

*В.В. Шаряк,
Н.Я. Возна,
Я.М. Николайчук*

**КОДУВАННЯ ДАНИХ
ТА ОРГАНІЗАЦІЯ
РОЗПОДІЛЕНИХ БАЗ ДАНИХ
У РОЗШИРЕНИХ ПОЛЯХ ГАЛУА**

Монографія

*Тернопіль
2024*

Шаряк В.В., Возна Н.Я., Николайчук Я.М. Кодування даних та організація розподілених баз даних у розширених полях Галуа. – Тернопіль: ЗУНУ, 2024. – 271 с.

Рецензенти:

В.К. Задірака, доктор фізико-математичних наук, професор, член-кореспондент Національної академії наук України, завідувач відділу Інституту кібернетики ім.В.М.Глушкова НАН України

М.П. Дивак, доктор технічних наук, професор, декан факультету комп'ютерних інформаційних технологій Тернопільського національного економічного університету

У монографії викладено фундаментальні основи кодування даних у теоретико-числовому базисі Галуа та їх застосування при організації баз даних. Здійснено аналіз використання теоретико-числових базисів в інформаційних технологіях, що показує широке застосування базису Радемахера, який породжує двійкові кодові системи числення. В той же час світовий досвід застосування теоретико-числових базисів для швидкодіючих перетворень показує, що значне прискорення швидкості процесорів та спрощення програмних алгоритмів забезпечують способи перетворення даних у базисах Уолша, Крестенсона та Галуа.

Оптимізація сучасних технологій побудови баз даних є одним із вирішальних факторів забезпечення ефективного зберігання корпоративної інформації та використання даних користувачем. Реляційна модель бази даних в даний час отримала широке застосування і ефективно використовується при створенні потужних баз даних. Перевагою реляційних баз даних по відношенню до ієрархічних є забезпечення можливості формалізації перетворення даних на основі алгоритмів реляційної алгебри.

Особливістю використання базису Галуа при організації БД є можливість не тільки компактного кодування даних, але і організований захист даних від помилок та несанкціонованого доступу.

Одною з переваг сучасних комп'ютерних систем є просторова та часова роздільність. На основі ТЧБ Галуа перспективним є створення інформаційної технології побудови баз даних глобальних розподілених комп'ютерних систем та конгломератів епохальних сховищ даних комп'ютеризованих систем.

Монографія призначена для науковців та спеціалістів у галузі "Інформаційні технології" та може бути корисна для аспірантів та студентів.

СТИСЛИЙ ЗМІСТ

СПИСОК СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ.....	9
ВЕЛИКИЙ ФРАНЦУЗЬКИЙ МАТЕМАТИК ЕВАРІСТ ГАЛУА І ЗАСТОСУВАННЯ ЙОГО ТЕОРІЇ У НАШ ЧАС.....	12
ПЕРЕДМОВА.....	18
ВСТУП.....	20
РОЗДІЛ 1. ПРИНЦИПИ СТРУКТУРНОЇ ОРГАНІЗАЦІЇ ТА ХАРАКТЕРИСТИКИ ДАНИХ І БАЗ ДАНИХ.....	28
РОЗДІЛ 2. ТЕОРІЯ ПОЛІВ ГАЛУА.....	46
РОЗДІЛ 3. ГРУПА ГАЛУА.....	52
РОЗДІЛ 4. ПОЛЯ ГАЛУА.....	58
РОЗДІЛ 5. КОДОВІ СИСТЕМИ ГАЛУА.....	66
РОЗДІЛ 6.ТЕОРЕТИКО-ЧИСЛОВІ БАЗИСИ (ТЧБ).....	80
РОЗДІЛ 7.ТЧБ ТА СИСТЕМИ ЧИСЛЕННЯ НА ОСНОВІ СУМ ПОЛІВ ГАЛУА.....	104
РОЗДІЛ 8. ЗМЕНШЕННЯ НАДЛИШКОВОСТІ ДАНИХ У ТЧБ ГАЛУА.....	137
РОЗДІЛ 9. ХАРАКТЕРИСТИКИ КОРЕКТУЮЧИХ КОДІВ.....	150
РОЗДІЛ 10. СИГНАЛЬНІ КОРЕКТУЮЧІ КОДИ В ТЧБ ГАЛУА.....	165
РОЗДІЛ 11. КРИТЕРІЇ ЕМЕРДЖЕНТНОСТІ ПРИ ОРГАНІЗАЦІЇ МОДЕЛЕЙ БАЗ ДАНИХ.....	194
РОЗДІЛ 12.СПОСОБИ ДОСТУПУ ДО БАЗ ДАНИХ ГЛОБАЛЬНИХ РОЗПОДІЛЕНИХ КОМП'ЮТЕРНИХ СИСТЕМ.....	197
РОЗДІЛ 13.МОДЕЛІ БАЗИ ДАНИХ В КОДАХ ПОЛЯ ГАЛУА.....	205
РОЗДІЛ 14.ОРГАНІЗАЦІЯ ЛІНІЙНО–РЕКУРЕНТНИХ МОДЕЛЕЙ БАЗ ДАНИХ НА ОСНОВІ ТЧБ ГАЛУА.....	213
РОЗДІЛ 15. ОРГАНІЗАЦІЯ ВЕРТИКАЛЬНО–РЕКУРЕНТНИХ МОДЕЛЕЙ БАЗ ДАНИХ НА ОСНОВІ ТЧБ ГАЛУА.....	217
РОЗДІЛ 16. ОРГАНІЗАЦІЯ РЕЛЯЦІЙНИХ МОДЕЛЕЙ БАЗ ДАНИХ НА ОСНОВІ ДВОМІРНИХ КОДІВ ПОЛЯ ГАЛУА.....	224
РОЗДІЛ 17. РЕЛЯЦІЙНО–РЕКУРЕНТНА МОДЕЛЬ БАЗИ ДАНИХ У ТЧБ ГАЛУА.....	230
РОЗДІЛ 18.ОБ'ЄКТНО–РЕЛЯЦІЙНО–РЕКУРЕНТНА МОДЕЛЬ СХОВИЩА ДАНИХ У ТЧБ ГАЛУА.....	232
РОЗДІЛ 19. КОНГЛОМЕРАТИ ЕПОХАЛЬНИХ СХОВИЩ ДАНИХ У ТЧБ ГАЛУА....	237
РОЗДІЛ 20. ФОРМУВАННЯ І ОРГАНІЗАЦІЯ БАЗ ДАНИХ РОЗПОДІЛЕНИХ КОМП'ЮТЕРНИХ СИСТЕМ.....	249
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	265

ЗМІСТ

СПИСОК СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ.....	9
ВЕЛИКИЙ ФРАНЦУЗЬКИЙ МАТЕМАТИК ЕВАРІСТ ГАЛУА І ЗАСТОСУВАННЯ ЙОГО ТЕОРІЇ У НАШ ЧАС.....	12
ПЕРЕДМОВА.....	18
ВСТУП.....	20
РОЗДІЛ 1. ПРИНЦИПИ СТРУКТУРНОЇ ОРГАНІЗАЦІЇ ТА ХАРАКТЕРИСТИКИ ДАНИХ І БАЗ ДАНИХ.....	28
1.1. Загальні принципи та концепція структурної організації даних.....	28
1.2. Системні характеристики даних та класифікація моделей даних.....	33
1.3. Типи баз даних та рівні їх проектування.....	36
1.3.1. Класифікація баз даних.....	37
1.3.2. Архітектура та характеристики моделей баз даних.....	38
РОЗДІЛ 2. ТЕОРІЯ ПОЛІВ ГАЛУА.....	46
2.1 Визначення поля.....	46
2.1.1 Типи полів.....	46
2.1.2 Властивості полів алгебри чисел.....	46
2.2. Група Галуа. Співвідношень між коренями поліномів.....	47
2.3 Основні властивості групи Галуа.....	49
2.4. Підстановки групи Галуа.....	50
2.4.1. Автоморфізм нормального поля.....	50
2.4.2. Нормальні поля. Теорема Лагранжа.....	51
РОЗДІЛ 3. ГРУПА ГАЛУА.....	52
3.1. Початкові визначення групи Галуа.....	52
3.1.1. Корені полінома. Розкладання полінома на лінійні множники.....	52
3.1.2. Найбільший спільний дільник поліномів. Алгоритм Евкліда.....	53
3.1.3. Подання коефіцієнтів полінома через його корені.....	53
3.1.4. Результант.....	54
3.1.5. Дискримінант.....	55
3.2. Звідні та незвідні поліноми.....	56
3.2.1. Критерій незвідності Ейзенштейна.....	57
РОЗДІЛ 4. ПОЛЯ ГАЛУА.....	58
4.1. Скінченні комутативні поля (поля Галуа).....	58
4.2. Сепарабельні і несепарабельні розширення.....	59
4.3. Досконалі і недосконалі поля.....	61
4.4. Простота алгебраїчних розширень. Теорема про примітивний елемент.....	62
4.5. Норми і сліди.....	63

РОЗДІЛ 5. КОДОВІ СИСТЕМИ ГАЛУА.....	66
5.1. Базис Уолша – теоретична основа кодів поля Галуа.....	66
5.2. Класи кодів поля Галуа (КПГ).....	70
5.2.1 Одновимірні дворівневі КПГ.....	70
5.2.2. Двовимірні дворівневі КПГ.....	73
5.2.3. Просторова форма двовимірних КПГ.....	75
5.2.4. Полярно-спіральна форма КПГ.....	77
5.2.5. Двомірний однобітовий КПГ.....	78
РОЗДІЛ 6.ТЕОРЕТИКО-ЧИСЛОВІ БАЗИСИ (ТЧБ).....	80
6.1. Характеристика кодових матриць теоретико – числових базисів.....	80
6.2.Системні характеристики СПД.....	80
6.3. Кодові матриці дискретних ТЧБ.....	95
6.3.1.ТЧБ Унітарний.....	95
6.3.2.ТЧБ Хаара.....	95
6.3.3. ТЧБ Крейга.....	96
6.3.4. ТЧБ Радемахера.....	97
6.3.5. ТЧБ Грея.....	98
6.3.6. ТЧБ Крестенсона.....	99
6.3.7. ТЧБ Уолша.....	100
6.3.8. ТЧБ Галуа.....	101
РОЗДІЛ 7. СИСТЕМИ ЧИСЛЕННЯ НА ОСНОВІ СУМ ПОЛІВ ГАЛУА.....	104
7.1. Теоретичні основи цілочисельної СЗК ТЧБ Крестенсона.....	104
7.2. Кодування інформаційних потоків в СЗК з довільним порядком реєстрації даних.....	107
7.3. Каскадне кодування даних на основі методу залишків та СЗК.....	108
7.4. Нормалізована форма СЗК.....	111
7.5. Досконалі форми СЗК.....	117
7.6. Міжбазисні перетворення на основі розмежованої СЗК.....	123
7.6.1. Розрахунок набору модулів для реалізації 16-бітного та 1024-бітного процесора.....	124
7.6.2. Інформаційна база розмежованої СЗК.....	125
7.6.3. Бінарно-розмежована СЗК.....	127
7.6.4. Реалізація архітектури та оцінка швидкодії суматора по модулю P_j	128
7.6.5.Оцінка складності арифметики у базисах Радемахера, Крестенсона та Галуа.....	129
7.7. Оптимізація міжбазисного перетворення Галуа-Крестенсона.....	131
7.8. Теорія арифметичних операцій у кодах поля Галуа.....	133

