(прізвище та ініціали)

Керівники Лукащук Б.С.
(прізвище та ініціали)

Сторожук О.Л.
(прізвище та ініціали)

Рецензент Кинаш Ю.Є.
(прізвище та ініціали)

Львів – 2024

Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

ННІ комп'ютерних наук та інформаційних технологій
Кафедра інформаційних систем та комп'ютерного моделювання
Рівень вищої освіти перший (бакалаврський)
Спеціальність 126"Інформаційні системи та технології"
(шифр і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри ІСКМ
_____ Сторожук О.Л.
"06" 02 2024 року

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Ляху Володимиру Володимировичу
(прізвище, ім'я, по батькові)

1. Тема роботи Програмно-алгоритмічне забезпечення генерації ортогональних кодів

керівники роботи Лукащук Б.С., Сторожук О.Л., к.т.н., доц.,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від "6"лютого 2024 року № С-84
2. Термін подання студентом роботи 10.06.2024
3. Вихідні дані до роботи
- провести огляд алгоритмів ортогонального кодування;
- дослідити математичну модель ортогонального кодування;
- розробити програмний продукт з інтуїтивним інтерфейсом;
- провести тестування роботи розробленого програмного продукту.
4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)
Вступ
Розділ 1. Стан проблемної області
Розділ 2. Інформаційне та математичне забезпечення
Розділ 3. Програмне забезпечення та технічне забезпечення
Висновки
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
- системний аналіз;
- розробка та відображення алгоритму роботи ортогонадьного кодування;
- структура програмного рішення;
- експериментальна частина.

6. Дата видачі завдання 07 лютого 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Отримання завдання	07.02.2024	виконано
2	Огляд літературних джерел	31.03.2024	виконано
3	Розділ 1. Стан проблемної області	15.04.2024	виконано
4	Розділ 2. Інформаційне та математичне забезпечення	30.04.2024	виконано
5	Розділ 3. Програмне забезпечення та технічне забезпечення	15.05.2024	виконано
6	Оформлення пояснювальної записки	31.05.2024	виконано
7	Подання готової роботи	10.06.2024	виконано

Студент _____ (підпис) Лях В.В. (прізвище та ініціали)

Керівники роботи _____ (підпис) Лукащук Б.С. (прізвище та ініціали)

_____ (підпис) Сторожук О.Л. (прізвище та ініціали)

АНОТАЦІЯ

Дипломна робота містить 45 сторінок пояснювальної записки, 13 рисунків, 3 таблиці, 1 додаток, 26 джерел.

Розроблені методи побудови завадостійких кодів за допомогою багатопозиційних комбінаторних структур типу ідеальних кільцевих в'язанок для створення систем кодування, які виявляють та виправляють помилки, з поліпшеними якісними показниками за потужністю та завадостійкістю.

Розроблена програмна реалізація для синтезу завадостійких кодів на основі ідеальних кільцевих в'язанок, а також представлена візуалізації отриманих результатів.

Ключові слова:
ортогональне кодування, коригуючі коди, ідеальна кільцева в'язанка.

ABSTRACT

The thesis contains 45 pages of explanatory note, 13 figures, 3 tables, 1 appendix, 26 sources.

Methods for constructing jamming-tolerant codes using multi-position combinatorial structures of the type of perfect ring-blankets are developed to create error-detecting and error-correcting coding systems with improved power and jamming performance.

A software implementation for the synthesis of interference-resistant codes based on ideal circular knittings is developed, and visualizations of the obtained results are also presented.

Keywords:
orthogonal coding, corrective codes, ideal ring bundle.

ТЕХНІЧНЕ ЗАВДАННЯ

Необхідно розробити програмне та алгоритмічне забезпечення ортогонального кодування для виправлення помилок, а саме:

1. провести попередній аналіз існуючих ортогональних кодів для виправлення помилок;
2. визначити математичну модель ортогональних кодів корекції помилок;
3. визначити кількість помилок, виявлених та виправлених ортогональним кодом;
4. розробити алгоритм синтезу ортогонального коду, що виявляє та виправляє помилки;
5. інтерпретувати отримані результати;
6. розробити програмне забезпечення для представлення результатів роботи.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

АКФ	автокореляційна функція
АС	абонентська станція
БС	базова станція
ВКФ	взаємно-кореляційна функція
ІКВ	ідеальна кільцева в'язанка
СМС	системи мобільного зв'язку
ПВП	псевдовипадкова послідовність
ПКФ	потрійна кореляційна функція
CDMA	Code Division Multiple Access
GSM	Groupe Spécial Mobile

ВСТУП

Актуальність. Перешкодостійкість є однією з найважливіших характеристик сучасних систем передачі інформації. Причина його подальшого збільшення при постійній швидкості потоку є дуже актуальною проблемою.

У роботі запропоновано поряд із перешкодостійким кодуванням використовувати додаткове ортогональне кодування в системах передачі цифрових повідомлень, а якщо це неможливо, то квазіортогональне кодування. Комбіноване використання обох типів кодувань дає значну перевагу перед використанням лише стійких до помилок кодів.

При обробці отриманих кодів на приймальній стороні системи передачі інформації виділяють первинний і вторинний види обробки. Початковий тип обробки передбачає визначення значення переданого символу, а іноді також оцінку ймовірності умовної помилки. Вторинна означає виправлення помилок у пристрої декодування з використанням жорстких рішень або умовних ймовірностей помилок, отриманих під час первинної обробки. Поділ на види обробки призначений для зниження трудомісткості, а отже, і вартості прийомного обладнання. У випадках, коли надійність зв'язку повинна бути особливо високою, обидва типи обробки виконуються разом. Такий спосіб прийому називається прийомом в цілому.

Робота показує, що між першим і другим рівнями обробки можна ввести ще один рівень, щоб ще більше знизити ймовірність помилки. Імовірність помилки зменшується за допомогою ортогонального або квазіортогонального кодування. Це кодування є аналогом згорткового кодування в полі дійсного числа та має максимально можливу швидкість передачі даних (кодову швидкість). Введення додаткового рівня обробки не впливає суттєво на схеми первинної та вторинної обробки.

Об'єктом дослідження є ортогональні коди.

Предметом дослідження є метод ортогонального завадостійкого кодування.

Метою роботи є розробка та дослідження методу квазіортогонального кодування для підвищення перешкодостійкості системи передачі інформації.

Для досягнення мети в роботі вирішуються наступні завдання:

- перегляд існуючих ортогональних кодів;
- проведення системного аналізу ортогональних кодів;
- аналіз методу побудови ортогонального завадостійкого коду на основі ідеальних кільцевих в'язанок;
- аналіз методів підвищення потужності ортогональних кодів на основі ідеальних кільцевих в'язанок;
- експериментальні дослідження за допомогою розробленого програмного забезпечення.

Практичне значення використання ортогональних кодових послідовностей на основі ідеальних кільцевих в’язанок полягає в розширенні сфери дослідження цих кодових послідовностей у напрямку покращення таких показників, як криптографічна міцність і завадостійкість.

РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ

1.1 Існуючі застосування ортогональних та квазіортогональних кодів

Загальновідомо, що характерною рисою сучасного суспільства є високий рівень інформованості. Завдяки успіхам радіо та мікроелектроніки створені передові системи, про які півстоліття тому можна було тільки мріяти. У ці системи було вкладено найкращі досягнення науки та техніки за останнє століття. На щастя чи на жаль, вони багато в чому визначають якість життя сучасної людини, адже кожному доступна більшість технічних новинок. Одним із найкращих прикладів усіх типів засобів зв'язку та доступу до інформації є система мобільного радіозв'язку на основі технології CDMA, яка базується на шумоподібних сигналах [3].

Існує багато публікацій про широкосмугові системи поділу коду. Однак їх опис часто виявляється або складним, недоступним пересічному читачеві технічною мовою, або надмірно спрощеним, що зазвичай дуже шкідливо. Зрештою, в обох випадках немає сенсу розуміти принципи роботи, перспективи розвитку та загальну філософію системи. Також спантеличує або, в кращому випадку, знеохочує велика кількість англійських скорочень у літературі та описах різних «приватних» стандартів і протоколів спілкування. Тому мета цієї роботи полягає в тому, щоб з'ясувати ці питання, а також спробувати відокремити принципово важливе від неважливого, що має створити вичерпну та перевірену картину поточного стану систем радіозв'язку на основі широкосмугового та стільникового зв'язку. . зокрема зв'язок CDMA, який призначений для багатокористувацького доступу в системах зв'язку [2, 8].

Найважливішою вимогою при проектуванні систем загального користування є забезпечення зв'язку між усіма абонентами без взаємних перешкод. Фізичним носієм інформації в системах мобільного зв'язку (СМС) є радіосигнал. Усі користувачі створюють складний єдиний електромагнітний процес у точці прийому. Розробники СРЗ повинні визначити, за яким критерієм інформація від конкретного користувача буде виділена із загального радіосигналу. Тому розглядаються п'ять «чистих» методів поділу сигналів [19]:

1. частотний;
2. часовий;
3. поляризаційний;
4. просторовий (по напряму приходу сигналу);
5. кодовий (формою, по вигляду сигналу).

Давайте розглянемо їх ближче. Поняття частотного поділу каналу тісно пов'язане з поняттям спектра сигналу, тобто з частотним представленням його часової реалізації. Виявляється, майже кожен радіосигнал є вузькосмуговим процесом, для якого має місце нерівність F/f0<<1. У цій нерівності F — ефективна смуга частот, яку займає сигнал, f0 — середня (центральна) частота спектру. Розташувавши сигнали від різних джерел на частотній осі так, щоб вони не накладалися, використовуючи пристрої частотної фільтрації з ідеальними характеристиками, можна розділити інформаційні сигнали, не заважаючи один одному. Цей метод є найвідомішим і почав використовуватися на зорі розвитку радіотехніки під час Першої світової війни, після того, як іскрові передавачі були замінені передавачами безперервної генерації з використанням вакуумних трубок. Згідно з історичними даними, піонерами в цьому напрямку були німці, що, до речі, призвело до невеликого замішання в розвідці їхніх супротивників, коли одного разу замість звичного лепету електричних розрядів у повітрі вони почули тиша.

У 1980-х роках аналогові СРЗ першого покоління були побудовані на основі частотного поділу: північноєвропейський мобільний телефон (NMT), американська розширена мобільна телефонна служба (AMPS), англійська система зв'язку повного доступу (TACS) та деякі інші [9, 10, 15, 16].

Ідея тимчасового поділу каналів з'явилася трохи пізніше частотного, але її реальне застосування почалося з розвитком імпульсної техніки в 1950-1960-х роках. Суть методу теж дуже проста. Якщо говорити про аналогові сигнали, то виявилося, що повідомлення можна передавати за допомогою послідовності імпульсів або вибірок з паузами в передачі. Амплітуда імпульсів повинна відповідати миттєвим значенням сигналу в момент відбору проби, а в перервах проби можуть надсилатися з інших джерел. Мінімальна частота дискретизації визначається теоремою

В.А. Котельникова і дорівнює подвоєній максимальній частоті спектра повідомлення. Жодних умов щодо тривалості пульсу не висувається. Стосовно цифрового SRZ, часовий поділ каналу означає передачу кожного бінарного потоку у власному часовому вікні. Найвідомішою СРЗ з часовим поділом каналів є Європейська глобальна система мобільного зв'язку (GSM), в якій на оператора припадає 8 (16) цифрових каналів [12].

Поляризаційне розділення сигналів засноване на вибірковості приймальних антен щодо положення вектора напруженості електричного (магнітного) поля - вертикального (V), горизонтального (H) - або напрямку його обертання - правого (R), ліворуч (L). Зі сказаного вище можна зробити висновок, що для лінійної поляризації на одній частоті і в один і той же час можна виділити максимум чотири канали: поляризація V/H електричної складової та поляризація V/H компоненти магнітного поля. Така ж ситуація виникає при використанні сигналів з круговою поляризацією. Мультиплексування каналів через поляризацію в основному реалізовано в супутникових і мовних системах зв'язку. Наприклад, в супутниковому телебаченні спектри сигналів сусідніх телеканалів значно перекриваються, але передаються з різною поляризацією і надійно розділяються на приймальному кінці.

Просторове розділення досягається при русі від багатьох різних напрямків до одного або у зворотному порядку. В основі цього методу поділу каналів лежать властивості спрямованості передавальної та приймальної антен. Отже, вибірковість антени по азимуту (куту місця) дозволяє розділяти сигнали з однаковою середньою частотою, поляризацією і одночасно, але надходять в різних напрямках в горизонтальній (вертикальній) площині. Це явище використовується в переважній більшості радіоелектронних систем. Що стосується СРЗ, то просторове розділення вхідних сигналів реалізовано у вигляді багатосекторних антен базових станцій, що дозволяє повторювати частоти прийому (передачі) в різних секторах.

Ідея кодового поділу (стиснення) каналів відома близько 70 років і заснована на кореляційних властивостях сигналів. Його розвиток бере свій початок з 1960-х років, коли були розроблені відносно швидкі електронні логічні схеми і пристрої для розбиття коду досягли прийнятних розмірів. Цей новий метод вперше був

використаний в радіостанціях армії США. Використаний широкосмуговий код був «стрибком» радіочастотного сигналу за певним законом. Така реалізація значно ускладнювала прослуховування розмов і характеризувалася високими якісними показниками.

В англомовній літературі системи з канальним поділом каналів називаються кодовим розділенням (CDMA), що дослівно перекладається на українську як кодовий множинний доступ. Правильне розуміння стиснення сигналу через форму (код) вимагає введення поняття кореляції сигналу. Взаємно-кореляційна функція (ВКФ) R12(τ) сигналів s1(t) і s2(t) зі скінченними енергіями називається функцією, заданою виразом [37]:

$$R_{12}(\tau) = \int_{-\infty}^{+\infty} s_1(t) s_2(t-\tau) dt \qquad (1.1)$$

де t - час;

τ- величина зміщення в часі другого сигналу відносно першого.

Основним фізичним значенням ВКФ є ступінь подібності між двома сигналами. Окремим випадком ВКФ є автокореляційна функція (АКФ), коли s1(t)=s2(t). Чим більше подібні сигнали один до одного, тим позитивніше значення PCF. Якщо значення функції R12(τ) має найбільше за модулем значення і негативний знак, то говорять, що сигнали s1(t) і s2(t) протилежні, тобто s1(t)=-s2(t). Для поділу кодового каналу, який використовується в стандартах CDMA, важливий третій випадок, коли R12(τ)=0 в точці τ=τ0 або R12(τ)≈0 на всьому інтервалі визначення зсуву τ. Сигнали, що задовольняють першу рівність, називаються ортогональними «в точці», а ті, що задовольняють другій наближеній рівності, — квазіортогональними. Немає сигналів, для яких PCF дорівнює строго нулю за всіх часових зсувів, тому далі, коли ми говоримо про ортогональні коди, ми матимемо на увазі ортогональний «в точці». У зв'язку з цими випадками розглядаються два типи систем адресації з мультиплексуванням кодового каналу: синхронна і асинхронна. СРЗ з мультиплексуванням кодового каналу, засновані на стандарті IS-95, є системами синхронної адресації.

Ідея мультиплексування кодового каналу на прикладі низхідної лінії зв'язку, тобто від базової станції (БС) до абонентської станції (АС), зводиться до наступного:

1. N інформаційним потокам, призначеним для N абонентів, призначається власна псевдовипадкова послідовність (ПВП);
2. кодові послідовності не корельовані;
3. двійкові інформаційні потоки модулюються власними ПВП;
4. коди широкосмугового каналу додаються до суматора;
5. результуюча модуляція зі складним широкосмуговим сигналом і випромінюванням радіосигналу в космос.

На приймальній стороні в абонентській станції:

1. відома «своя» кодова послідовність;
2. сигнал передається з радіочастоти в область низьких частот;
3. на вхід корелятора подається низькочастотний імпульсний сигнал, на другий вхід якого синхронно надходить кодування ПВП;

   корелятор, що складається з множника та інтегратора, обчислює взаємну кореляційну функцію обох кодів;
4. зворотній зв'язок на виході корелятора виникає тільки тоді, коли стиснутий композитний сигнал містить «власний» PVP, інакше на виході спостерігається тільки шум.
5. стандарт IS-95 є основою для існуючих стандартів супутникових систем персонального зв'язку, а також мобільного стільникового зв'язку, включаючи: CDMA2000 і WB-CDMA. Кількість каналів N визначається розміром набору псевдовипадкових послідовностей.

В даний час відомо багато методів генерації наборів ортогональних і квазіортогональних послідовностей. Серед систем ортогональних сигналів у стандарті IS-95 використовуються коди, які є рядками матриці Адамара 64x64. Матриця Адамара розміром 2nx2n створюється з матриці розміром nxn таким чином [40]:

$$H_{2n} = \begin{bmatrix} H_n & H_n \\ H_n & -H_n \end{bmatrix} \quad (1.2)$$

Початковою матрицею в цьому рекурентному обчисленні виступає матриця розміром 1X1: H1=[1]. Таким чином, матриця Адамара розміром 2X2.

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (1.3)$$

Аналогічно матриця 8X8 виглядає так:

$$H_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \quad (1.4)$$

Ви можете перевірити, що якщо помножити елементи двох різних рядків матриці попарно, а потім додати результати, ви отримаєте нуль. Це означає, що будь-яка пара рядків в матриці Адамара є ортогональною (звичайно, якщо немає взаємного зсуву). Крім того, співвіднесення послідовності з самою собою дає число 8, що очевидно. Якщо виправити рядок і його зворотне представлення, результат буде -8. Тому інформацію «0» першого каналу можна передати з першим рядком матриці Адамара, а інформацію «1» з першим рядком, але з інверсією; двійковому потоку другого каналу можна призначити другий рядок символів і так далі. Оскільки рядки є ортогональними матрицями, сигнали різних каналів можуть бути розділені на приймальному кінці. Рядки матриці Адамара частіше називають функціями Уолша.

Конкретний пристрій додавання в принципі може бути пристроєм додавання алгебри, тоді результуючий складний код буде багаторівневим, у нашому випадку послідовністю 2N рівнів. Однак прийом і обробка таких кодів у реальних каналах зв'язку пов'язана зі значними труднощами, тому CDMA приймає більшість підсумовування кодів каналів. Мажоритарний суматор двійкових послідовностей обчислює знак алгебраїчної суми канальних кодів. Для обчислення такої

багатовимірної функції існують спеціальні мажоритарні логічні схеми підсумовування.

Кореляційна функція третього порядку та біспектр широко використовуються в цифровій обробці сигналів для виявлення та відновлення негаусових сигналів, що спостерігаються в адитивному гаусовому шумі в радіолокаційних системах [1-3], астрономії [4], біомедичних пристроях обробки сигналів [5] та багато інших застосувань [6].

Потрійна кореляційна функція (ПКФ) і біспектр мають унікальні властивості. Зокрема, ПКФ і біспектр дозволяють реконструювати Фур'є-фазовий спектр сигналу і оцінювати поведінку спектральних компонент, що мають фазові зв'язки. Важливою властивістю ПКФ і біспектру є висока стійкість до адитивного шуму від симетричної функції щільності ймовірності.

Основною причиною підвищення ймовірності помилок в системах радіозв'язку є наявність електричних перешкод.

Надійність p_e появи некоректного біта при детектуванні двійкових сигналів у гаусовому шумі за критерієм максимальної надійності прийому сигналу визначається за відомою формулою виду [7]

$$p_e \le \frac{M}{2} Q\left(\sqrt{\frac{E_w(1-\rho)}{N_0}}\right) = \frac{M}{2} Q\left(\sqrt{\frac{P_w(1-\rho)}{\sigma^2}B}\right), \qquad (1.5)$$

де $Q(x)$ – інтеграл помилок Гауса;

$M=2^k$ – розмір множини кодових слів;

k - число інформаційних біт в кодовому слові;

$E_w = k\,E_b$, $P_w = kP_b$, E_b и P_b – енергія, потужність кодового слова, енергія і потужність біта, відповідно;

N_0 – спектральна щільність потужності шуму;

σ^2 – дисперсія шуму;

$B=FT_b$ – база сигналу, рівна твору робочої смуги частот F на тривалість біта T_b;

ρ – коефіцієнт взаємної кореляції двійкових сигналів $s_1(t)$ $s_2(t)$, рівний $\rho=\frac{1}{E_b}\int_0^{T_b} s_1(t)s_2(t)dt$.

Завадостійкість системи зв'язку при кореляційному прийомі простої бітової послідовності (M=2, k=1) відповідно до (1.1) може бути покращена при збільшенні бази сигналу B. Шумоподібні сигнали [8] з великою базою знаходять застосування в широкосмугових системах зв'язку саме завдяки можливості придушення завади в B разів. Однак використання шумоподібних кодів неминуче передбачає розширення діапазону робочих частот або збільшення тривалості біта.

Іншим поширеним способом підвищення якості зв'язку є блокове кодування [7] у вигляді класу надлишкових перетворень інформаційного потоку, внаслідок чого процес виявлення є менш ефективним. У цьому випадку інформаційний потік бітів перетворюється в послідовність слів (блоків) довжиною k кожне, створюючи набір з M = 2k кодових слів. Нова вдосконалена послідовність містить надлишкові біти для виявлення та виправлення помилок. Перешкодостійке кодування використовується в системах зв'язку, де запити на повторну передачу неможливі або неможливі, або де рівень перешкод настільки високий, що необхідна дуже велика кількість повторних сеансів передачі інформації.

Таким чином можна підвищити стійкість системи зв'язку до перешкод:

- збільшення бази (довжини, потужності) кодування з використанням шумоподібних (широкосмугових) кодів;
- вживання ортогональних та квазіортогональних кодів;
- використання надлишкового кодування для визначення і виправлення помилок.

Метою роботи є підвищення стійкості системи зв'язку до перешкод на основі використання надлишкового кодування, побудованого із вибірок квазіортогональних кодів (максимально помітних кодових слів).