РОЗДІЛ 8. ЗМЕНШЕННЯ НАДЛИШКОВОСТІ ДАНИХ У ТЧБ ГАЛУА.....	137
8.1. Стиснення даних на основі логіко-статистичних інформаційних моделей та кодів Галуа.....	137
8.2. Теорія та застосування базисних функцій кодів поля Галуа.....	141
8.3. Стиснення алфавітно-цифрової інформації у базисі Галуа.....	145
РОЗДІЛ 9. ХАРАКТЕРИСТИКИ КОРЕКТУЮЧИХ КОДІВ.....	150
9.1. Класифікація класів коректуючих кодів.....	150
9.2. Теоретичні основи захисту даних від помилок.....	159
9.3. Порівняльний аналіз методів формування кодових ключів Галуа.....	160
9.4. Методи формування кодових ключів Галуа на основі ТЧБ Унітарного.....	162
9.5. Методи формування кодових ключів на основі М- послідовності.....	162
РОЗДІЛ 10. СИГНАЛЬНІ КОРЕКТУЮЧІ КОДИ В ТЧБ ГАЛУА.....	165
10.1. Методологія формування та оцінка ефективності формування сигнальних кодів базису Галуа.....	165
10.1.1. Імпульсні методи маніпуляції.....	165
10.1.2. Потенціальні методи маніпуляції.....	167
10.1.3. Квазітрійкові методи маніпуляції сигналів.....	168
10.1.4. Частотні, фазові та амплітудно-фазові сигнали, двовимірні та багатовимірні сигнальні простори.....	170
10.2. Сигнальні коди на основі теорії кодів Галуа.....	171
10.2.1. Спіральні властивості коректуючих кодів в ТЧБ Галуа.....	175
10.2.2. Метод формування позиційно-сигнального коду.....	176
10.2.3. Несиметричний рекурентний сигнальний код.....	183
10.2.4. Дослідження надлишковості захисту даних від помилок існуючих протоколів передавання даних.....	184
10.2.5. Симетричний рекурентний сигнальний код.....	185
10.3. Теоретичні основи оцінки ефективності сигнальних кодів.....	186
10.4. Теоретичні основи ентропійних методів опрацювання сигналів.....	187
10.4.1. Ентропійні характеристики двовимірних та одновимірних ШПС.....	189
10.4.2. Кореляційні характеристики ШПС.....	189
10.4.3. Оцінка ентропії одновимірних та двовимірних ШПС.....	191
РОЗДІЛ 11. КРИТЕРІЇ ЕМЕРДЖЕНТНОСТІ ПРИ ОРГАНІЗАЦІЇ МОДЕЛЕЙ БАЗ ДАНИХ.....	194
11.1. Оцінка надлишковості БД.....	194
11.2. Критерії та методи підвищення ефективності БД.....	194
11.3. Фактори, які впливають на підвищення ефективності БД.....	195
11.4. Результати досліджень емерджентності різних типів моделей БД.....	195

РОЗДІЛ 12. СПОСОБИ ДОСТУПУ ДО БАЗ ДАНИХ ГЛОБАЛЬНИХ РОЗПОДІЛЕНИХ КОМП'ЮТЕРНИХ СИСТЕМ.....	197
12.1. Керування правами доступу.....	197
12.2. Середовища доступу до БД глобальних розподілених комп'ютерних систем в базисі Галуа.....	197
12.3. Пошукові системи в інтернет.....	197
12.4. Об'єм та швидкість доступу до БД.....	202
12.5. Перспективні принципи зберігання та доступу до інформації в ГРКС в базисі Галуа.....	203
РОЗДІЛ 13. МОДЕЛІ БАЗИ ДАНИХ В КОДАХ ПОЛЯ ГАЛУА.....	205
13.1. Методи кодування елементів ієрархічних моделей БД у різних ТЧБ.....	205
13.2. Ієрархічно–рекурентна модель БД у ТЧБ Галуа.....	209
13.3. Ієрархічно–реляційна модель БД у ТЧБ Галуа.....	210
РОЗДІЛ 14. ОРГАНІЗАЦІЯ ЛІНІЙНО–РЕКУРЕНТНИХ МОДЕЛЕЙ БАЗ ДАНИХ НА ОСНОВІ ТЧБ ГАЛУА.....	213
14.1. Однорівнева лінійно-рекурентна модель БД у ТЧБ Галуа.....	213
14.2. Дворівнева лінійно-рекурентна модель БД у ТЧБ Галуа.....	214
14.3. Багаторівнева лінійно-рекурентна модель БД у ТЧБ Галуа.....	214
РОЗДІЛ 15. ОРГАНІЗАЦІЯ ВЕРТИКАЛЬНО–РЕКУРЕНТНИХ МОДЕЛЕЙ БАЗ ДАНИХ НА ОСНОВІ ТЧБ ГАЛУА.....	217
15.1. Однорівнева вертикально-рекурентна модель БД у ТЧБ Галуа.....	217
15.2. Дворівнева вертикально-рекурентна модель БД у ТЧБ Галуа.....	218
15.3. Багаторівнева вертикально-рекурентна модель БД у ТЧБ Галуа.....	219
РОЗДІЛ 16. ОРГАНІЗАЦІЯ РЕЛЯЦІЙНИХ МОДЕЛЕЙ БАЗ ДАНИХ НА ОСНОВІ ДВОМІРНИХ КОДІВ ПОЛЯ ГАЛУА.....	224
16.1. Двомірні коди поля Галуа.....	224
16.1.1. Матричні двомірні коди поля Галуа.....	224
16.1.2. Критерії оцінки ефективності кодування даних в РБД.....	225
16.2. Рекурентні двомірні коди поля Галуа.....	225
16.2.1. Теоретичні основи одновірних кодів поля Галуа.....	225
16.2.2. Теоретичні основи двовірних кодів поля Галуа.....	225
16.3. Дослідження і обробка даних результатів аналізу об'ємів матриці ідентифікаторів атрибутів і кортежів відношення реляційних БД.....	226
16.3.1. Алгоритм визначення коефіцієнта надлишковості реляційної БД в ТЧБ Радемахера при багатьох атрибутів і кортежів реляційної БД.....	227
16.3.2. Алгоритми визначення коефіцієнта надлишковості при багатьох масивах доменів реляційної БД.....	227

РОЗДІЛ 17. РЕЛЯЦІЙНО–РЕКУРЕНТНА МОДЕЛЬ БАЗИ ДАНИХ У ТЧБ ГАЛУА.....	230
17.1. Побудова відношення реляційних баз даних в базисі Галуа.....	230
17.2. Розробка рекурентних двомірних кодів поля Галуа.....	230
17.3. Аналіз ефективності кодування ідентифікаторів в реляційно–рекурентній БД у ТЧБ Галуа.....	231
РОЗДІЛ 18. ОБ’ЄКТНО–РЕЛЯЦІЙНО–РЕКУРЕНТНА МОДЕЛЬ СХОВИЩА ДАНИХ У ТЧБ ГАЛУА.....	232
18.1. Об’єктно-реляційно–рекурентна модель сховища даних глобальних розподілених комп’ютерних систем у базисі Галуа (СД ГРКС БГ).....	232
18.2. Логічна структура об’єктно-реляційно-рекурентної моделі сховищ даних глобальних розподілених комп’ютерних систем в ТЧБ Галуа.....	234
18.3. Фізична структура об’єктно-реляційно-рекурентної моделі сховищ даних глобальних розподілених комп’ютерних систем в ТЧБ Галуа.....	234
18.4. Аналіз ефективності кодування ідентифікаційних даних об’єктно–реляційно-рекурентної моделі сховищ даних глобальних розподілених комп’ютерних систем в ТЧБ Галуа.....	236
РОЗДІЛ 19. КОНГЛОМЕРАТИ ЕПОХАЛЬНИХ СХОВИЩ ДАНИХ У ТЧБ ГАЛУА.....	237
19.1. Перспективні напрямки розвитку конгломератів епохальних моделей СД ГРКС у ТЧБ Галуа.....	243
19.2. Аналіз характеристики кодування моделей конгломератів епохальних сховищ даних в базисі Галуа у структурі світового господарства.....	244
19.3. Принципи організації моделей конгломератів епохальних сховищ даних в базисі Галуа у структурі світового господарства.....	247
19.4. Синтез циклічних конгломератів епохальних сховищ даних в базисі Галуа..	248
19.5. Моделювання логічної структури циклічних конгломератів епохальних моделей сховищ даних в ТЧБ Галуа.....	248
РОЗДІЛ 20. ФОРМУВАННЯ І ОРГАНІЗАЦІЯ БАЗ ДАНИХ РОЗПОДІЛЕНИХ КОМП’ЮТЕРНИХ СИСТЕМ.....	251
20.1. Порівняння типових БД з запропонованою організацією БД у ТЧБ Галуа....	251
20.2. Кодування та доступ до сховищ алфавітно-цифрових даних, дворівневих зображень, аудіо-відео інформації.....	256
20.3. Організація БД в ТЧБ Галуа для ведення комерційної інформації в ГРКС....	260
20.4. Принципи організації багаторівневої рекурентної БД в базисі Галуа.....	265
СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ.....	267

СПИСОК СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

К	– коди;
ПНК	– перешкодонестійкі К;
ПС	– перешкодостійкі К (коректуючі К);
КК	– класи кодів;
БК	– 1 клас: блокові К;
РК	– рівномірні К;
РДК	– розділені К;
СГЛ	– систематичні (групові) К, лінійні загального виду;
ПКК	– підкласи кодів;
ЛБК	– підклас: лінійні блокові К з поелементним формуванням перевіркою групи;
ВНП	– К виправляючі незалежні помилки;
ВПП	– К виправляючі пакети помилок;
КПП	– К з перевіркою на палельність;
ТК	– типи кодів;
ЛКЗВ	– лінійні коди загального виду;
ЛАК	– лінійні арифметичні R–К;
РГК	– реєстровані К;
АН	– АН – К;
ЗК	– залишкові К;
КХ	– К Хемінга;
ЛЦК	– підклас: лінійні циклічні К з формуванням перевіркою групи;
CRC	– (cyclic redundancy check – К циклічної надлишкової перевірки);
СКК	– складені і композиційні К ;
БЧХ	– Боудза-Чоудхури-Хоквінгема К ;
РСК	– Ріда Соломона К корекції помилок недвійкові циклічні;
КП	– К з повторюванням;
ННЛ	– несистематичні нелінійні К;
РВК	– рівновагові К;
ІК	– інтерактивні К;
КК	– каскадні К ;
ККТ	– каскадне кодування турбо –К;
БК	– Бергера К;
АК	– Адамара К;
ХК	– Хаара К;
КПГ	– К поля Галуа;
СКГ	– Сигнально–коректуючі К в ТЧБ Галуа;
ПСК	– Позиційно-сигнальні коди
СРСК	– Симетричний рекурентний сигнальний код
НРСК	– Несиметричний рекурентний сигнальний код
СК	– сигнальний код каналу зв'язку (Манчестерський код);
СК1	– сигнальний код передавання "1" в каналах зв'язку;
G(0)	– код Галуа, який генерує нулі;
SK01	– сигнальний код нулів, коли біт- Галуа "1" передає кодом "11", а біт-Галуа "0" передає кодом "00";
SK02	– сигнальний код нулів, коли біт-Галуа "1" передає "10", а "0" - "00";
GK01	– сигнальні коди маніпуляції, в яких нулі нумеруються кодом Галуа;
GK02	– сигнальні коди маніпуляції в яких нулі нумеруються кодом Галуа по фронту спаду.
НЗК	– нерозділені –рівновагові К;
НК	– нерівномірні К;

МВГ	– Морзе (Вейля-Герке) К телеграфного зв'язку, нерівномірний;
ХКФ	– Хафмена К факсомільного зв'язку, нерівномірний;
НРК	– 2 клас: неперервні (рекурентні) К;
ЗГК	– згортаючі К;
ТЧБ	– теоретико-числові базиси;
Fur	– ТЧБ Фур'є;
Uni	– ТЧБ Унітарний;
Har	– ТЧБ Хаара;
Crg	– ТЧБ Крейга;
Rad	– ТЧБ Радемахера;
Gry	– ТЧБ Грея;
Crs	– ТЧБ Крестенсона;
Wal	– ТЧБ Уолша;
Gal	– ТЧБ Галуа;
I	– інформація;
Д	– дані;
БД	– бази даних;
БДГ	– бази даних в ТЧБ Галуа;
СК	– системи керування БД;
БКД	– банк даних (БД + СКБД);
БЗ	– бази знань;
СКБЗ	– системи керування БЗ;
БКЗ	– банк знань (БЗ + СКБЗ);
ІБД	– ієрархічна БД;
М	– мережеві БД;
Р	– реляційні БД;
О	– об'єктні БД;
ОО	– об'єктно-орієнтовані БД;
ОР	– об'єктно-реляційні БД;
НС	– напіструктуровані БД;
РБД	– розподілені БД;
ВБД	– віддалені БД телекомунікаційних систем;
БВ	– багатовимірні БД;
КД	– куби Д;
ПД	– простори Д;
НВ	– надвеликі БД;
КБД	– комп'ютерні БД;
ЕС	– експертні системи;
Ш	– штучний інтелект;
СД	– сховища даних;
СДГ	– сховища даних в ТЧБ Галуа;
ІРк	– ієрархічно-рекурентна БДГ;
ІРл	– ієрархічно-реляційна БДГ ;
ЛР	– лінійно-рекурентні БДГ ;
ОЛР	– однорівневі ЛР;
ДЛР	– дворівневі ЛР;
ТЛР	– трьохрівнева ЛР;
ЧЛР	– чотирьохрівнева ЛР;
БЛР	– багаторівневі ЛР;
ВР	– вертикально-рекурентні БДГ ;
ОВР	– однорівневі ВР ;
ДВР	– дворівневі ВР;

ТВР	– трьохрівнева ВР;
ЧВР	– чотирьохрівнева ВР;
БВР	– багаторівневі ВР;
КС	– комп'ютерні системи;
ГРКС	– глобальні розподілені КС;
ОРР	– об'єктно-реляційно-рекурентна модель БД ГРКС;
КЕ	– конгломерати епохальних СДГ ГРКС;
СККЕ	– системи керування конгломератами епохальних СДГ ГРКС.

Великий французький математик Еваріст Галуа і застосування його теорії у наш час

*Присвячується 200-річчю
від Дня народження
великого математика
Еваріста Галуа*



Більше двохсот років відділяє нас від того часу, коли народився Еваріст Галуа – видатний французький математик, заслуги якого визнані вченими всього світу. Його життя обірвалося дуже рано – у 20 років. Тепер Галуа – гордість французької науки, кращі риси якої відображені у його роботах, що майже на століття випередили розвиток фундаментальної математики.

Еваріст Галуа народився 25 жовтня 1811 року в Бур-ля-Рен - передмісті Парижа. У віці 12 років вступив у королівський коледж Луї-де-Граж. У 16 років Е. Галуа почав читати фундаментальні математичні твори, у тому числі мемуари Нільса Абелья про розв'язок рівнянь довільного степеня, де довів, що для рівнянь степеня 5 і вище розв'язки “в радикалах” неможливі. Ця тема захопила Еваріста, він почав власні дослідження і просунувся у теорії набагато далі. Галуа знайшов необхідну та достатню умову для того, щоб корені рівняння виражалися в радикалах. Найціннішим був не тільки цей результат, а й ті методи, за допомогою яких Галуа вдалося його отримати.

Уже в 17 років Е. Галуа опублікував свою першу наукову роботу в журналі “Annales de Gerdonne”. У 1829 році він вступає у Вищу нормальну школу, в якій провчився тільки один рік і був виключений за підтримку республіканців.

Але талант не допомагав визнанню Е. Галуа, тому що його рішення перевищували рівень професійності викладачів, виясненню його розумових та математичних висновків не сприяло також те, що він не старався чітко викладати їх на папері й часто пропускав очевидні для нього положення. На думку математиків, саме математика перетворила його зі слухняного учня у видатного вченого. Правда, не всі математики того часу відповідно оцінили його наукові праці.

Наприклад, про роботу Галуа в двох частинах схвально відгукнувся відомий математик Коші, якому вона була відправлена на рецензію. Цей рукопис був загублений Коші і не потрапив у Паризьку Академію на конкурс математичних робіт.

Негаразди у науковому житті Е. Галуа тривають. Мемуар про свої відкриття Е. Галуа посилає Фур'є для участі у конкурсі на приз Паризької Академії, але через кілька днів Фур'є раптово помирає, не встигнувши опрацювати цей текст. Приз Паризької Академії отримує Нільс Генрік Абель.

Стаття, відіслана Сімеону-Дені Пуассону, була повернута з наступною резолюцією: “...У всякому випадку, ми зробили все від нас залежне, щоб зрозуміти доведення пана Галуа. Його міркування та виклади не володіють ні достатньою ясністю, ні достатньою повнотою для того, щоб ми могли бути впевнені та вважати їх точними, тому ми не в змозі дати про них уявлення в цій доповіді”.

Чому саме ім'я Галуа і його теорія стали предметом широкомасштабних фундаментальних досліджень великої когорти вчених усього світу? Тому, що саме життя і творча діяльність Еваріста Галуа - приклад альтернативи класичній теорії математики - нескінченно малих приростів аргументів та функцій для інтегрального і диференціального числення. Ця теорія, яка два століття тому стала альтернативою, альтернативно розвивається у наш час.

У книзі Андре Дальма є коротка характеристика діяльності Еваріста Галуа: "...погано миряться з думкою, що геній може приєднатися до прогресивного руху народу. Щоб мати право відрізнятись від інших, учених, перш за все, повинен дати докази своєї безпечності. Якщо він не є цілком безпечний, то добиваються, щоб він став таким".

За спогадами вчителів Королівського коледжу, Е. Галуа був "не вживчивий, дивний і надмірно балакучий", мав "неабиякі здібності" та "досить нестандартні манери".

При житті Галуа не добився ніякої слави, хоча сам ніколи не сумнівався в цьому. Сучасні на той час математики не тільки не розуміли, що його роботи знаменують собою нову епоху в розвитку математики, але навіть не звернули на них уваги. Лише більш, ніж через півстоліття науковий світ оцінив оригінальність і глибину його мислення, а також властивий Галуа дар передбачення.

У наш час знаменитими стали не тільки спеціальні наукові проблеми, але й окремі прогнози Галуа, в яких передбачені нова система організації науки і комунікації вчених, котрими, як правило, завжди нехтували.

Від самого початку Галуа відмовлявся від шкільних підручників, автори яких мистецтво мислити підміняли мистецтвом вводити в оману за допомогою наборів певних слів. Поглинувши книгу Лажандра "Елементи геометрії" і роботи Лагранжа "Рішення численних рівнянь", "Теорія аналітичних функцій", "Лекції з теорії функцій" Галуа прийшов до ідеї і фундаментального поняття групи. Ознайомившись із роботами Ейлера, Гаусса і Якобі, Галуа висловив здивування методами розв'язків задач, якими користувалися викладачі. А викладачем математики був 33-хрічний професор Рішар, в якого у цей час вчився знаменитий математик Шарль Ерміт, котрий зберіг для людства рукописи робіт Галуа. Рішар мав велике задоволення відкривати таланти. Розв'язки математичних задач, які пропонував Галуа, захоплювали його. У записках Рішар зазначив: "Галуа працює тільки в вищих основах математики".

Шістдесят написаних рукою Е. Галуа сторінок відкрили світові ім'я цього воістину геніального вченого. Галуа цікавили насамперед не окремі математичні задачі, а загальні ідеї, що визначають увесь ланцюг усвідомлень і скеровують хід думок. Його доведення базовані на глибокій теорії, яка дає змогу об'єднати всі досягнення математики і визначити розвиток науки надовго вперед. Гільберт назвав цю теорію "визначенням певної архітектури понять симетрії".

Отже, за 20 років життя Е. Галуа встиг зробити відкриття, що прирівнюють його до найвидатніших математиків XIX століття. Розв'язуючи задачі з теорії алгебричних рівнянь, він заклав основи сучасної алгебри, вийшов на такі фундаментальні поняття, як група (Е. Галуа перший використав цей термін, досліджуючи симетричні групи) і поле (тепер кінцеві поля названі полями Галуа).

Відкриття Е. Галуа започаткували новий напрям – теорію абстрактних алгебричних структур. У наступні 20 років Келі та Жордан розвивали й узагальнювали ідеї Галуа, які якісно перетворили теоретичне обличчя всієї математики.

Огюст Шевальє, товариш Е. Галуа, і молодший брат Е. Галуа Альфред послали останні роботи Галуа Гаусу і Якобі, але відповіді не отримали. Тільки у 1843 році відкриття Галуа зацікавили Ліувілля, який опублікував і прокоментував їх.

Роботи Галуа з теорії груп ознаменували кінець передісторії і початок істинної історії математики.

"Я займаюсь аналізом аналізу. Я хочу підкорити обчислення своїй волі, згрупувати математичні операції, навчитись їх класифікувати за ступенем труднощі, а не за зовнішніми прикметами – це завдання математиків майбутнього", - пише Галуа.