РОЗДІЛ 2. ІНФОРМАЦІЙНЕ ТА МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

2.1 Існуючі методи кодування і декодування за допомогою ортогональних та квазіортогональних кодів

2.1.1 Алгоритм побудови надлишкових кодів

Припустимо, що оригінальне буквено-цифрове повідомлення, визначене послідовністю чисел і символів, що належать до стандартного комп'ютерного алфавіту ANSI, що складається з 8-розрядних двійкових символів, має бути передано по каналу радіозв'язку. Алгоритм генерації запропонованих надлишкових кодів включає набір наступних процедур [31].

1. Вводяться дві нові ортогональні послідовності a=(00000000) і b=(22222221) завдовжки, дорівнює 8 кожен, послідовність a відповідає двійковому символу «нуль», а послідовність b відповідає символу «один» для потоку бітів ANSI у вихідному повідомленні. Кількість позицій, у яких послідовності a і b відрізняються одна від одної, або відстань Хеммінга дорівнює d(a,b)=8.

2. Виконуються розрахунки ПКФ вище введених послідовностей ***a*** і ***b*** у вигляді:

$$R_a(l,m) = \sum_{n=1}^{8} a(n)a(n-l)a(n-m), \tag{2.1}$$

$$R_b(l,m) = \sum_{n=1}^{8} a(n)a(n-l)a(n-m), \tag{2.2}$$

де $l = 1,2,\ldots,8$,

$m = 1,2,\ldots,8$ індекси зрушень.

Результати розрахунків ПКФ (2.1) і (2.2) представлені на рис. 2.1.

```
54 52 52 52 54          0 0 0 0 0
52 52 52 52 54 52       0 0 0 0 0 0
52 52 52 52 54 52 54    0 0 0 0 0 0 0
52 52 52 52 54 54 52 52 0 0 0 0 0 0 0 0
54 54 54 54 57 54 54 54 54 0 0 0 0 0 0 0 0
   52 52 54 54 52 52 52 52    0 0 0 0 0 0 0 0
      54 52 54 52 52 52 52       0 0 0 0 0 0 0
         52 54 52 52 52 52          0 0 0 0 0 0
            54 52 52 52 54             0 0 0 0 0
```

Рисунок 2.1 – ПКФ послідовностей *a* і *b*

Аналіз величин, представлених на рис. 2.1, демонструє симетричність відліків ПКФ, розташованих в межах шестикутної області.

3. Перетворення десяткових відліків ПКФ $R_a(l,m)$ і $R_b(l,m)$ у двійковий 6-розрядний код. В результаті формують надлишкові бінарні ПКФ- коди $C(l,m)$ і $D(l,m)$, складені з двійкових 6-розрядних відліків ПКФ $R_a(l,m)$ і $R_b(l,m)$.

Надлишкові ПКФ- коди $C(l,m)$ і $D(l,m)$, складені з 6-розрядних слів $R_a(l,m)$ і $R_b(l,m)$, запишемо у вигляді:

$$C(l,m) = \begin{pmatrix} R_a(1,1),\ldots,R_a(1,8) \\ R_a(2,1),\ldots,R_a(2,8) \\ \ldots\ldots\ldots\ldots \\ R_a(8,1),\ldots,R_a(8,8) \end{pmatrix}, \tag{2.3}$$

$$D(l,m) = \begin{pmatrix} R_b(1,1),\ldots,R_b(1,8) \\ R_b(2,1),\ldots,R_b(2,8) \\ \ldots\ldots\ldots\ldots \\ R_b(8,1),\ldots,R_b(8,8) \end{pmatrix}. \tag{2.4}$$

4. Блокові коди (2.3) і (2.4) використовуються для правильного кодування кожного біта вихідного буквено-цифрового 8-бітового символу ANSI. В результаті кодування вихідного потоку бітів виходить нове повідомлення, що складається з потоку 6-бітних слів. Кожне слово відповідає значенню лічильника PCF (2.3) або (2.4). Максимальна кількість 6-розрядних слів у цьому наборі становить 64.

2.2 Характеристика об'єкта дослідження

Ортогональність — поняття, яке є узагальненням перпендикулярності лінійних просторів із введенням скалярного добутку.

Якщо скалярний добуток двох елементів простору дорівнює нулю, ми говоримо, що вони ортогональні один одному.

Важливою особливістю поняття є його зв'язок із конкретним скалярним добутком, який використовується. При зміні добутку ортогональні елементи можуть виявитися неортогональними і навпаки.

Залежно від способу створення та статистичних властивостей ортогональні кодові послідовності поділяються на істинно ортогональні та квазіортогональні. Відмітна ознака послідовності - коефіцієнт взаємної кореляції ρ_{ij}, який в загальному випадку змінюється від -1 до +1.

У теорії кодів доведено, що граничне досяжне значення коефіцієнта взаємної кореляції визначається з умови [2, 3]:

$$\rho_{ij} = \begin{cases} -1/N, \text{ де } N - \text{непарне} \\ -1/(N-1), \text{ де } N - \text{парне} \end{cases} \quad (2.5)$$

Мінімальне значення TCF забезпечує коди, в яких коефіцієнти кореляції для будь-яких пар рядків негативні (трансортогональні коди). Коефіцієнт крос-кореляції ортогональних послідовностей, за визначенням, дорівнює нулю, тобто $\rho_{ij} = 0$. При великих значеннях N різницею між коефіцієнтами кореляції ортогонального і трансортогонального кодів можна практично знехтувати. Шкода, однак, що ортогональні коди існують не завжди для всіх довжин, тому завдання зводиться до пошуку методів синтезу квазіортогональних кодів, тобто коли коефіцієнт крос-кореляції мінімальний $\rho_{ij} \to \min$.

2.3 Базові поняття квазіортогональних послідовностей

Розробка системи CDMA раніше була більше мистецтвом, ніж наукою. Інженери вибирали сигнали, використання яких має покращити основні характеристики систем (якість зв'язку, стійкість до перешкод), спираючись виключно на свою інтуїцію. Моментом повернення стало створення теорії створення, обробки та передачі кодів. Він дозволяє визначити ефективність використання конкретного набору (набору) кодів, спираючись виключно на знання їх авто- та інтеркореляційних характеристик.

Кодові послідовності, що використовуються в системах CDMA для передачі коду, складаються з елементарних N символів (чіпів). Кожен кодовий інформаційний символ складається з однієї N-символьної послідовності, яка називається «розширювальною послідовністю», оскільки «результуючий» код випромінюється в

повітря з навмисно розширеним спектром. Виграш, наприклад, у разі проблем зв'язку залежить як від кількості елементів (чіпів, довжини) кодової послідовності, так і від характеристик набору кодів, насамперед від їх взаємних кореляційних властивостей і способу модуляції.

Розглянемо довжину кодової послідовності. У вітчизняній літературі сигнали (побудовані на основі цих кодових послідовностей), базис яких набагато більше одиниці (B=TF>>1, де T - тривалість елемента сигналу, F - смуга частот) , зазвичай називають складними. Порівняно з вихідним (інформаційним) складеним сигналом (кодовою послідовністю) присутній шум майже з такою ж спектральною щільністю потужності.

Відомо, що чим розширеніший спектр сигналу в повітрі, тим менша його спектральна щільність. Завдяки цій властивості сигнали з великою базою можуть використовуватися в «чужому» (вже зайнятому) діапазоні частот «на вторинній основі», практично не впливаючи на операційну систему.

Розглянемо характеристики квазіоптимальних кодових послідовностей. Весь набір кодових послідовностей, що використовуються в CDMA, поділяється на два основні класи: ортогональні (квазіортогональні) послідовності та псевдовипадкові послідовності з низькою взаємною кореляцією (PRS).

В оптимальному приймачі CDMA кодові послідовності, що надходять на його вхід, є, по суті, адитивним білим гаусовим шумом і завжди обробляються за допомогою методів кореляції. Тому процедура пошуку зводиться до пошуку кодової послідовності, яка максимально співвідноситься з індивідуальним кодом абонента. Кореляція між двома послідовностями {x(t)} і {в(t)} здійснюється шляхом множення однієї послідовності на зміщену в часі копію іншої. Залежно від типу послідовності системи CDMA використовують різні методи кореляції:

- автокореляція, якщо псевдовипадкові послідовності множення виглядають однаково, але зміщені в часі;
- взаємний, якщо PVP мають різні типи;
- періодичні, якщо зміни між двома ПВП є циклічними;
- аперіодичні, якщо зміни не циклічні;

- для частини періоду, якщо результатом множення є лише сегменти двох послідовностей заданої довжини.

Щоб отримати переваги ідентифікації та виправлення пошкоджених кодів у каналі зв'язку під час використання будь-якого методу обробки кореляції, необхідно, щоб набір кодів мав «хороші» властивості автокореляції. Бажано, щоб кодова послідовність мала один пік автокореляції, інакше можлива неправильна синхронізація за допомогою інших функцій автокореляції (ACF). Зверніть увагу, що чим довша довжина кодової послідовності, тим кращі параметри функції автокореляції.

Пари кодових послідовностей вибираються так, щоб функція крос-кореляції (MCF) мала мінімальне значення для їх попарної кореляції. Це гарантує мінімальний рівень взаємних перешкод.

Отже, вибір оптимального набору (множини) кодових послідовностей зводиться до пошуку такої структури кодових послідовностей, в якій центральний пік АКФ має найвищий рівень, а залишкові максимальні викиди АКФ максимально мінімальні.

2.4 Існуючі методи побудови ортогональних та квазіортогональних кодових послідовностей

Існує кілька способів генерації ортогональних кодових послідовностей. Найчастіше використовуються послідовності Уолша завдовжки 2^n, які утворюються на основі рядків матриці Адамара [10]:

$$H_{2n}=\begin{bmatrix} H_n H_n \\ H_n - H_n \end{bmatrix}. \qquad (2.6)$$

Принцип формування цієї матриці досить простий; його пояснює рисунок 2.2.

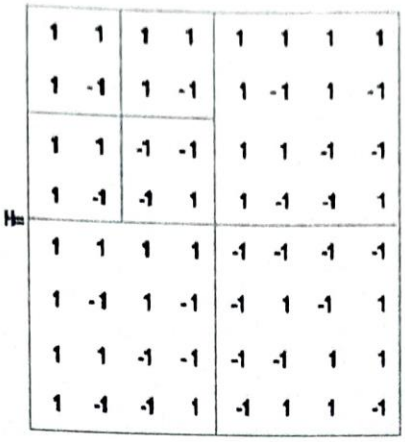

Рисунок 2.2 – Побудова кодових послідовностей Уолша

Початковим є код вигляду H1={1}. Підставляючи його в матрицю H_{2n}, отримуємо нову матрицю більшого розміру:

$$H_2=\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{2.7}$$

Багаторазове повторення процедури дозволяє створити матрицю будь-якого розміру, яка характеризується взаємною ортогональністю всіх рядків і стовпців.

Цей метод генерації кодових послідовностей реалізований у стандарті IS-95, де довжина послідовності Уолша дорівнює 64. Зауважимо, що різниця між рядками матриці Адамара та послідовностями Уолша полягає лише в тому, що в останніх використовуються уніполярні типи коду { 1,0}.