Поняття групи визначається як сукупність абстракцій, що мають певні спільні властивості. Наприклад група цілих чисел, група простих чисел, група невід’ємних залишків тощо.

Абстракціями в групах можуть бути операції. Саме цей випадок вивчив Галуа. Суттєво, що ні самі абстракції, ні операції над ними ніяк не конкретизовані. Вивчаючи властивості та користуючись формальними правилами перетворення груп, можна передбачити їхні властивості, а значить, у принципі, вирішити конкретну задачу.

Тут проявляється альтернатива мислення Галуа, про що Паскаль сказав: “Дійсний дослідник відкриває передовсім не нові об’єкти, а нові зв’язки між ними, які визначають можливі властивості відомих і невідомих об’єктів”.

Таким чином, геніальне прозріння Галуа полягає у розробленні однієї з найфундаментальніших альтернативних теорій математики, головне значення яких полягає у пізнанні того, що ідея груп симетрій, яка була виключно пов’язана з геометрією, грає фундаментальну роль у всій математиці й взагалі у природознавстві. У тому числі групою Галуа класичної механіки є група Галілея, а механіки теорії відносності – група Лоренца. Класичними стали відомі групи Абеля й абстрактна теорія груп О. Ю. Шмідтта.

А як протікає розвиток ідей Галуа у наш час? Кажуть, що немає нічого практичнішого, ніж хороша теорія.

Проте в школах теорію полів Галуа навіть не згадують, у технічних вузах – глибоко не вивчають. На математичних факультетах університетів теорію груп вивчають у загальному розділі теорії чисел. Останній раз праці Галуа були видані російською мовою в 1935 році й перевидані у 2003 році.

Я тривалий час вивчав різні теоретико-числові базиси та системи числення, в т.ч. мінус двійкову та систему залишкових класів, яка базується на теорії полів Галуа і намагався боротись із недоліками двійкової системи, яка з легкої руки Фон-Неймана і К. Шеннона зайняла провідне місце у розвитку комп’ютерних систем та цифрових мереж передавання інформації.

Потрібно було об’єднувати канали, стискати повідомлення, захищати дані від завад у лініях зв’язку і помилок в обчислювальних машинах. Потрібний був якийсь універсальний, гнучкий та ефективний спосіб кодування чисел, повідомлень і сигналів на всіх стадіях руху інформації від джерел даних, об’єктів управління – до процесорів опрацювання даних та серверів, їх компактного зберігання на носіях і в базах знань.

Двійкова система числення не могла розв’язати цю проблему в такому розумінні, бо виявилася частковим випадком полімодульної системи залишкових класів.

Після кількох лекцій професора І. Я. Акушського з теорії залишкових класів, які я прослухав у 1970 році у Санкт-Петербурзі в Інституті точної механіки і оптики, я внутрішньо крикнув “еврика” і детально простудіював книжку І. Я. Акушського і Д. І. Юдіцкого “Машинна арифметика в залишкових класах”. Згодом – книгу академіка В. М. Амербаєва “Основи машинної арифметики комплексних чисел”, В. М. Торгашева “Коректуючі коди в системі залишкових класів”, Б. М. Коляди “Теорія і застосування математичних машин”.

Система числення залишкових класів, так звана СЗК, базована на тому, що в кожному розряді число має окремий модуль, тобто свій набір цифр – залишків, ансамбль яких по простому модулю якраз утворює групу у полі цілих чисел, яке є одночасно полем Галуа.

Фундаментальна математична основа СЗК базована на доведенні кількатисячолітньої давності так званої Китайської теореми про залишки, математичною основою якої є властивості операцій над ідеалами.

У наш час числення залишкових класів знайшло ряд надзвичайно ефективних застосувань при побудові спеціалізованих супершвидкодіючих обчислювальних машин, наприклад, до 30-ти мільярдів операцій на секунду, з надвисокою надійністю і нечутливістю до відмов обладнання та помилок обчислень.

Є ряд аргументів, які теоретично й експериментально дослідив професор А. Канн, в котрого я прослухав курс лекцій з біоніки, що в пучках нейронів використовується кодування імпульсних сигналів у базисі груп, полів Галуа та залишкових класів.

Застосування теорії груп привело до сучасних успіхів у розробці швидкодіючих алгоритмів і процесорів цифрових перетворень Фур'є на основі чисел Мерсена, Ферма та Фібоначчі.

Я детально досліджував теорію канонічних форм прямого і зворотного перетворення залишкових класів. У тому числі: цілочисельне, нормалізоване і знайдене мною досконале перетворення СЗК.

На базі цих перетворень розроблені та впроваджені у техніку ефективні методи і технічні засоби стискування та надійного передавання цифрової інформації на низових рівнях комп'ютеризованих систем управління.

У той самий час числення залишкових класів виявилось безпосереднім чином зв'язане з перетвореннями комплексних чисел у базисах Фур'є, Гільберта та Ейлера. Це числення виявилось частковим випадком фундаментальної теорії систем базисних функцій, які саме й утворюють найдосконаліші групи.

У порядку наростання загальності довелося вивчити базисні функції Радемахера, Хаара, Уолша, Віленкіна-Крестенсона.

Стало зрозумілим, що базис Радемахера – це прямокутні симетричні функції, які породжують двійкову систему числення. Функції Хаара породжують розрядно позиційні системи числення. Функції Уолша породжують відомі коди Грея і Хеммінга. Вибірка функцій Уолша, для яких властива операція рекурсії, породжує двійкові коди і числення Галуа. Сюди належать популярні коректуючі коди Голя, Боуза-Чоудхурі-Хоквінгема та інші, що базовані на діадних операціях зсуву. Цим базисом породжуються так звані шумоподібні сигнали (М-сигнали) та коди Баркера, які знайшли винятково необхідне застосування у космічній галузі, на наддалекому підводному зв'язку. На принципах застосування кодів Баркера побудовані сучасні сотові комп'ютерні радіомережі та системи передавання повідомлень у каналах з інтенсивними завадами.

Числення залишкових класів виявилось досконалим, частковим випадком функцій Віленкіна-Крестенсона, що склали основу базисних перетворень Галуа.

Така фундаментальна генеалогія числення базису Галуа віщувала надію надзвичайно широкого застосування у різних галузях сучасної науки – молекулярній хімії, механіці, обчислювальній техніці, кібернетиці, теорії сигналів, теорії інформації, молекулярній біології, соціології, екології, енергетиці та інших галузях знань.

Числення Галуа виявилось справді плідним ельдорадо на сучасному етапі розвитку кібернетичної науки і техніки. Важливою ідеологічною властивістю числення Галуа є нове ефективне рішення задач, які стандартно вважаються неймовірними та супроводжуються психологічними бар'єрами і дуже часто недовірливими запитаннями й константаціями типу: «Це бути не може тому, що воно не може бути».

Розглянемо ряд ефективних прикладів застосування числення Галуа при розв'язуванні задач у галузі автоматики і цифрових обчислювальних систем.

Перше – це кодові шкали Галуа. В наш час розроблені однобітові нереверсивні, двохбітові реверсивні та трьохбітові позиційні змінно-якісні кодові шкали Галуа, що відкривають нові широкі можливості побудови високоточних вимірювачів переміщень, кута повороту і позиціювання транспортних засобів в одно-, дво- і трьохмірних декартових, а також полярних координатах. Особливістю таких шкал є незалежність числа кодових доріжок від довжини кодової шкали.

Друге – це аналого-цифрові перетворювачі Галуа паралельно-послідовного скануючого типу. Унікальною властивістю таких АЦП є висока регулярність обчислювального середовища, дворазове зменшення числа компараторів стосовно до наявних структур і практичне виродження дешифратора, що визначає значну перспективу їх реалізації на базі великих інтегральних схем.

Третє – це формування місцевого, регіонального або всесвітнього Галуа-часу. Галуа-час - це генерація в ефір кратnoseкундних бітів Галуа, сприймаючи які, можна синхронізувати всі електронні годинники, створити зелені хвилі світлофорів у містах, керувати виробничими машинами, технологічними процесами та персональним обладнанням.

Четверте – це алгоритми і процесори стискування інформації, які порівняно з адаптивним кодуванням за ефективністю переважають останні у два-три рази. Основу таких алгоритмів становить рекурентна зв'язність елементів поля Галуа, що дає змогу нумерувати активні цифрові відліки одним або двома бітами Галуа незалежно від довжини послідовності та числа пропущених неактивних цифрових повідомлень.

П'яте – це побудова електронної пам'яті з колективним паралельним доступом на базі адресації Галуа, що дозволяє багатьом віддаленим абонентам із нешвидкодiючим обладнанням одночасно звертатися до бази даних і одночасно отримувати файли даних нешвидкісними каналами зв'язку. Галузь застосування – інформаційно-пошукові системи, бази знань, довідкові системи в бібліографії, медицині та ін.

Шосте – це побудова інтелектуальних сенсорів і перетворювачів енергетичних параметрів з інтегрально-імпульсним кодом Галуа. На основі таких перетворювачів нині впроваджені необхідні комерційні комп'ютерні системи контролю та обліку енергоспоживання цехів і промислових підприємств.

Сьоме – це розвиток теорії обчислень у полі Галуа, побудова нового покоління процесорів, обчислювальних систем та низових обчислювальних мереж на базі сформованої нами вертикальної інформаційної технології в базисі дискретних перетворень Галуа.

Цю нумерацію можна було б продовжити. Але найвизначнішим може бути застосування ідей Галуа до розкриття фундаментальних теоретичних основ побудови кодів генетичної основи всього живого – коду молекули ДНК.

Справді, якщо позначити букви коду ДНК А, Т, G, С – відповідно 0, 1, 2, 3, то ми одержимо не що інше, як послідовність Галуа за модулем чотири. Ми вже генеруємо за допомогою комп'ютерів синтезовані ДНК-подібні послідовності. На стадії розроблення теорія спірально-циліндричних полів Галуа. Наприклад: у полі Галуа немає повторень і ДНК не має повторень. Код Галуа має липкі кінці і ДНК має липкі кінці. У полі Галуа будь-яку помилку можна виявити за допомогою ключів і в ДНК рибосома зупиняється в процесі синтезу білка, коли виявляє недопустиму помилку. Код Галуа можна розмежувати, наприклад, на межі букв АТ і потім абсолютно точно відновити попередню послідовність. Подібно, в ДНК рестриктаза розриває біологічну структуру на границі конкретних букв, а лігаза абсолютно точно відновлює попередню структуру, яка спроможна після цього правильно створити організм.

Кожна амінокислота кодується одним або кількома триплетами, причому всі триплети – кодони, число яких рівне Шеннонівському числу $4^3=64$. Деякі з кодонів використовуються тільки як розділові знаки, а промотори – як мітки блокової синхронізації. ДНК – це яскравий приклад досконалої групи симетрій, і вона не може мати іншу теоретичну основу, крім теорії Галуа.

Автори висловлюють подяку академіку НАН України, професору О. В. Палагіну – фундатору теорії розподілених комп'ютерних систем за ідею інтегрованого підходу до теорії джерел інформації та міжбазисних перетворень, рецензентам: члену-кореспонденту НАН України, професору В. К. Задіраці, професору А.О.Мельнику та професору М. П. Диваку за цінні рекомендації, науковим співробітникам Карпатського державного центру інформаційних засобів і технологій Технічного центру НАН України, директору Інституту мікропроцесорних систем керування об'єктами електроенергетики, доценту І.О.Сабадашу, директору Інституту проблемно-орієнтованих комп'ютерних систем, доценту М.І.Чирці, працівникам кафедри спеціалізованих комп'ютерних систем факультету комп'ютерних інформаційних технологій Західноукраїнського національного університету за участь в обговоренні та допомозу в оформленні книги.

Щиро дякуємо науковому вчителеві заслуженому діячеві науки і техніки д.т.н, професору А.М.Лучуку за те багатство ідей та щедрий науковими результатами шлях.

Монографія орієнтована на фундаментальні засади кібернетики, теорії та методології інформатики, інформаційної теорії ДНК, розподілених комп'ютерних систем та комп'ютерної криптології, викладені в наукових працях академіків НАН України В.С.Дейнеки, І. В. Сергієнка, О. В. Палагіна, член-кореспондентів НАН України Б.М.Малиновського, В.П.Боюна, В. К. Задіраки, А.М.Гупала, професорів С.Г.Буніна, А.О.Мельника та інших вчених і системно охоплює результати сумісних досліджень учнів наукової школи Я.М.Николайчука: доктора технічних наук, професора Л. Б. Петришина, В. В. Яцківа, М. М. Касянчука, Н.Я.Возної; кандидатів технічних наук С. М. Іщерякова, Г.Я.Ширмовського, С. І. Мельничука, А. І. Сегіна, Н. Г. Яцків, І. М. Лазаровича, Н.Д.Круцкевича, О. М. Заставного, І.Р.Пігуха, Т. М. Гринчишина, І. З. Якименка, І. О. Погонця, І.Б. Албанського, О. І. Волинського, П. В. Гуменного, Т.О.Заведюк, А. Р. Воронича, С. В. Івасьєва та аспірантів: В. В. Шаряка, А.Я.Давлетової.

ПЕРЕДМОВА

На початку XXI „Золотого століття” високорозвинуті країни світу, перебуваючи у стадії переходу від індустріального суспільства в 1860–1960 роках, до постіндустріальної фази свого розвитку – інформаційного суспільства, а починаючи з 1960–80 років основою якого стала глобальна інформаційна інфраструктура. Майбутня фаза „Золотого століття” розвитку суспільства з 2013–37 років передбачає стрімкий розвиток біо-нано технологій. При цьому спостерігається потужний розвиток та вдосконалення інформаційних технологій збору, формування, передавання, опрацювання, перетворення, захисту та зберігання інформації у комп’ютерних системах та базах даних.

Значну роль в успішному вирішенні даної проблеми посідає досвід розробки та застосування методів кодування даних у ТЧБ Галуа та їх застосування при організації баз даних. Сучасні методи кодування даних передбачають формування та обробку інтенсивних потоків інформаційних даних в реальному масштабі часу. До складу таких даних входять діагностичні, технологічні та техніко-економічні дані, а також формалізовані моделі джерел інформації.

Технологічні дані, що формуються на рівні сенсорів спеціалізованих та універсальних комп’ютерних засобів у вигляді структуризованих фреймів, використовуються на різних рівнях розподілених комп’ютерних систем для представлення даних операторам, їх передавання по каналах зв’язку, формування архівів баз даних, а також управління виробничими процесами в реальному часі. Моделі об’єктів управління джерел інформації, побудовані на основі структуризованих даних, використовуються для аналізу відхилень технологічних процесів від норми, виявлення передаварійних та передбачення аварійних ситуацій, кодування даних, стиснення даних, їх захисту від помилок та несанкціонованого доступу.

Сучасні технології організації баз даних та експертних систем штучного інтелекту є одним із визначних факторів вдосконалення процесів зберігання та доступу до великих масивів даних. Потужність сучасних баз даних базується на результатах досліджень та технологічних розробках, які отримані на протязі кількох останніх десятиліть. Керування базами даних виконується спеціалізованими програмами–системами керування базами даних, які є ефективним інструментом збору реєстрації та використання даних. Особливим видом баз даних є розподілені бази даних, важливими характеристиками яких є ступінь паралалелізму операцій та швидкого пошуку даних. Слід зауважити, що технології розподіленої обробки інформації ускладнюють всі компоненти і заставляють переглядати їх структури аж до базових елементів. При чому, в багатьох розподілених базах даних затрати на підтримку телекомунікацій можуть переважити вартість обчислювальних процесів та програмно-технічного забезпечення.

Тому при дослідженні та проектуванні глобальних розподілених комп’ютерних системах найважливішу роль отримують задачі компактного кодування даних, а також високої швидкості обміну та доступу до даних. Отже задача вдосконалення архітектури та методів кодування інформації в базах даних є актуальною у промислово–господарському аспекті. В даний час у теорії та техніці цифрового опрацювання даних спостерігається ефективне використання різних теоретико – числових базисів.

Аналіз використання теоретико-числових базисів в інформаційних технологіях показує виключно широке застосування базису Радемахера, який породжує двійкові кодові системи числення. В той же час світовий досвід застосування теоретико – числових базисів для швидкодіючих теоретико – числових перетворень, які реалізують числові згортки та цифрову обробку сигналів, показує що значне прискорення швидкості процесорів та спрощення програмних алгоритмів забезпечують базиси Уолша, Крестенсона та Галуа.

Оптимізація сучасних технологій побудови баз даних є одним із вирішальних факторів забезпечення ефективного зберігання корпоративної інформації та використання даних користувачем. Реляційна модель бази даних в даний час отримала широке застосування і ефективно використовується при створенні потужних баз даних . Перевагою реляційних баз

даних по відношенню до ієрархічних є забезпечення можливості формалізації перетворення даних на основі алгоритмів реляційної алгебри.

Вдосконалення сучасних технологій побудови БД є одним із вирішальних факторів забезпечення компактного та ефективного зберігання корпоративної інформації. Одним із перспективних напрямків теорії і організації БД є реалізація доступу до даних на основі багаторівневих рекурентних архітектур БД.

При цьому особливістю використання базису Галуа при організації БД є можливість не тільки компактного кодування даних, але і організований захист даних від помилок та несанкціонованого доступу.

Одною з переваг сучасних комп'ютерних систем є просторова та часова роздільність. На основі ТЧБ Галуа перспективним є створення інформаційної технології побудови: ієрархічно-рекурентної, ієрархічно-реляційної, однорівневої, дворівневої та багаторівневої лінійно-рекурентної, однорівневої, дворівневої та багаторівневої вертикально-рекурентної, реляційно-рекурентної, об'єктно-реляційно-рекурентної, баз даних глобальних розподілених комп'ютерних систем, та конгломератів епохальних сховищ даних комп'ютеризованих систем.

Успішне вирішення названого класу задач може бути досягнуто за рахунок раціонального використання математичного апарату та системних можливостей ТЧБ, Унітарного, Хаара, Крейга, Грея, Крестенсона, Уолша, Галуа.

Значний вклад у розвиток названого напрямку вдосконалення інформаційних технологій формування та цифрової обробки даних теорії та архітектури баз даних внесли відомі закордонні вчені: Дж. Мартін, Ахо, Гарсія-Молина, Гектор, Ульман, Д. Джефри, Видом, Дженнифер Дейт К.Дж., Дженнингс Роджер., Джефф Когевелл, С.М. Диго, Томас Коннолли, Бег. Каролин Т.Тиори, Дж. Фрай, Д.Мейер, Э.Берлекэмп, Р.Блейхаут, В.Э.Вольфенгаген, У.Питерсон, Э. Уэлдон, Т.Кассаами, Н.Токура, Е.Ивадари Я. Инагаки.

У зв'язку з освоєнням потужної комп'ютерної техніки в останні роки ефективно почали реалізуватися складні алгоритми методів кодування БД на основі фундаментальних досліджень вчених – класиків України в галузі інформаційних технологій організації БД: А.М. Андреев, В.Ю.Биков, Н.Т.Березюк, А.Г.Андрущенко, С.С.Мощицкий А.Ф.Верлань, Ф.Е.Коваленко, Д.Г.Валеев, Н.И.Вюнова, В.А.Галатенко, С.Г.Глушков., Д.В.Ломотько., С.Д.Кузнецов., Мельничук С.І., Таянов С.А., С.І.Мельничук, В.В.Пасічник, В.А.Резніченко., Я.М.Николайчука, И.В.Кузьмин, В.И. Ключко, В.А. Литвин.

Одним з перспективних напрямків розвитку теорії технології організації БД є використання ТЧБ Галуа. Теоретичною основою для розвитку ефективного рішення такого класу задач необхідно відмітити роботи В.В.Пасічника, В.А. Резніченка, В.А. Головка, Я.М. Николайчука та інші.

ВСТУП

Порівняння БД із файловими системами. Взаємопов'язаність даних

Локалізація певної групи даних програми загалом полегшує доступ до інших груп даних цього ж застосування. Внаслідок орієнтації БД на велику кількість сеансів використання виникає необхідність у підтримці різноманітних зв'язків між даними. Розглянемо приклад, який ілюструє вказану властивість у порівнянні з простою файловою системою.

Припустимо, що є мішок з різнокольоровими кульками. Завдання полягає в тому,

Аби знайти кульку певного кольору, витягаючи їх по одній з мішка. Якщо операції витягування кульок з мішка є повністю незалежними, то цей приклад є аналогом технології роботи в межах простої файлової системи з послідовними доступом.

Припустимо, що потрібно знайти зелену кульку. Експериментатор наосліп вибирає й виймає з мішка по одній кульці, доки не побачить шуканий колір. При цьому попередня кулька ніяк не впливає на вибір наступної. Можливий інший спосіб пошуку. Експериментатор витягує з мішка червону кульку, до якої прив'язані кілька мотузок різних кольорів, зокрема і зелена. Тоді наступна кульку він витягує за зелену мотузку, і це буде шуканий об'єкт. Якщо ж червона кулька не матиме зеленої мотузки, то експериментатор може потягти за жовту мотузку і витягти жовту кульку, вже від неї шукати зелену.

Для такої роботи необхідно, щоб експериментатор знав структуру зв'язків між кульками. Якщо припустимо, що кульки зв'язані між собою різнокольоровими мотузками, то у такому випадку пошук можна було б прискорити, використавши на черговому кроці мотузку потрібного кольору. В наведеному вище прикладі продемонстровано технологію роботи зі взаємопов'язаними даними, характерну для БД.