На прикладі матриці Адамара легко проілюструвати принцип побудови трансортогональних кодів. Так, ви можете бути впевнені, що якщо ви видалите з матриці перший стовпець лише одиниць, ортогональні коди Уолша будуть перетворені на трансортогональні коди, де для будь-яких двох рядків кількість невідповідностей символів перевищує кількість збігів рівно на одиницю, тобто:

$$\rho_{ij} = -1/(N-1). \tag{2.8}$$

Іншим важливим типом ортогональних кодів (кодових послідовностей) є біортогональний код, який створюється з ортогонального коду та його інверсії. Головною перевагою біортогональних кодів у порівнянні з ортогональними кодами є можливість передачі коду в половині довжини кодової послідовності (менша смуга

частот). Наприклад, біортогональний блоковий код (32, 6), що використовується в WCDMA, дозволяє передавати сигнал у транспортному форматі TFI.

Слід зазначити, що ортогональні кодові послідовності мають два основних недоліки [17, 19].

1. Максимальна кількість можливих рядків коду обмежена їх довжиною (у стандарті IS-95 довжина рядка коду становить 64), тому вони мають обмежений адресний простір.

Для розширення набору кодових послідовностей поряд з ортогональними послідовностями використовуються квазіортогональні послідовності. Таким чином, новий стандарт cdma2000 пропонує метод генерації квазіортогональних кодових послідовностей шляхом множення послідовностей Уолша за допомогою спеціальної функції маскування. Цей метод дозволяє за допомогою однієї такої функції отримати набір квазіортогональних послідовностей Quasi-Orthogonal Function Set (QOFS). За допомогою m маскуючих функцій і ансамблю кодових послідовностей Уолша завдовжки 2^n можна створити (m+1) 2^n QOF- послідовностей.

2. Іншим недоліком ортогональних кодових послідовностей (і тих, що використовуються в стандарті IS-95 не є винятком) є те, що взаємна кореляційна функція дорівнює нулю лише «в певній точці», тобто за відсутності зсуву в часі між кодами. . Тому такі сигнали використовуються тільки в синхронних системах і переважно в прямих каналах (від базової станції до абонента).

Можливість адаптації системи CDMA до різних швидкостей передачі досягається за допомогою спеціальних ортогональних рядків зі змінним коефіцієнтом розширення (OVSF), званих кодами змінної довжини. При передачі CDMA-сигналу, створеного з використанням такої послідовності, швидкість чіпа залишається постійною, але швидкість інформації змінюється в два рази. Стандарти третього покоління пропонують використання ортогональних багатошвидкісних кодів Голда як кодів OVSF. Принцип їх синтезу досить простий; Копати. пояснює це. 3, де показано дерево кодів, що дозволяє будувати коди різної довжини.

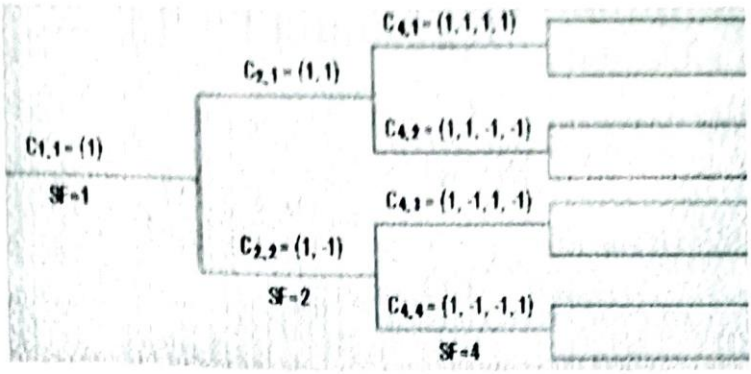

Рисунок 2.3 – Побудова ортогональних кодів Голда

Розглянемо кодове дерево для генерації кодів OVS (SF – spreading factor). Кожен рівень кодового дерева визначає довжину кодових слів (коефіцієнт розсіювання, SF), при цьому кожен наступний рівень подвоює можливу кількість кодів. Таким чином, якщо тільки два кодових слова можуть бути згенеровані на рівні 2 (SF=2), то на рівні 3 генеруються чотири кодові слова (SF=4) і так далі. Повне дерево коду містить вісім рівнів, що відповідають SF=256 (на малюнку 3.2 показано лише три нижні рівні).

Таким чином, кодовий набір OVSF не є фіксованим: він залежить від коефіцієнта розширення SF, тобто, власне, від швидкості каналу.

Зауважте, що не всі комбінації дерева коду можна реалізувати одночасно в одній комірці CDMA. Головною умовою вибору комбінації є неприпустимість порушення їх ортогональності.

2.5 Побудова квазіоптимальних кодових послідовностей на основі псевдовипадкових послідовностей (ПВП)

На додаток до ортогональних кодів, PVP відіграють ключову роль у системах CDMA, які, хоча й генеруються детерміновано, мають усі властивості випадкових сигналів. Однак вони вигідно відрізняються від ортогональних послідовностей тим, що вони інваріантні до часових зрушень. Існує кілька типів PVP з різними характеристиками. Простіше кажучи, сьогодні з'явилися технічні засоби, які можуть «вивести» практично будь-який набір кодових послідовностей із заданими властивостями [30, 31, 34].

Розглянемо m-послідовності. Одним із найпростіших і надзвичайно ефективних способів генерації детермінованих двійкових рядків є використання регістра зсуву (SR). Послідовність на виході n-розрядного реле зворотного зв'язку завжди періодична, а її період n (кількість циклів, після яких схема повертається у вихідний стан) не перевищує 2n.

Теоретично, використовуючи n-розрядний регістр і відповідно обрану логіку зворотного зв'язку, ви можете отримати рядок будь-якої довжини N в межах x від 1 до 2^n включно. Послідовність максимальної довжини, або m-послідовність, матиме період 2^n-1.

Функція автокореляції m-послідовності є періодичною і двозначною:

$$\rho_{ij} = \begin{cases} N \text{ при } I=j, \\ -1 \text{ при } I \neq j \end{cases}.$$

Рисунок 2.4 – Автокореляційна функція для m-послідовності:

а) аперіодична, б) періодична

Рівень бокових максимумів автокореляційної функції (рис. 3.3) не перевищує значення

$$\rho_{ij}=1/\sqrt{N}.$$

Потім розглянемо кодові послідовності Голда та Касамі. Псевдовипадкові послідовності Gold і Kasami найчастіше використовуються в системах CDMA, забезпечуючи низькі випромінювання PCF. Послідовності золотого коду з періодом 2n-1 створені на основі двох m-послідовностей з виділенням т.зв. бажані пари.

Золоті коди створюються шляхом поєднання двох символів m-послідовностей за символом за модулем 2 (рис. 3.4). Конструкція WCDMA визначає три типи кодів Gold: первинний і вторинний ортогональний код Gold (обидва по 256 біт) і довгий код.

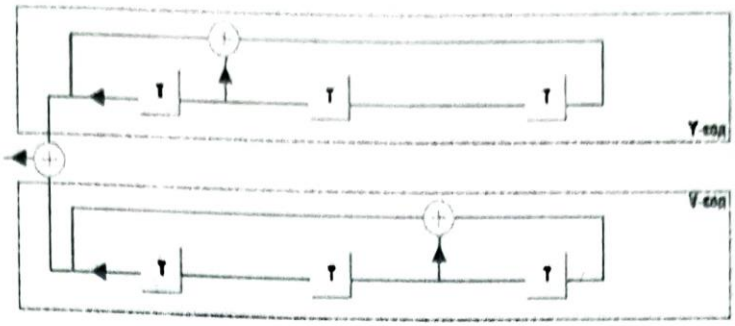

Рисунок 2.5 – Генератор коду Голда

(Т - елемент регістра зрушення; & - схема збігу; + - суматор по модулю 2)

Ортогональні коди Голда створюються на основі 255-бітної послідовності m з додаванням одного надлишкового символу. Базовий код синхронізації має функцію аперіодичної автокореляції та використовується для початкової синхронізації. Вторинний код синхронізації — це немодульований ортогональний код золота, який передається паралельно з основним кодом синхронізації. Кожен додатковий код синхронізації вибирається з 17 різних золотих кодів {C1...,C17}.

Код довгого прямого каналу - це фрагменти коду Голда довжиною 40960 токенів. Система зв'язку на основі WCDMA є асинхронною, із сусідніми базовими станціями, які використовують різні коди Голда (загалом 512), які повторюються кожні 10 мс. Асинхронний принцип роботи базових станцій робить їх незалежними від зовнішніх джерел синхронізації. Передбачається, що довгий код використовується в зворотному каналі, але тільки в тих комірках, де не включений багатокористувацький режим виявлення.

Сімейство кодових послідовностей Касамі містить 2^{κ} послідовностей з періодом 2^{n-1}. Вони вважаються оптимальними в тому сенсі, що для будь-якої «переважної» пари забезпечується максимальне значення автокореляційної функції, рівне $(1+2^{\kappa})$.

Кодові послідовності Kasami реалізуються за допомогою трьох послідовно з'єднаних регістрів зсуву (u, v і w) з різними зворотними зв'язками (рис. 3.5), кожен з яких створює свою власну m-послідовність. Щоб отримати кодові послідовності Kassami із заданими властивостями, послідовності v і w повинні мати різні зсуви.

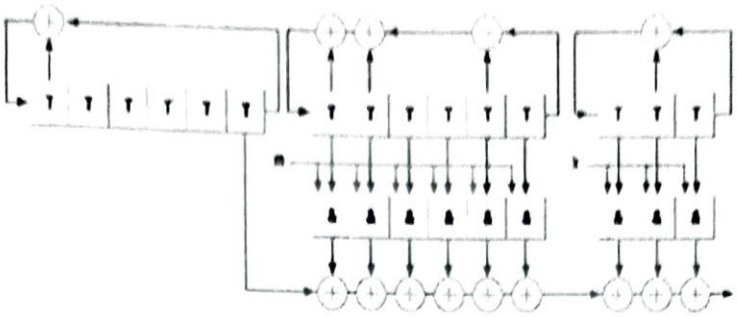

Рисунок 2.6 – Генератор кодових послідовностей Касамі типа kas (6, m, до), де m і до - циклічні зрушення v- і w-кодов відповідно (див. умовні позначення до рис. 2.4)

Коди Kasam довжиною 256 біт використовуються як короткі послідовності у зворотному каналі (дизайн WCDMA) у комірках, де використовується виявлення кількох користувачів.

Ми також розглянемо найбільш часто використовувані послідовності коду Баркера. Псевдовипадкові послідовності з низьким аперіодичним значенням ACF здатні забезпечити синхронізацію переданих кодів, прийнятих за досить короткий час, зазвичай дорівнює довжині самої послідовності. Найпоширенішими є кодові послідовності Баркера (див. табл. 2.1).

Ефективність кодових послідовностей з аперіодичною АКФ зазвичай оцінюють за допомогою індексу якості F, що визначається як відношення квадратів синфазних компонентів сигналу на основі кодових послідовностей Баркера до суми квадратів його розбіжностей. компоненти фази. Тому мірою ефективності аперіодичної кореляції двійкової послідовності є показник якості [34].

Таблиця 2.1 – Кодові послідовності Баркера

Структура послідовності Баркера (N=7, 11, 13)		
N	Вигляд послідовності	Показник якості
7	1110010	9,85
11	11100010010	12,1
13	1111100110101	14,08

Вибір псевдовипадкової кодової послідовності в системах передачі інформації дуже важливий, оскільки її параметри визначають поліпшення обробки системи, її стійкість до перешкод і чутливість до спотворень. При однаковій довжині кодової послідовності параметри системи можуть відрізнятися.