Мінімальна надлишковість

Вимоги мінімізації надлишковості полягає в тому, що в базі має зберігатися мінімальна кількість копій одних і тих самих даних. Це необхідно у зв'язку з орієнтацією БД на кілька застосувань. Надлишкові копії використовуються для підтримки зв'язків між даними.

Незалежність даних від програм

Незалежність даних від програм означає можливість зміни структури даних без зміни програм, що її використовують. Під це поняття підпадає й рівень само-інтерпретованості даних. Для файлової системи він найнижчий, оскільки дані файлів інтерпретуються виключно через прикладні програми, що їх використовують. БД, окрім самих даних, зберігає також їхні описи даних та процедури їхньої обробки.

Цілісність та захист від неавторизованого доступу

Під цілісністю БД розуміють несуперечність (відповідність певним умовам) даних,

Що в ній зберігаються. Наприклад, для кадрових даних рік народження співробітника не може бути більше за рік призначення на посаду або поточний рік.

Для уникнення суперечливих ситуацій при модифікації БД співвідношення між даними контролюються спеціальними засобами підтримки цілісності БД. Умовами, що накладаються на дані та на яких дотримуються за будь-котрих оновлень, розробляються спеціальною службою—адміністраторами БД (АБД), а СКБД надають необхідні інструментальні засоби.

Оскільки БД орієнтована на широке кола застосувань, то зрозумілим є існування засобів захисту від неавторизованого доступу (навмисного чи ненавмисного) користувачів до даних. Серед них – система паролів та ідентифікацій користувачів, а також розподіл даних і користувачів на групи з різними правами.

Крім наведених, БД мають ще низку переваг порівняно з файловими системами.

Інтегроване зберігання даних. Сукупність даних розглядається як єдине цілісне, незалежно від характеру використання та способу зберігання. Тобто БД є інформаційною моделлю всієї предметної області.

Централізоване керування. Передбачає адміністрування БД, що дає можливості забезпечити ефективне зберігання даних, усунення їхньої суперечності, підтримки єдиної

політики у використанні даних та необхідного рівня їхньої безпеки збалансування суперечливих вимог різних користувачів.

Ефективне керування доступом до даних. СКБД надає ефективні механізми керування доступом до даних, незважаючи на складність структур, у яких вони зберігаються.

Скорочення часу розробки програмних систем. За наявності засобів створення і обробки даних, які надаються СКБД, усі зусилля розробників програмних систем концентруються на розв'язанні прикладних задач.

Спільне використання даних. Передбачає одночасний доступ до інформації багатьох користувачів та ефективний розподіл ресурсів.

Відновлення бази даних. Жодна непередбачувана непередбачена ситуація – навіть перебоїв програмного чи апаратного забезпечення – не може вивести з ладу БД.

Дотримання стандартів. Формати зберігання даних, технологія їхньої обробки, вхідні й вихідні форми можуть бути розроблені з урахуванням єдиних стандартів підприємства або або інших нормативних документів.

Функції адміністратора бази даних

Функціонування складної структури на зразок СКБД потребує наявності певних служб, відповідальних за підтримання експлуатаційних характеристик БД. Чи не найважливішою з них є служба адміністратора БД, функції якого перелічені нижче.

Проектування й розробка описів даних на різних рівнях узгодження.

Розробка структур зберігання і стратегій доступу до даних згідно з вимогами ефективного (чи економічного) зберігання, швидкої обробки та виведення даних за бажанням користувача.

Реструктуризація та реорганізація БД у випадку зміни вимог до характеристик зберігання і обробки даних.

Проектування і застосування механізмів захисту даних.

Реєстрація користувачів (імен, паролів), визначення їхніх прав доступу та повноважень.

Розробка і використання механізмів резервного копіювання даних та їхнє відновлення під час перебоїв.

Настроювання роботи БД (підвищення продуктивності механізмів обробки даних, підтримання планової надлишковості, забезпечення ефективності їхнього зберігання).

Систематичне наглядання за використанням БД.

Основні вимоги до систем керування базами даних

До сучасних СКБД висуваються такі вимоги:

Простота і гнучкість створення застосувань. СКБД має значно полегшувати процес створення і супроводу застосувань, що працюють з БД.

Багаторазове й багатоаспектне використання даних. Користувачі, які інакше розуміють одні й ті самі дані, мають отримати змогу використовувати їх по-різному.

Простота і гнучкість зміни, розширення й настроювання бази даних. Будь-які зміни в БД, додавання до неї інформація чи її розширення повинні виконуватися незалежно від застосувань, що вже існують і використовують БД. До того ж ці операції мають виконуватися максимально просто й ефективно.

Ефективність і гнучкість зберігання й обробки даних. СКБД має забезпечувати можливість простого й ефективного збільшення обсягів даних без порушення наявних способів їхнього використання, а також легкого використання реорганізації і реструктуризації даних. СКБД повинно підтримувати заплановану надлишковість даних або надавати можливість їх знижувати. Запити на отримання даних мають виконуватися зі швидкістю, потрібною для їхнього ефективного використання.

Низька вартість зберігання й використання даних. СКБД мають забезпечувати нижчу вартість зберігання і використання даних, ніж будь-які інші інформаційні системи.

Захист від несанкціонованого доступу, викривлення і знищення. Система має забезпечувати необхідний рівень захисту даних. Інформація має бути захищена від перебоїв,

впливу форс-мажорних ситуацій, а також від некомпетентного чи зловмисного звернення до даних осіб, які можуть оновити, викривити або ж видалити їх.

Підтримання необхідного рівня незалежності даних. Як правило, від СКБД вимагається наявність механізмів підтримання логічної та фізичної незалежності даних.

Підтримання необхідного рівня цілісності даних. Засоби опису і підтримання обмежень цілісності мають бути достатніми для опису правил, законів і обмежень, що діють у предметній області. Якщо ті чи інші обмеження не підтримуються безпосередньо, то СКБД має надавати можливості для їхнього опису і підтримання через відповідні засоби програмування.

Розвинені засоби адміністрування. Передбачається, що СКБД надає всі необхідні інструменти для підтримання функцій адміністрування БД.

Архітектура баз даних

Термін архітектура, може мати різні тлумачення. Наприклад, коли вказуються функціональні модулі системи й способи їхньої взаємодії, йдеться про функціональну архітектуру. Спосіб реалізації функцій системи, її компоненти та взаємозв'язки між ними фіксує архітектура реалізації системи. У цьому контексті можна також згадати архітектуру технічних засобів систем. Говорячи ж про термін „архітектура БД”, ми матимемо на увазі архітектуру інформаційного забезпечення.

Архітектура БД у перше була специфікована дослідницькою групою ANSI/X3/SPARC Study Group on Data Base Management Systems (ANSI – Американський національний інститут стандартів, X3 – його комітет обчислювальної техніки й обробки інформації, SPARC – підкомітет ANSI/X3 з планування стандартів ANSI). Метою дослідницької групи було визначення галузей і технологій баз даних, в яких було б доречно проводити стандартизацію, а також вироблення рекомендацій для роботи в кожній із таких галузей. Група дійшла висновку, що, можливо, єдиним аспектом систем баз даних, який можна стандартизувати, є інтерфейси. Нею було докладено чимало зусиль для визначення загальної архітектури системи бази даних. Запропонована цією групою архітектура стала класичною та є актуальною й донині.

Основною ідеєю специфікації ANSI /SPARC є виділення трьох архітектурних рівнів баз даних, а саме: зовнішнього, концептуального та внутрішнього.

Концептуальний рівень

На концептуальному рівні здійснюється інтегрований опис предметної області, для якої розробляється БД, незалежно від її сприйняття окремими користувачами та способів реалізації в комп'ютерній системі. Дамо означення основних понять, що використовуються на концептуальному рівні.

Предметна область (ПО) – частина реального світу, для якої здійснюється концептуальне моделювання.

Концептуальна модель ПО – формальне зображення сукупності думок, які характеризують можливі стани ПО, а також переходи з одного стану в інший (виключно з класифікацією наявної у ПО сутностей, чинних правил, законів, обмежень, тощо).

Концептуальне моделювання ПО – процес побудови концептуальної моделі ПО, яка б відображала ПО з урахуванням вимог, висунутих до цього процесу.

Концептуальна схема – фіксація концептуальної моделі ПО засобами конкретних мов моделей даних. У СКБД концептуальна модель подається у вигляді концептуальної схеми.

Опишемо властивості концептуальної моделі (схеми) й характерні особливості концептуального моделювання.

Спільне та однозначне тлумачення предметної області зацікавленими особами. До розробки складної бази даних залучається великий колектив: експерти, системні аналітики, проєктувальники, розробники, ті, хто впровадження і супроводом. Усі вони повинні однозначно розуміти, чим є ПО, в чому зміст використаних понять, як вони взаємопов'язані між собою, які обмеження висуваються до моделі ПО тощо. Спільність понять має забезпечувати концептуальна модель.

Концептуальна схема відображає лише концептуально важливі аспекти ПО, виключаючи будь-які аспекти зовнішнього або внутрішнього відображення даних. Ця

модель не повинна відображувати конкретні потреби окремих користувачів або застосувань. Вона має фіксувати, чим є ПО в цілому, а не з точки зору інтересів або потреб користувачів. Для отримання цілісного уявлення про ПО її модель має інтегрувати думки, погляди та інтереси окремих користувачів, але саме інтегрувати, а не виражати їхні конкретні побажання.

Визначення допустимих меж еволюції бази даних. Саме через концептуальну схему зовнішні дані відображаються на внутрішні, й навпаки. У такий спосіб створюється єдина основа для опису даних і підтримки цих відображень.

Забезпечення незалежності даних. Наявність відображень концептуальний–зовнішній, концептуальний–внутрішній дає змогу вирішувати проблему логічної та фізичної незалежності даних. Будь-які зміни в тій чи іншій зовнішній моделі не повинні спричиняти зміни в концептуальній або внутрішній моделях. У цьому випадку має змінитися тільки відповідне відображення „концептуальний–зовнішній”. Аналогічно, будь-які зміни у внутрішній моделі не зачіпають концептуальну модель і моделі зовнішнього рівня, а тільки приводять до змін відображення „концептуальний – внутрішній”.

Централізоване адміністрування. Саме через концептуальну схему здійснюється адміністрування баз даних.

Стійкість. Концептуальна схема не має піддажуватися до вимог тих чи інших користувачів (зовнішній рівень) або до вимог зберігання даних (внутрішній рівень). Будучи моделюю ПО, вона має змінюватися тільки тоді, коли входить у суперечність з нею.

Існує багато мов, які претендують на роль мов концептуального моделювання ПО. Найпопулярнішими і широкоживаними є мови, що належать до класу так званих графічних мов, які оперують поняттями „сутність – атрибут – зв'язок”. (Entity-Relationship language).