Системи, що використовують складні шумоподібні сигнали, використовуються вже понад 50 років. Відомі такі переваги шумоподібних кодових послідовностей, як висока стійкість до потужних вузькосмугових перешкод, можливість розділяти абонентів за кодом, секретність передачі, висока стійкість до багатопроменевого поширення і навіть висока роздільна здатність при вимірюванні в різних радарах і навігаціях, системах зв'язку та локації.

РОЗДІЛ 3. ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1 Дослідження властивостей квазіоптимальних послідовностей

Важливим параметром системи, що використовує оптимальні та квазіоптимальні кодові послідовності, є приріст обробки. Підсилення обробки (PG) показує ступінь поліпшення відношення сигнал/шум при перетворенні отриманого приймачем коду шуму в необхідний інформаційний сигнал. Ця процедура називається стисненням або розтягуванням. За класичним визначенням ВО дорівнює [35, 48]:

$$ВО=10Lg[Ск/Сі], \quad (3.1)$$

де Ск – частота дотримання чіпів псевдовипадковій послідовності, чіп/секунду.

Сі – швидкість передачі інформації, біт/секунду.

Згідно з цим визначенням, система зі швидкістю передачі даних 1 Мбіт/с і швидкістю відповідності мікросхеми 11 Мчіп/с (це означає, що кожен біт інформації закодовано в псевдовипадковій послідовності з 11 біт) матиме рівень 10,41 дБ. Цей результат означає, що робота системи передачі інформації залишиться на тому ж рівні BER, якщо зменшити корисний сигнал на вході на 10,41 дБ.

У звичайних комерційних чутливих до перешкод радіомодемах, таких як Arlan, Wavelan тощо, більше значення часто надається швидкості передачі інформації, ніж приховуванню чи стійкості до перешкод. Оскільки інструкції Федеральної комісії зі зв'язку США (FCC) передбачають мінімальне значення 10 дБ для таких пристроїв, а також виділяють мінімально допустиму смугу частот одного каналу (встановлюючи обмеження на максимальну частоту, сумісну з мікросхемами SK), довжина Псевдовипадкова кодова послідовність повинна складати не менше 11 маркерів на біт. Якщо збільшити довжину кодової послідовності до 64 чіпів на біт (це максимально можлива довжина для відомого процесора ShPS Z87200 Zilog), то при тій же частоті чипа 11 Мчип/с виграш обробки буде 10Lg(64)=18,06 дБ, швидкість передачі інформації при цьому зменшиться в 64/11=5,8 разів.

Щоб кодові послідовності використовувалися в системі NPS, вони повинні мати певні математичні та інші властивості, основними з яких є дуже хороші властивості

автокореляції та крос-кореляції. Крім того, кодова послідовність повинна бути добре збалансованою, тобто кількість одиниць і нулів у ній має відрізнятися не більше ніж на один символ. Ця остання вимога важлива для усунення компонента фіксованого інформаційного коду, як постулював Голомб [36].

Приймач DSSS порівнює отриману кодову послідовність із точною копією, яка зберігається в пам'яті. При виявленні кореляції між ними він переходить в режим прийому інформації, встановлює синхронізацію і починає операцію декодування корисної інформації. Будь-які часткові кореляції можуть призвести до помилкових спрацьовувань і збоїв у роботі приймача, тому кодова послідовність повинна мати хороші кореляційні властивості. Давайте детальніше розглянемо поняття кореляції.

Кореляційні характеристики кодових послідовностей, що використовуються в системах ШПС, залежать від типу кодової послідовності, її довжини, частоти символів і посимвольної структури.

Загалом автокореляційна функція (АКФ) визначається інтегралом:

$$Y(t)=\int f(t)f(t-\phi)dt \tag{3.2}$$

і показує зв'язок сигналу з його копією, зміщеною в часі на величину f. Тестування АКФ відіграє важливу роль у виборі кодових послідовностей з точки зору найменшої ймовірності встановлення некоректної синхронізації.

З іншого боку, функція крос-кореляції (ФКК) має велике значення в системах кодового поділу, таких як CDMA, і відрізняється від АКФ лише тим, що різні функції, а не ті самі, лежать під інтегральним знаком [9, 28]:

$$Y(t)=\int f(t)g(t-\phi)dt \tag{3.3}$$

Тому ФКК показує ступінь узгодженості однієї кодової послідовності з іншою. Щоб спростити концепції АКФ і ВКФ, ми можемо представити значення конкретної функції як різницю між кількістю збігів A і B невідповідностей символів кодової послідовності під час їхнього порівняння символ за символом. Щоб проілюструвати цей приклад, розглянемо функцію автокореляції послідовності коду Баркера з 11 маркерів, яка має наступний вигляд [28, 37]:

$$1\,1\,1\,0\,0\,0\,1\,0\,0\,1\,0 \tag{3.4}$$

Посимвольне порівняння цієї послідовності з її зсунутою копією зведемо в табл. 3.1.

Графічне зображення АКФ даної послідовності Баркера показане на рис. 4.6:

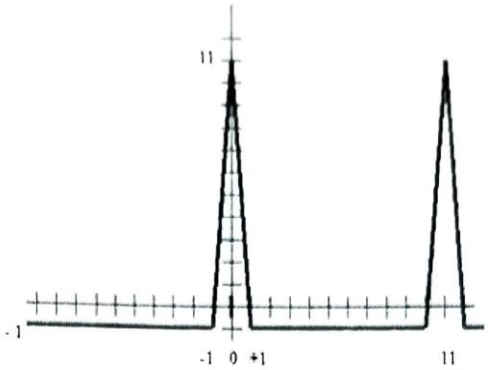

Рисунок 3.1 – АКФ послідовності Баркера

Таблиця 3.1 – Посимвольне порівняння послідовності Баркера з її зсунутою копією

Значення зрушення	Послідовність Баркера	Число збігів А	Число неспівпадінь Б	Значення різниці
1	01110001001	5	6	-1
2	10111000100	5	6	-1
3	01011100010	5	6	-1
4	00101110001	5	6	-1
5	10010111000	5	6	-1
6	01001011100	5	6	-1
7	00100101110	5	6	-1
8	00010010111	5	6	-1
9	10001001011	5	6	-1
10	11000100101	5	6	-1
0	11100010010	11	0	11

Такий АСF можна назвати ідеальним, тому що немає бічних списів, які могли б вняти некоректному розпізнаванню коду.

Як негативний приклад можна розглянути будь-яку кодову послідовність, приклад:

$$1\,1\,1\,0\,0\,0\,1\,1\,1\,0\,0 \qquad (3.5)$$

Після виконання обчислень, що відповідають попередньому прикладу, римуємо наступне графічне зображення автокореляційної функції, зображеної на ис. 3.2:

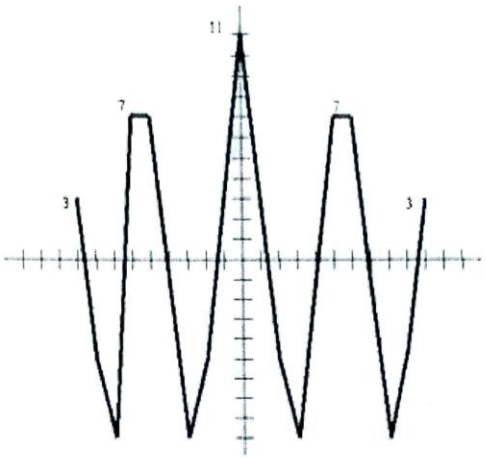

Рисунок 3.2 – АКФ довільної послідовності

Бічні списи 7 і 3 одиниць можуть призвести до збою в роботі системи при використанні такої послідовності для розповсюдження коду.

Для високошвидкісних широкосмугових систем, призначених для передачі інформації, але не для кодового поділу абонентів, зазвичай використовуються коди Баркера з хорошими властивостями автокореляції. Завдяки комп'ютерному моделюванню, т. зв Коди Вілларда, які, такої ж довжини, як і коди Баркера, іноді мають кращі кореляційні властивості. Послідовності коду Баркера довжиною понад 13 символів невідомі, тому для отримання більшого ПЗ, більшої стійкості до перешкод, а також для розділення кодів абонента використовуються послідовності більшої довжини, значну частину яких складають *M*-послідовності.

Це кодові послідовності, які є послідовностями максимальної довжини або m-послідовностями. Регістри зсуву або елементи затримки певної довжини зазвичай

використовуються для побудови m-послідовностей. Довжина m-послідовності дорівнює $2N$-1, де N — кількість бітів регістра зсуву. Різні варіанти підключення розрядних виходів до ланцюга зворотного зв'язку забезпечують певний набір послідовностей.

АКФ m послідовності дорівнює -1 для всіх значень затримки, за винятком області 0±1, де його значення змінюється від -1 до значення $2N$-1. Крім того, m послідовностей має ще одну цікаву властивість: у кожній послідовності на один більше нуля. Значна частина літератури присвячена методам створення та характеристики послідовностей, тому ми не будемо зупинятися на цьому детально.

Щоб дослідити можливості нового чіпсета PRISMTM, компанія Harris Semiconductor провела практичне дослідження коротких *M*-послідовностей і кодів Баркера, щоб знайти оптимальні для функцій автокореляції.

У роботі аналізується послідовність довжиною 15, яка має вигляд:

111 1000 1001 1010 (3.6)

Як виявилося, вона має гірші автокореляційні властивості, ніж послідовність Баркера, що складається з 13 символів наступного типу:

1 1111 0011 0101 (3.7)

Практичний вигляд АКФ *M*-послідовності показаний на малюнку:

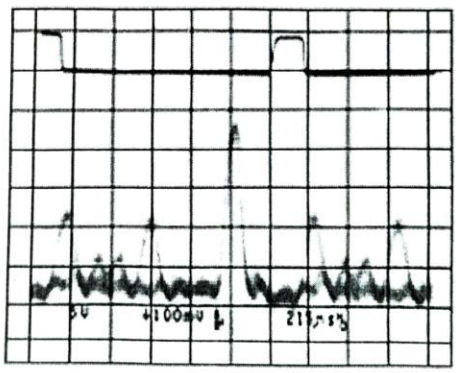

Рисунок 3.3 – АКФ м – послідовності довжиною 15

Для порівняння, АКФ кодової послідовності Баркера завдовжки 13:

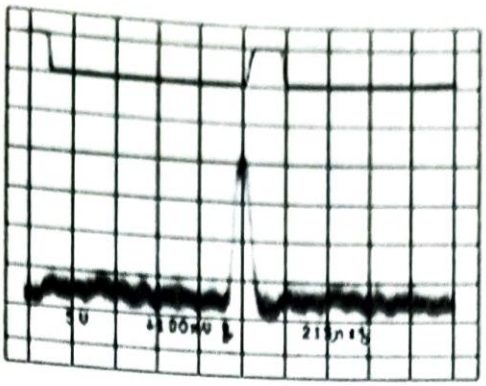

Рисунок 3.4 – АКФ кодової послідовності Баркера довжиною 13

На фото вище показаний імпульс синхронізації осцилографа. Як ви можете бачити на фотографіях, послідовність *M* має кілька великих бічних піків, які можуть значно погіршити якість прийому системи ShPS і іноді призвести до виявлення помилкового коду.