Зовнішній рівень. Через зовнішній рівень користувачі та застосування отримують доступ до бази даних. Мета зовнішнього рівня – видати користувачу /застосуванню лише ті дані, які йому потрібні (а отже, до яких дозволений доступ) і в потрібному вигляді. Це індивідуальний рівень користувача, яким може бути кінцевий користувач, програміст чи застосування. Кожен з них має свою мову спілкування: для кінцевого користувача – це спеціальна мова запитів, для програміста – одна з мов програмування, розширена командами звернення до СКБД, для застосувань – це як правило, певний стандартний інтерфейс звернення до бази даних через СКБД.

Зовнішня модель – це засоби зображення концептуальної моделі ПО з урахуванням інтерфейсів конкретних користувачів або застосувань. Кожна зовнішня модель подається в СКБД у вигляді зовнішньої схеми.

Зовнішній рівень виконує такі функції.

Забезпечує зображення даних зручним для людини або застосування способом. Ступінь незалежності зовнішнього зображення від концептуального рівня визначається потужністю засобів опису відображення „концептуальний – зовнішній”. Сприяє вирішенню проблеми безпеки (захисту) даних. Надаючи користувачу лише ті дані, що його цікавлять, ми залишаємо поза межами його доступу решту даних.

Сприяє вирішенню проблеми логічної незалежності даних. Це досягається завдяки відображенню „концептуальний – зовнішній”, що встановлює відповідність між концептуальною схемою і конкретною зовнішньою схемою. Потужність його засобів визначає ступінь логічної незалежності застосувань від даних.

Внутрішній рівень

Внутрішня модель є відображення концептуальної моделі ПО з урахуванням способів зберігання даних і методів доступу до них. Внутрішня модель відображається в СКБД у вигляді внутрішньої схеми. Внутрішня модель – це не модель фізичної пам'яті з характеристиками конкретних пристроїв зберігання даних (циліндри, доріжки, тощо); вона описується у вигляді нескінченної абстрактною лінійної пам'яті, яка може структуруватися за допомогою інших абстрактних понять на зразок блоків, кластерів, індексів тощо.

Доступ до фізичної пам'яті надається за допомогою опису відображень внутрішньої моделі на фізичну пам'ять операційної системи.

Загалом внутрішній рівень виконує такі функції.

Забезпечує налаштування бази даних для підвищення продуктивності обробки даних, опису й підтримки планової надлишковості.

Дає змогу описувати й підтримувати структуру зберігання та методи доступу.

Сприяє вирішенню проблеми фізичної незалежності даних: зміни у внутрішній схемі не повинні призводити до змін у зовнішній схемі.

Сприяє вирішенню проблеми безпеки (захисту) даних.

Вирішує проблему відображення даних на структури ОС, у яких дані зберігаються (до таких структур належать зокрема файли).

Відображення

Відображення зовнішнього рівня на концептуальний і концептуального рівня на внутрішній. Відображення „зовнішній – концептуальний” визначає відповідність між зовнішнім рівнем і концептуальним. Незалежність зовнішньої схеми. Відтак і ступінь логічної незалежності даних, обумовлюється потужністю засобів опису цих відображень. Тобто можна описувати або змінювати зовнішню схему тільки в тих межах, які допускає це відображення.

Подібне можна сказати і про відображення „концептуальний –внутрішній”, яке встановлює відповідність між концептуальною і внутрішньою моделями. Потужність його засобів визначає ступінь фізичної незалежності застосувань від даних. Будь-які зміни у фізичній структурі не повинні призводити до змін у концептуальній моделі – змінюється лише відображення „концептуальний –внутрішній”.

За створення і ведення схем усіх рівнів (концептуальної, зовнішньої і внутрішньої)

А також відображень відповідає адміністратор баз даних. Окрім того, він виконує Є багато інших функцій.

Функції систем керування базами даних

Системи керування базами даних надає засоби для здійснення таких операцій.

Визначення даних (зовнішні, внутрішні і концептуальна схеми) та відображень у певній формальний спосіб і реалізація цих визначень у вигляді відповідних об'єктів.

СКБД має обробляти вказівки мови опису даних для опису концептуальної схеми, підмови опису даних для опису зовнішніх схем і мови опису збережених даних для опису внутрішньої схеми.

Маніпулювання даними. СКБД має сприймати, інтерпретувати й обробляти запити користувачів на вибирання, оновлення наявних даних або на додавання нових даних до баз даних. Ці запити мають бути сформульовані мовою запитів або мовою маніпулювання даними.

Архітектура ANSI/SPARC не підтримується в повному обсязі жодною з сучасних

СКБД, особливо, якщо це стосується концептуального рівня. У більшості наявних систем концептуальна схема насправді є простим об'єднанням усіх окремих зовнішніх схем, доповненим засобами гарантування безпеки даних і правилами забезпечення цілісності. Зовнішні схеми підтримуються за допомогою так званих віртуальних таблиць. Щодо внутрішнього рівня, то він, як правило, підтримується різноманітними механізмами опису структур зберігання даних і методів доступу до них.

У сучасних СКБД зазвичай підтримується єдина інтегрована мова, що містить усі необхідні засоби для роботи з БД, починаючи від її створення, і забезпечує базовий, призначений для користувача, інтерфейс баз даних. Стандартною мовою найпоширеніших на сьогодні реляційних СКБД є мова SQL (Structured Query Language – мова структурованих запитів).

Історія розвитку баз даних

Історію розвитку БД можна поділити на чотири періоди.

Період становлення – 1960-ті роки

Ідея застосування файлів для зберігання даних спільного використання виникла наприкінці 1950-х років. Проте саме у 60-ті з'явився сам термін „база даних” і було створено кілька систем баз даних. Цього ж десятиліття з'явилася класифікація БД за структурами даних,

що в них використовується, – почали вирізняти системи БД з ієрархічною й мережною структурами.

Ієрархічні системи, в яких базова структура даних мала деревоподібний вигляд, досягли найвищої ефективності функціонування, але виразові можливості цих систем лишилися відносно низькими. Системам зі структурами даних типу мереж, навпаки, вдалося надати значно кращих виразових можливостей, але вони програють в ефективності функціонування, тобто для плідної експлуатації таких систем від користувача вимагався (і вимагається) значно вищий рівень кваліфікації. Останніми з'явилися СКБД реляційного типу, які характеризуються найпростішою структурою даних (таблиця, або так званий плоский файл) з одного боку та високим рівнем мов маніпулювання даними – з іншого. Це зробило їхні виразові можливості максимально потужними, але знизило ефективність функціонування. Використання в мейфреймах магнітних дисків замість магнітних стрічок сприяло створенню в середині 60-х років перших систем керування БД, з яких найрозвиненішою виявилася система IMS фірми IBM, що підтримувала ієрархічну структуру даних. Головний ідеолог мережевого підходу Ч. Бахман 1963 року розробив першу промислову систему БД IDS, орієнтованого на мережну організацію даних.

Асоціація CODASYL, що створила мову програмування COBOL, у 1967 році організувала робочу групу з питань БД, яка узагальнила мовні специфікації систем БД. (Відповідні звіти були опубліковані в 1969 та 1971 роках, які за найменуванням робочої групи – Data Base Task Group – отримала назви DBTG 69 і DBTG 71). Взавши за основу реалізовану в системі IDS мережну структуру даних та методи навігації нею, група DBTG істотно розвинула й обґрунтувала мережну модель.

Одним з типових представників систем, що відповідали пропозиції CODASYL DBTG, була система Integrated Database Management System компанії Cullinet Software, призначена для використання на мейфреймах IBM.

У цей же період чітко окреслилися два підходи до проблеми замкненості систем БД.

Створення систем замкнутого типу, що містить у своєму складі не традиційні мови програмування, а не процедурні мови запитів. Головна мета розробників полягала в тому, аби з цими системами міг працювати пересічний оператор, а не лише фахівець з програмування. До таких систем належали TDMS і UL/1.

Створення систем з базовою мовою. Такі системи, окрім власне мов маніпулювання даними, надають мовні й інструментальні засоби розробки прикладних програм з використанням наявних мов програмування. Цього принципу дотримувалася група DBTG.

Наприкінці періоду становлення виник термін інформаційно-керуюча система. У той час під цим терміном розуміли орієнтовану на пошук даних систему БД, що забезпечувала можливість робити з віддаленого терміналу.

Період розвитку – 1970-ті роки

Концепція БД отримала широке розповсюдження завдяки поліпшенню характеристик апаратного забезпечення комп'ютерів. Успішно впроваджувалися системи, орієнтовані на підтримку ієрархічної та мережевої структур даних.

Тривала далі робота CODASYL DBTG. Була специфікована система мов для БД CODASYL, яка складається з кількох груп мовних специфікацій. У 1975 році з'явився звіт робочої групи ANSI/X3/SPARC, у якому розглядалося питання про стандартизацію БД і СКБД, а також про те, що саме може підлягати стандартизації. Група вирішила, що ця проблема стосується лише інтерфейсів, які можуть існувати між різними компонентами СКБД, самі ж програмні компоненти стандартизації не підлягають. Учені скерували свої подальші зусилля на виявлення таких інтерфейсів і, врешті-решт, запропонували концепцію трирівневої архітектури БД, яка стала класичною і дотепер не втратила актуальності.

Період розвитку більш відомий завдяки створенню реляційної моделі даних, як 1970 року запропонував співробітник інституту фірми IBM у Сан-Хосе Е.Ф.Кодд.

Протягом десятиліття всебічно досліджувалися теоретичні й прикладні питання цієї моделі, розроблялися експериментальні реляційні СКБД. Плідна праця дала можливості

створити формальну теорію БД, яка до цього мала описовий характер. Протягом кількох років багато провідних фірм проводили експериментальні дослідження, спрямовані на винайдення прототипів реляційних СКБД, підвищення їхньої ефективності та функціональності. І нарешті наприкінці десятиліття були створені перші промислові реляційні СКБД.

Період зрілості – 1980-ті роки

Реляційна модель отримала повне теоретичне обслуговування. Було розроблено великі реляційні СКБД – Oracle, Informix та інші. Промислові реляційні системи почали використовуватися в усіх сферах людської діяльності. У такий спосіб реляційні системи практично витіснили зі світового ринку попередні СКБД ієрархічних та мережових типів.

У цей період проводилися також теоретичні й експериментальні дослідження в галузі баз знань та були створені численні експертні системи. У більшості випадків БЗ розробляються на основі реляційних СКБД.

Подальший розвиток реляційних СКБД відбувався за такими напрямками.

Зручність застосування. З поширенням персональних комп'ютерів постало принципове питання щодо зручності використання програм, яке також стосувалося СКБД. У зв'язку з цим у СКБД почали інтенсивно застосовуватися засоби інтерфейсу користувача.

Багатоплановість. Зростання попиту на БД у нетрадиційних галузях їхнього застосування (системи автоматизації проектування, видавнича справа тощо) привело до виникнення потреби у зберіганні й обробці в БД зображень, звуків та повнотекстової, а не лише символічної інформації.

Постреляційний період – з початку 1990-х років

Розпочалися інтенсивні дослідження з питань дедуктивних та об'єктно-орієнтованих БД, які сприяли створенню дослідницьких прототипів таких систем.