Як з'ясувалося в ході подальших досліджень, якщо до 13-символьної кодової послідовності Баркера додати два нулі, то АКФ отриманої послідовності

$$001\ 1111\ 0011\ 0101 \tag{3.8}$$

буде набагато кращим за описану послідовність m АКФ, яка також складається з 15 символів. АКФ новоотриманої послідовності:

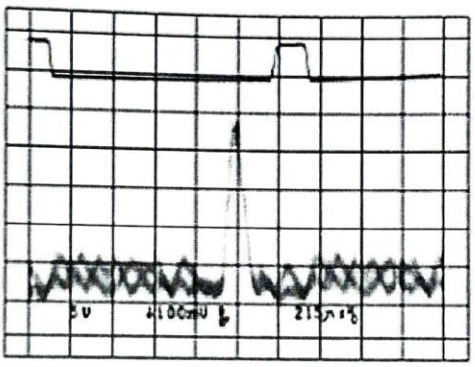

Рисунок 3.5 – АКФ квазібаркерової кодової послідовності довжиною 15

Таким чином, короткі m послідовності значно поступаються послідовностям Баркера щодо властивостей автокореляції, незважаючи на кращий баланс нулів і одиниць.

Найвідоміші системи, що використовують m-послідовності, включають систему мобільного зв'язку CDMA і глобальну систему навігації GPS. Система CDMA використовує три кодові послідовності. Перший, який використовується для синхронізації роботи всіх пристроїв, має змінну довжину $N(32131)$ 103 символів. Друга послідовність m має максимальну довжину $N=242-1$ і використовується для ідентифікації абонентських станцій від базової станції. Третя послідовність використовується для передачі корисної інформації між базовою станцією і абонентською станцією і є однією з послідовностей Уолша. Послідовності Уолша (у вигляді рядків або стовпців матриці Адамара) мають властивість бути ортогональними одна одній. Математично ортогональність означає, що за відсутності зсуву в часі між послідовностями Уолша їх скалярний добуток дорівнює нулю. З точки зору радіотехніки це дозволяє усунути взаємні перешкоди при передачі інформації від базової станції до кількох абонентських станцій і тим самим різко збільшити пропускну здатність системи зв'язку. Ця перевага ортогональності існує лише тоді, коли передача послідовності точно синхронізована для всіх абонентів. Точна синхронізація базових і абонентських станцій CDMA здійснюється в основному за допомогою глобальної навігаційної системи GPS. На додаток до послідовностей Уолша в системах зв'язку використовуються інші ортогональні послідовності: послідовності Діджілок і Стіффлера.

Окрім *M*-послідовностей, системи зв'язку використовують складні кодові послідовності, які є комбінаціями m-послідовностей і мають певні специфічні властивості. Найбільш відомі і використовувані з них - золоті послідовності. Послідовності золотого коду генеруються за допомогою простого генератора послідовностей на основі двох зсувних регістрів однакової ширини та мають дві переваги перед m послідовностями. По-перше, генератор кодової послідовності, побудований на основі двох регістрів зсуву довжиною N кожен, може генерувати, крім двох початкових m-рядків, N наступних послідовностей довжиною $2N$-1, тобто кількість генерованих кодових послідовностей значно збільшується. По-друге, коди Гулда можна вибирати так, щоб PCF для всіх кодових послідовностей, отриманих від одного генератора, був однаковим, а розмір його бічних піків був обмеженим. Для m

послідовностей неможливо гарантувати, що бічні списи АКФ не перевищуватимуть певне значення. Кодові послідовності Гулда використовуються в глобальних навігаційних системах, таких як GPS. Так званий "грубий" код (C/A - clear/acquisition) використовує послідовність Голда завдовжки 1023 символи, що передається з тактовою частотою 1,023 Мгц. Точно такий же код (P - precision), до якого мають доступ військові і спецслужби, використовує наддовгу складну послідовність з періодом повторення 267 днів і тактовою частотою 10,23 МГц. На додаток до складених послідовностей Гулда, найчастіше використовуються послідовності Касамі.

Послідовності М, послідовності Голда, послідовності Касама відносяться до послідовностей, які мають лінійний алгоритм формування. Головним недоліком таких послідовностей є їх передбачуваність і, як наслідок, відсутність секретності передачі. Нелінійні послідовності більш непередбачувані [50].

Останнім часом з'явилося багато публікацій про генерацію шумоподібних сигналів через явище динамічного хаосу. Феномен динамічного хаосу полягає в тому, що рух детермінованої динамічної системи за певних умов має всі властивості широкосмугового хаотичного процесу. При цьому основною особливістю алгоритмів, що описують це явище, є їх нелінійність, а особливістю генерованого часового процесу – його аперіодичність. Це відкриває можливість пошуку нового класу випадкових послідовностей для використання в IT-системах різного призначення, які б краще відповідали вимогам до псевдовипадкових послідовностей [2, 3].

Мобільні системи третього покоління, вже розроблені в рамках міжнародних європейських програм, використовуватимуть широкосмугові сигнали, що генеруються псевдовипадковими послідовностями. Зокрема, WCDMA, тобто широкосмуговий CDMA, розроблений Ericsson, був обраний базовим стандартом для UMTS - універсальної системи мобільного зв'язку. Існує більше двадцяти проектів, які більшою чи меншою мірою об'єднують усі розвинені телекомунікаційні компанії та провідні університети світу, намагаючись підійти до проблеми глобальних комунікацій майбутнього світу з різних точок зору.

Звичайно, у далекому майбутньому у кожного жителя нашої планети буде свій термінал, який має невеликі розміри і забезпечує свого власника всіма доступними видами зв'язку - від відеотелефону до доступу до глобальної світової інформаційної системи.

І є велика ймовірність того, що такі системи будуть використовувати кодове розділення абонентів за допомогою псевдовипадкових послідовностей. Ідея ортогонального кодування заснована на особливостях обробки сигналу на приймальній стороні системи передачі. Ця особливість полягає в тому, що ми можемо вибирати побічні сигнали передачі на свій розсуд, і сума переданого коду та шуму обробляється однаково. Якщо правильно вибрати передані коди, ефект буде полягати в посиленні переданих кодів і ослабленні ефекту шуму. Ця ж властивість зберігається, коли шум не є просто додатковим.

Для реалізації ортогонального кодування необхідно синтезувати квадратні матриці спеціального вигляду так, щоб їх добуток давав матрицю, близьку до одиничної, але в якій головна діагональ містить мономи, що визначають корекційні властивості ортогональних кодів. Пошук таких матричних пар здійснюється комбінаторними методами, що дозволяють знаходити приклади ортогональних кодів, які використовуються для оцінки підвищення завадостійкості.

Тому постало завдання знайти регулярний алгоритм побудови таких пар матриць для синтезу ортогональних кодів. Ця проблема може бути повністю вирішена в цій роботі, припускаючи, що використовувані кодові послідовності будуть квазіоптимальними. Завдяки цьому обмеженню ми отримуємо широкий клас таких квазіоптимальних кодових послідовностей, достатніх для вирішення практичних задач підвищення завадостійкості сучасних систем обробки інформації.

Розвиток квазіортогонального кодування на основі поліномів, незвідних у полі Галуа, призводить до значних математичних труднощів і збільшення складності реалізації. Тому в даній роботі вони не розглядаються.

3.2 Побудова квазіоптимальних послідовностей

Запропоновано метод побудови квазіортогональних послідовностей, заснований на перетворенні в'язальних структур, таких як ідеальні кільцеві переплетення (ІКВ).

Багатократна ідеальна кільцева в'язанка утворена послідовністю N цілих чисел $K_N = (k_1, k_2, ..., k_i, ..., k_N)$, на якій кільцеві суми набувають значення чисел натурального ряду, а кожне з чисел $1,2,...,S_N^R = S_N - 1$ є значенням R різних кільцевих сум. Між кількістю чисел N, кратністю R та сумою S_N всіх чисел R - кратного ідеального кільця існує співвідношення [23, 26, 27]:

$$S_N = \frac{N(N-1)}{R} + 1. \tag{3.9}$$

Щоб побудувати квазіортогональну послідовність з використанням N порядку R кратності ІКВ, ми вибираємо рядок S із клітинок одновимірного масиву, пронумерованих у порядку зростання, і заповнюємо клітинки, номери яких збігаються з числами, визначеними з ІКВ, інформацією про «одиниці». У решту незаповнених клітинок введемо «нулі».

Результуюча послідовність одиниць і нулів є S_N-бітовою квазіоптимальною кодовою послідовністю, шляхом циклічного зсуву якої можна отримати решта дозволених комбінацій.

Прикладом такого коду є таблиця кодових комбінацій, підготовлена за наказом ІКВ $N = 9$ кратності $R = 4$ (табл. 3.2):

$$(1, 1, 1, 2, 2, 5, 1, 3, 3). \tag{3.10}$$

Таблиця 3.2 – Квазіоптимальна кодова послідовність на основі ІКВ з $N=9$ та $R=4$

1	1	1	1	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0
0	1	1	1	1	0	1	0	1	0	0	0	0	1	1	0	0	1	0
0	0	1	1	1	1	0	1	0	1	0	0	0	0	1	1	0	0	1
1	0	0	1	1	1	1	0	1	0	1	0	0	0	0	1	1	0	0
0	1	0	0	1	1	1	1	0	1	0	1	0	0	0	0	1	1	0
0	0	1	0	0	1	1	1	1	0	1	0	1	0	0	0	0	1	1
1	0	0	1	0	0	1	1	1	1	0	1	0	1	0	0	0	0	1
1	1	0	0	1	0	0	1	1	1	1	0	1	0	1	0	0	0	0
0	1	1	0	0	1	0	0	1	1	1	1	0	1	0	1	0	0	0
0	0	1	1	0	0	1	0	0	1	1	1	1	0	1	0	1	0	0
0	0	0	1	1	0	0	1	0	0	1	1	1	1	0	1	0	1	0
0	0	0	0	1	1	0	0	1	0	0	1	1	1	1	0	1	0	1
1	0	0	0	0	1	1	0	0	1	0	0	1	1	1	1	0	1	0
0	1	0	0	0	0	1	1	0	0	1	0	0	1	1	1	1	0	1
1	0	1	0	0	0	0	1	1	0	0	1	0	0	1	1	1	1	0
0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	1	1	1	1
1	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	1	1	1
1	1	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	1	1
1	1	1	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	1

Будь-яка з $S_N(S_N\text{-}1)/2$ різних пар кодових комбінацій містить точно R N одиночних символів в однакових цифрах, що є результатом властивості ІКВ. Решта $N\text{-}R$ символів однієї і тієї ж кількості інших квазіоптимальних кодових послідовностей відрізняються від символів, що містяться в однойменних бітах. Тому мінімальна кодова відстань для даної квазіоптимальної кодової послідовності визначається як:

$$d_{\min} = 2(N\text{-}R) \tag{3.11}$$

Кількість помилок, які можна виявити t_1, і кількість помилок, які можна виправити t_2 за допомогою квазіоптимальної кодової послідовності коригування, визначаються мінімальною кодовою відстанню.:

$$t_1 \le d_{\min}\text{-}1,\ t_2 \le (t_1\text{-}1)/2 \tag{3.12}$$

Формули для визначення кількості помилок, які можуть бути виправлені t_2 або виявлені t_1 квазіоптимальною кодовою послідовністю:

$$t_1 \leq 2(N\text{-R})\text{-}1,\ t_2 \leq N\text{-}R\text{-}1 \tag{3.13}$$

Кодова відстань знаходиться як

$$d_{1,2} = S_N\text{-}2(N\text{-}R) \tag{3.14}$$

Формули, що визначають кількість помилок, які необхідно виявити або виправити за допомогою описаної квазіоптимальної кодової послідовності [23]:

$$\left.\begin{aligned} t_1 &\leq 2(N-R)-1 \\ t_2 &\leq N-R-1 \end{aligned}\right\},\ \text{якщо } S_N \geq 4(N\text{-}R) \tag{3.15};$$

$$\left.\begin{aligned} t_1 &\leq S_N - 2(N-R)-1 \\ t_2 &\leq \frac{S_N - 2(N-R+1)}{2} \end{aligned}\right\},\ \text{якщо } S_N < 4(N\text{-}R) \tag{3.16};$$

У розглянутих випадках значення параметрів N і R не пов'язані один з одним і можуть вибиратися довільно. У цьому випадку постає питання встановлення оптимального співвідношення між N і R за умови, що розглянута квазіоптимальна кодова послідовність отримує додаткові переваги. Завадостійкість квазіоптимальної кодової послідовності зростає зі збільшенням різниці $P=N\text{-}R$.

Максимальне значення P досягається за умови:

$$S_N = 2N. \tag{3.17}$$

Зв'язок між параметрами N і R, коли квазіоптимальна кодова послідовність набуває здатності ідентифікувати та виправляти максимально можливу кількість помилок [23, 26]:

$$\mathrm{P} = \begin{cases} N/2,\ N - \textit{парне} \\ (N-1)/2,\ N - \textit{непарне} \end{cases} \tag{3.18}$$

Квазіоптимальні кодові послідовності, створені за допомогою ІКВ, можуть виявляти до N-1 або виправляти до N/2-1 помилок для парних значень і виявляти до N або виправляти до (N-1)/2 помилок для непарних значень N.

3.3 Опис програмної реалізації та інструкція для користувача

Будь-який алгоритм можна реалізувати за допомогою відповідної програми. Переваги цієї реалізації очевидні: програмне забезпечення для кодування легко копіювати, легко використовувати та легко модифікувати відповідно до ваших конкретних потреб.

Усі популярні операційні системи мають вбудовані можливості шифрування файлів. Зазвичай вони використовуються для шифрування окремих файлів і повністю покладаються на керування ключами користувача. Тому застосування цих препаратів вимагає особливої уваги. По-перше, ключі ні в якому разі не можна зберігати на диску разом із закодованими ними файлами, а по-друге, незашифровані копії файлів слід видаляти відразу після шифрування.

Персональні комп'ютери практично не мають обмежень за призначенням, а їх можливості визначаються програмами, які обробляють будь-яку інформацію. Щоб отримати корисний результат за допомогою комп'ютера, необхідно мати відповідний додаток.

Сьогодні існує багато готових систем і пакетів. Однак будь-який пакет або програма з моменту своєї появи застаріває, тобто нові пропозиції як би покращують їх можливості, а самі завдання трансформуються в нові, не передбачені заздалегідь. Виходом із цієї ситуації є самостійна розробка програм, що вирішують конкретні завдання. У цьому випадку важливі навички та досвід програміста, його здатність створювати ефективні та надійні програми. З появою операційної системи Windows 7 принципово змінилися основні принципи створення додатків, які тепер можуть мати відмінний і сучасний графічний інтерфейс, можливість підключення і використання стандартних функцій Windows, підтримку роботи в локальних мережах і обмін даними з інші програми під час виконання. Безпосереднє програмування у Windows можливе на базі програмних пакетів, які це підтримують, наприклад, Borland Pascal for Windows, Delphi або Borland C++.

Програма написана в інтегрованому середовищі Delphi 7.0. Ніяких додаткових компонентів не використовувалося, тільки стандартні. Програма працює в таких операційних системах: Windows 7, Windows 8, Windows 10, Windows 11.

Після запуску програми її необхідно встановити перед кодуванням із виправленням помилок і декодуванням за допомогою квазіоптимальних послідовностей (рис. 3.6):

- ✓ вхідні дані (елементи квазіоптимальної послідовності) (рис. 3.6);
- ✓ кількість помилок, які відповідно до формул 3.14 та 3.15, знаходяться та виправляються (рис. 3.6);
- ✓ шлях до файлу, який необхідно закодувати та декодувати (рис. 3.6).

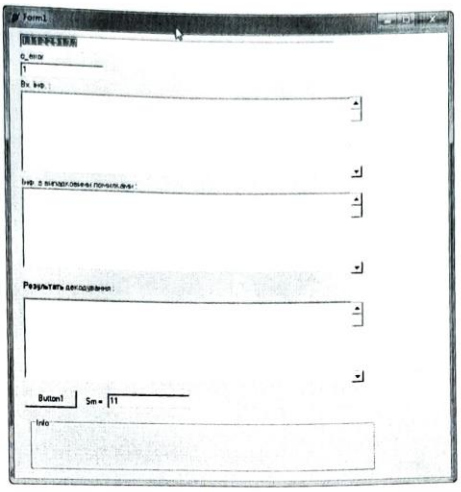

Рисунок 3.6 – Задання елементів квазіоптимальної послідовності

Після цього можна запускати процес синтезу. Для цього потрібно натиснути кнопку <Button1>.

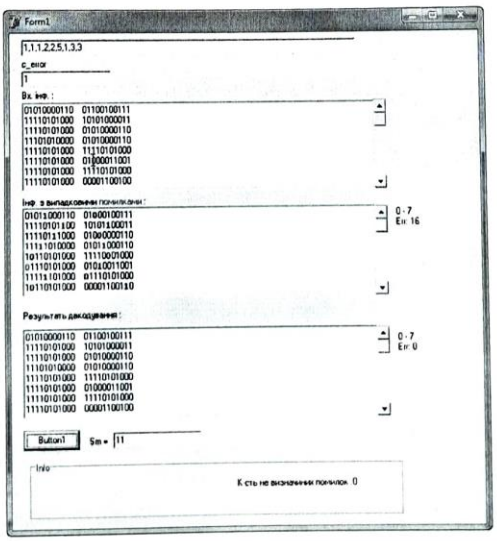

Рисунок 3.7 – Результати кодування та декодування з виправленням помилок за допомогою квазіоптимальних послідовностей.

Результати кодування та декодування з виправленням помилок з використанням квазіоптимальних послідовностей показані на рис. 3.7.

ВИСНОВКИ

1. Аналіз включення різних типів квазіоптимальних кодових послідовностей підкреслює переваги моделей, заснованих на зв'язувальних структурах, таких як ідеальні кільцеві зв'язки, що забезпечує більшу завадостійкість під час обробки інформації.
2. Показано можливість спрощеної побудови перешкодостійких квазіоптимальних моделей кодової послідовності, створення ефективних алгоритмів кодування та декодування інформації та ряду інших завдань з використанням ідеальних кільцевих в'язанок.
3. Використання квазіоптимальних кодових послідовностей на основі ідеальних кільцевих в'язанок дозволяє розширити сферу дослідження цих кодових послідовностей у напрямку покращення таких показників, як криптографічна міцність і завадостійкість.

СПИСОК ЛІТЕРАТУРИ

1. В.Ємець, А.Мельник, Р.Попович Сучасна криптографія. Основні поняття. – Львів: Бак, 2021. – 144с.
2. Різник В. В., Різник О. Я., Парубчак В. О. Застосування ідеальних кільцевих в'язанок для шифрування файлів. VIII международная научно-практическая конференция студентов аспирантов и молодых ученых САІТ-2016. Київ, 2016, с.82-85.
3. Різник В.В. Елементи теорії впорядкованих комбінаторних наборів: Навч. посібник. К.,1992.
4. Різник В.В. Комбінаторні моделі на одновимірних і багатовимірних в'язанках // Технічні засоби автоматизації вимірів та керування науковими дослідженнями. Вісник. Львів. політехн. інст. – 1992.
5. Різник В.В. Комбінаторні моделі та методи оптимізації в задачах інформатики // Навч. посібник. – Київ, 1991.
6. Різник В.В. Синтез оптимальних комбінаторних систем. Львів. Вища школа., 1989.
7. Різник В.В., Різник О.Я., Межерицький М.А. Про перелік комбінаторних моделей типу ідеальних кільцевих відношень // Контрольно-вимірювальна техніка. – Львів: Світ. – 2012 – Вип. 49.
8. Різник О.Я., Парубчак В.О. Кодування інформації на основі Використання монолітних кодів. Праці 13-ої міжнародної конференції з автоматичного управління, м. Вінниця, 2019, т.1, с.30-32.
9. Різник О.Я., Парубчак В.О., Скрибайло-Леськів Д.Ю. Використання числових в'язанок для кодування інформації. Праці міжнародної конференції "Сучасні комп'ютерні системи та мережі: розробка та використання", с.112-114.
10. A.Dollas, W.T.Rankin and D.McCracken published "A New Algorithm for Golomb Ruler Derivation and proof of the 19 Mark Ruler" in IEEE Transactions On Information Theory (January, 2018, Volume 44, Number 01).
11. Bartelt H., Lohmann A.W., Wirnitzer B. Phase and amplitude recovery from bispectra, Applied Optics, Vol. 23, 2022, P. 3121–3129.
12. Програмування. [Електронний ресурс] – Режим доступу: http://www.bib.com.ua (дата звернення: 10.04.2024).

13. Project OGR. [Електронний ресурс] – Режим доступу: http://www.distributed.net/ogr (дата звернення: 30.03.2024).
14. Gauley M. Direct product difference sets // J. of Combinatorial Theory. 2017, ser. A. Vol. 23. № 3. P. 321-332.
15. Liao X., Bao Z. Signal reconstruction from accumulation of bispectral radial slices, Optical Engineering, Vol. 39, No. 7, 2020, P. 2065–2074.
16. MathWorld's description of Golomb Rulers http://mathworld.wolfram.com/GolombRuler.html.
17. Nakamura M. Waveform estimation from noisy signals with variable signal delay using bispectrum averaging, IEEE Transactions on Biomedical Engineering, Vol. 40, No. 2, 2023, P. 118–127.
18. Nikias C.L., Raghuveer M.R. Bispectral estimation: A digital signal processing framework, Proc. IEEE, Vol. 75, No. 7, 2021, P. 869–891.
19. Технології програмування. [Електронний ресурс] – Режим доступу: http://www.relib.com (дата звернення: 10.04.2024).
20. Solomon W. Golomb. University Professor. http://commsci.usc.edu/faculty/golomb.html.
21. Totsky A.V., Kurbatov I.V., Lukin V.V., Egiazarian K.O., Astola J.T. Combined bispectrum-filtering techniques for radar output signal reconstruction in ATR applications, Proceedings of International Conference "Automatic Target Recognition XIII"; Ed. Firooz A. Sadjadi; Orlando (USA), April 2023, SPIE Vol. 5094, P. 301–312.
22. Zhang X., Shi Y., Bao Z. A new feature vector using selected bispectra for signal classification with application in radar target recognition, IEEE Transactions on Signal Processing, Vol. 49, No. 9, 2021, P. 1875–1885.
23. Einführung in die Constraint-Programmierung. [Електронний ресурс] – Режим доступу: http://commsci.usc.edu/faculty/golomb.html (дата звернення: 20.05.2024).
24. [Електронний ресурс] – Режим доступу: http://www.rsdn.ua (дата звернення: 04.05.2024).
25. [Електронний ресурс] – Режим доступу: http://www.soliday.com/stephen/papers/#ICGA23 (дата звернення: 16.04.2024).
26. [Електронний ресурс] – Режим доступу: http://www.unl.edu/tcweb/Faculty/fowler/golomb/genrulPaper.html (дата звернення: 10.03.2024).

ДОДАТКИ

ДОДАТОК А. ЛІСТИНГ ПРОГРАМИ ДОСЛІДЖЕННЯ ЗАВАДОСТІЙКИХ ВЛАСТИВОСТЕЙ КВАЗІОРТОГОНАЛЬНИХ КОДІВ

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ComCtrls, ExtCtrls;

type
  int = integer;
  str = string;
  v_mas=array of int;
  Tmas_v= array of str;

  TForm1 = class(TForm)
    Label3: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Edit1: TEdit;
    LabeledEdit1: TLabeledEdit;
    ScrollBar1: TScrollBar;
    ScrollBar2: TScrollBar;
    ScrollBar3: TScrollBar;
    RichEdit1: TRichEdit;
    RichEdit2: TRichEdit;
    RichEdit3: TRichEdit;
    Button1: TButton;
    od: TOpenDialog;
    Label2: TLabel;
    Label1: TLabel;
    GroupBox1: TGroupBox;
    Label5: TLabel;
    Edit2: TEdit;
    Label8: TLabel;
    procedure FormCreate(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure ScrollBar1Scroll(Sender: TObject; ScrollCode: TScrollCode;
      var ScrollPos: Integer);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

  function LinesVisible(Memo: TrichEdit): integer;

var
  Form1: TForm1;
  matrix:Tmas_v;
  error:int;
  sn,count:int;
  _in,_er,_out:TStringList;
  c_lines:int;
  scroll_n:int;
```

```
implementation

{$R *.dfm}

function LinesVisible(Memo: TRichEdit): integer;
var
 OldFont : HFont;
 Hand : THandle;
 TM : TTextMetric;
 Rect : TRect;
 tempint : integer;
begin
 Hand := GetDC(Memo.Handle);
 try
  OldFont := SelectObject(Hand, Memo.Font.Handle);
  try
   GetTextMetrics(Hand, TM);
   Memo.Perform(EM_GETRECT, 0, longint(@Rect));
   tempint := (Rect.Bottom - Rect.Top) div
   (TM.tmHeight + TM.tmExternalLeading);
  finally
   SelectObject(Hand, OldFont);
  end;
 finally
  ReleaseDC(Memo.Handle, Hand);
 end;
 Result := tempint;
end;

function GetMax(m:v_mas):int;
var i:int;
begin
Result:=0;
 for i:=0 to high(m) do
  if m[i]>Result then Result:=m[i];
end;


function StrToIKB(s:str; var m:v_mas):bool;
var p,pbeg:pchar;
  len:int;
begin
s:=s+',#0;
len:=0;
p:=@s[2];
pbeg:=p-1;
while p^<>#0 do
 begin
  if p^=',' then
   begin
    p^:=#0;
    inc(len);
    SetLength(m,len);
    m[len-1]:=strtoint(pbeg);
    inc(p);
    pbeg:=p;
   end else inc(p);
 end;
Result:=true;
end;

function rolmas(m:v_mas; i:int; len:int):v_mas;
```

```
var slen,j:int;
  i:int;
begin
slen:=mmax;//;len shr 1 + 1;
i:=0;
SetLength(Result,slen);
while bool(slen)do
 begin
  Result[j]:=m[i];
  inc(i);
  inc(j);
  if i>= len then i:=0;
  dec(slen);
 end;
end;

function IKBToBarker(m:v_mas):v_mas;
var len:int;
  i,count,j:int;
  p1:pint;
begin
len:=0;
for i:=0 to high(m) do
 len:=len+m[i];
SetLength(Result,len);
  p1:=@Result[0];
  count:=high(m)+1;
  p1^:=1;
  inc(p1);
  for j:=0 to count-2 do
    begin
      FillMemory(p1,(m[j]-1)*4,0);
      Inc(p1,m[j]-1);
      p1^:=1;
      inc(p1);
    end;
  FillMemory(p1,(m[j]-1)*4,0);
end;

function Bild_Matrix(m:v_mas):Tmas_v;
var len,len2:int;
  i,j:int;
  v:v_mas;
begin
 len:=high(m)+1;
 count:=len;
 len2:=mmax;//;len shr 1 + 1;
 SetLength(Result,len);
 for i:= 0 to len-1 do
  begin
   v:=rolmas(m,i,len);
   for j:=1 to len2 do
    Result[i]:=Result[i]+inttostr(v[j-1]);
  end;
end;

function rorstr(s:str; i:int):str;  ////
var
 c:char;
 len,j,slen:int;
begin
 len:=length(s);
```

```
  slen:=len;
  j:=1;
  SetLength(Result,len);
  while bool(slen) do
   begin
    Result[j]:=s[i];
    inc(i);
    inc(j);
    if i>=len then i:=1;
    dec(slen);
   end;
end;


function cyclic_fild(s1,s2:str):int;
var i,j,s,max:int;
 t_s:str;
begin
max:=0;
for i:=1 to length(s1) do
 begin
  t_s:=rorstr(s1,i);
  FillMemory(@t_s[1],i-1,48);
  s:=0;
  for j:=1 to length(s2)do
    if t_s[j] = s2[j] then inc(s);
  if s > max then
    max:=s;
 end;
 Result:=max;
end;

function find_d (s:str):str;
var v:str;
   i,j,sum,sum1,max,max_id:int;
begin
v:=copy(s,1,sn);
max:=0;
//max_id:=-1;
for i:=0 to count-1 do
 begin
  sum:=0;
  sum1:=0;
  for j:=1 to sn do
   if matrix[i][j] = v[j] then
    inc(sum);
 // sum:=cyclic_fild(matrix[i],v);
  if (max < sum)  then
//  if (max < sum) and (length(v)-sum = sum1) then
  begin
   max:=sum;
   max_id:=i;
  end;
 end;
Result:=matrix[max_id]+'   ';
v:=copy(s,5+sn,sn);
max:=0;
for i:=0 to count-1 do
 begin
 sum:=0;
 sum1:=0;
 for j:=1 to sn do
```

```
    if matrix[i][j] = v[j] then
      inc(sum);
//    sum:=cyclic_fild(matrix[i],v);
    if (max < sum) then
//if (max < sum)and(sum<>sum1) then
    begin
     max:=sum;
     max_id:=i;
    end;
  end;
Result:=Result+matrix[max_id];
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
od.InitialDir:=GetCurrentDir+'\';
Randomize();
c_lines:=LinesVisible(RichEdit1);
ScrollBar1.SmallChange:=c_lines-1;
ScrollBar2.SmallChange:=c_lines-1;
ScrollBar3.SmallChange:=c_lines-1;
end;

procedure m1;
var
   k,p,a:str;
   i,j,E,r,q,f:int;
   o,l:pchar;
   qa,qs:TStringStream;
   dm:int;
   c_er:int;
   max,za:int;

begin
c_er:=0;
za:=0;
if scroll_n>_er.Count-c_lines then
max:=_er.Count-c_lines else
max:=scroll_n+c_lines;
Form1.RichEdit2.Lines.Clear;
 for i:=max-c_lines to max-1 do
  begin
   k:=_er.Strings[i];
   p:=_in.Strings[i];
   form1.RichEdit2.Lines.Add(k);
   q:=2*sn+4;
   if (k<>'')and (p<>'') then
   for j:=1 to q do
   if (k[j]<>' ')then
    if k[j]<>p[j] then
     with form1.RichEdit2 do
     begin
      inc(c_er);
      if c_er mod 20 = 0 then
      form1.Label1.Caption:=inttostr(c_er);
      SelStart:=Perform(EM_LINEINDEX,za, 0) + j-1;
      SelLength:=1;
      SelAttributes.Color:=clRed;
      SelAttributes.Style:=[fsBold];
      SelText:=k[j];
     end;
   inc(za);
```

```
  end;
form1.Label1.Caption:=inttostr(max-c_lines)+' - '+inttostr(max-1)+#13#10'Err: '+inttostr(c_er);
end;

procedure m0;
var i,max:int;
begin
if scroll_n>_in.Count-c_lines then
max:=_in.Count-c_lines else
max:=scroll_n+c_lines;
Form1.RichEdit1.Lines.Clear;
 for i:=max-c_lines to max-1 do
  form1.RichEdit1.Lines.Add(_in.Strings[i]);
end;

procedure m2;
var
   k,p,a:str;
   i,j,E,r,q,f:int;
   o,l:pchar;
   qa,qs:TStringStream;
   dm:int;
   c_er,max,za:int;
begin
c_er:=0;
za:=0;
if scroll_n>_out.Count-c_lines then
max:=_out.Count-c_lines else
max:=scroll_n+c_lines;
q:=2*sn+4;
Form1.RichEdit3.Lines.Clear;
 for i:=max-c_lines to max-1 do
  begin
   k:=_out.Strings[i];
   p:=_in.Strings[i];
   form1.RichEdit3.Lines.Add(k);
   if (k<>'')and (p<>'')and(k<>p)then
   for j:=1 to q do
   if (k[j]<>' ')then
    if k[j]<>p[j] then
     with form1.RichEdit3 do
     begin
      inc(c_er);
      if c_er mod 20 = 0 then
      form1.Label2.Caption:=inttostr(c_er);
      SelStart:=Perform(EM_LINEINDEX,za, 0) + j-1;
      SelLength:=1;
      SelAttributes.Color:=clRed;
      SelAttributes.Style:=[fsBold];
      SelText:=k[j];
     end;
     inc(za);
  end;
form1.Label2.Caption:=inttostr(max-c_lines)+' - '+inttostr(max-1)+#13#10'Err: '+inttostr(c_er);
end;

procedure TForm1.Button1Click(Sender: TObject);
var F:TFileStream;
  m:v_mas;
  s:array of byte;
  k,p,a:str;
  i,j,r,q:int;
```

```
      inc(id);
     end;
   end;
Label5.Caption:='К-сть не визначиних помилок :'+inttostr(id);
end;


procedure TForm1.ScrollBar1Scroll(Sender: TObject; ScrollCode: TScrollCode;
  var ScrollPos: Integer);
begin
scroll_n:=ScrollPos;
ScrollBar1.Position:=scroll_n;
ScrollBar2.Position:=scroll_n;
ScrollBar3.Position:=scroll_n;
m0;
m1;
m2;
end;

end.
```