Так, 1991 року з'явилася ODMG (Object Data Management Group) – група з питань керування об'єктними БД, яка посіла особливе місце в галузі стандартизації об'єктно-орієнтованих СКБД. В 1993 році група видала свій перший стандарт ODMG-93, а в 1995 був опублікований його вдосконалений варіант. З розвитком інтернет – технологій розробники спрямували свої зусилля на впровадження БД в Інтернет. Широке приєднання СКБД разом з БД до „всесвітньої павутини”, окреслилися різні підходи – починаючи від найпростіших „публікацій” БД в Інтернеті й завершуючи розробкою веб-серверів, які надають користувачам Інтернету весь спектр послуг, що стосуються роботи з БД на сервері. Інтенсивного розвитку набули дослідження й розробки, присвячені маніпулювання структурами даних в Інтернеті.

Слід зазначити, що незважаючи на розвиток більш передових технологій БД, реляційні БД і досі залишилися домінуючими як у настільних системах, так і на промисловому рівні.

Моделі даних. Поняття про моделювання даних. Дані та їхня семантика.

Слово „дані” походять від латинського „datum” – факт, проте дані не завжди відповідають конкретним чи навіть реальним фактам. Іноді вони неточні або описують те, чого насправді не існує. Даними ми вважатимемо опис будь-якого явища (чи ідеї), що викликає зацікавленість через певні потреби. З даними нерозривно пов'язані їхня інтерпретація (або семантика), тобто той зміст який їм приписується. Дані описуються тією чи іншою мовою і фіксуються на певному носії (скажімо папері). Зазвичай дані (факти) та їхня семантична інтерпретація фіксуються спільно. Проте в деяких випадках дані й інтерпретація розділяються. Так, у зведений статистичній таблиці зверху, в її папці, міститься інформація стосовно того, чим є дані (заголовок таблиці, описана інформація, назви стовпців тощо), а нижче розташовуються власне рядки чисел.

Застосування комп'ютерів для введення й обробки даних сприяло ще більшому розділенню даних та їхньої інтерпретації. Оскільки комп'ютери оперують даними як такими, велика частина інтерпретуючої інформації взагалі явно не фіксується. Програма одержує певні вхідні дані, обробляє їх і видає результат у вигляді сукупності вихідних даних.

З розвитком обчислювальної техніки з'явилася можливість фіксувати за допомогою комп'ютерів семантику даних, тобто закладати інтерпретацію в програми, що оперують даними. Проте за умов спільного використання даних різноманітними прикладними програмами цей підхід можна застосовувати лише частково. Інакше процес написання програми, які містять схожі, хоча й не ідентичні механізми інтерпретації, стає не ефективним. У подібній ситуації доцільно зв'язувати дані з механізмами інтерпретації, що має забезпечувати уніфікованість подання семантичної інформації. У результаті змінюється роль даних, їх вже не можна розглядати лише як сукупність бітів, вони отримують певне семантичне навантаження.

Засоби інтерпретації мають бути гнучкими, аби разом з незмінними параметрами даних відображувати й аспекти їхньої еволюції. Цього досягають двома способами.

По-перше, можна відтворювати різні погляди на одні й ті самі дані. Наприклад, розглядати певну особу з точки зору кадрової системи як службовця, з погляду виробничих процесів – як виконавця робіт, а з боку медичного обслуговування – як пацієнта. По-друге, різні дані можна подавати одноманітно. Так, адміністратори, клерки, агенти, секретарі незалежно від роду своєї діяльності можуть розглядатися в кадровій системі як службовці.

Моделювання даних

Існує багато типів моделей – фізичні, математичні, економічні, тощо, які відображають різні аспекти реального світу.

РОЗДІЛ 1

ПРИНЦИПИ СТРУКТУРНОЇ ОРГАНІЗАЦІЇ ДАНИХ ТА БАЗ ДАНИХ

1.1. Загальні принципи та концепція структури організації даних

Викладена концепція та теоретичні основи вирішення задач структуризації потоків даних в інформаційних комп'ютеризованих системах.

Викладена концепція теорії структуризації потоків даних дозволяє чітко визначити її задачі в інформаційних системах, комп'ютерних та комп'ютеризованих мережах. Проведено аналіз атрибутів та компонентів теорії структуризації у розрізі наукових галузей. Оскільки сьогодні практично всі напрямки розвитку сучасної науки тісно пов'язані з інформаційними технологіями, то узагальнення теорії та методології структуризації інформаційних потоків є важливою науковою проблемою.

Викладено семантико-евристичне поняття структуризації в широкому аспекті, який охоплює теорію систем та взаємодію їх компонентів, теорію інформації та архітектури комп'ютеризованих систем.

Поняття структуризації пов'язується з процесами розвитку та вдосконалення інформаційних систем: «Напрямок розвитку відбувається не просто до складного, але й до реального Світу, тобто структуризованому під його реалії.» «Оскільки все, що існує у Світі структуризується під його реалії».

Аналіз відомих рішень.

Фундаментальним прикладом структуризації інформаційних конгломератів є структура та інформаційна організація ДНК, яка демонструє роль відбору та виживання в системі структурно стійкіших та інформаційно досконаліших форм живої матерії. При цьому фундаментальною теоретико-логістичною основою виступає ентропійний принцип оптимальності. Тому усвідомлення концепції структуризації процесів, інформаційних даних та узагальнення фундаментальних теоретичних основ є актуальною проблемою.

Швидкий розвиток сучасних інформаційних технологій, комп'ютерних мереж та комп'ютеризованих систем викреслює дослідження проблеми структуризації в даній галузі як найперспективнішою задачею, оскільки практично всі напрямки розвитку сучасної цивілізації тісно пов'язані з інформатизацією суспільства та відповідною структуризацією інформаційних потоків.

Щоб краще зрозуміти роль інформації та ефективно використовувати її у процесі функціонування систем, необхідно знати її структуризацію. Найвищим рівнем структуризації інформації є виділення її як системи для конкретного об'єкта і відповідних підсистем. Проте будь-яку систему можна розчленовувати аж до виявлення неподільних одиниць.

Функції та задачі структуризації даних.

Часто користуються поняттями логічних і фізичних структур. Поняття логічних структур особливо цінне при проектуванні та використанні складних автоматизованих систем, банків і баз даних.

Логічний аспект структури даних визначає загальні зв'язки між окремими елементами даних безвідносно до способу їх реєстрації та зберігання. До логічних структур (залежно від обраної структуризації) можна віднести поле, запис, файл; атрибут, групу, групове відношення, статтю, файл, базу даних і т. ін. У даному разі поняття «поле» і «атрибут» — найменші структурні одиниці, що не піддаються подальшому змістовному розчленуванню.

Фізичний аспект структури даних пов'язується зі способом реєстрації даних на носіїв чи способом запам'ятовування їх у пам'яті процесора. В умовах використання сучасних процесорів фізичними структурними одиницями є біт, байт, кілобайт, мегабайт, гігабайт та ін; у паперових документах — позиція, рядок, графа тощо.

У випадках, коли можна візуально сприймати інформацію на носіях, користуються таким поняттям мінімальної її структурної одиниці як реквізит (атрибут). Реквізити можуть характеризувати якісну чи кількісну сторону інформаційної сукупності.

Повний набір показників, що характеризують певний об'єкт, називають позицією (наприклад, рядок класифікатора), а ті, що описують господарську операцію, — документом. Ряд однотипних документів із різним значенням створюють інформаційну сукупність — так званий масив (машинний аналог — файл).

За видом взаємозв'язку між окремими елементами структури даних бувають лінійні (однорівневі) та нелінійні (ієрархічні чи багаторівневі).

Лінійні структури можуть бути послідовні (з упорядкованими чи неупорядкованими елементами) та рядкові (з фізично розкиданими і логічно пов'язаними з допомогою ланок зв'язку елементами).

Нелінійні структури включають складні списки, дерева, мережі, таблиці і їх комбінації (гібридні).

Структури зі складними списками складаються з інших списків; деревоподібні передбачають ієрархічний зв'язок елементів; мережні мають довільний зв'язок між елементами, табличні структури мають вигляд 2Д, 3Д або МД-таблиць.

Можливість структуризації інформації і переходу від одних структур до інших (частіше від традиційних безмашинних до машинних і навпаки) забезпечує основу для ефективного аналізу, оцінювання та опрацювання інформації, її перегрупування. При цьому вдається уникнути дублювання й надмірності.

Викладення концепції вирішення проблеми структуризації інформаційних процесів та даних дозволяє підійти до узагальнення фундаментальної теорії структуризації та піти шляхом систематизації наукових дисциплін, які є атрибутами та компонентами теорії та методології структуризації. На рис.1.1 приведено приклад такої систематизації.

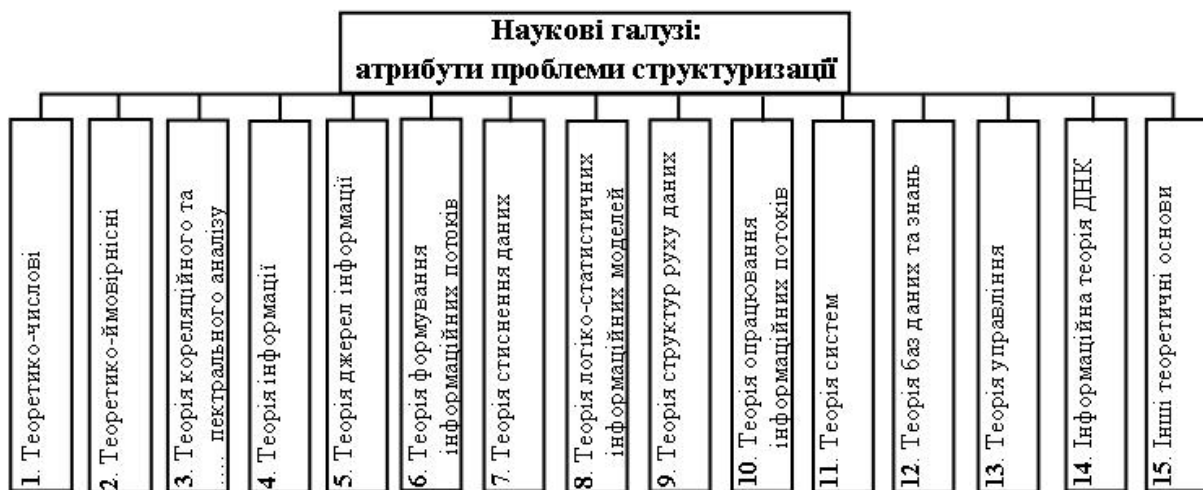


Рис.1.1. Класифікація атрибутів теорії структуризації.

На рис.1.2 показано узагальнені процеси структуризації інформаційних потоків джерел інформації, де ДІ – джерело інформації, S - сенсорна система, # - процеси дискретизації у Хеммінговому просторі, FS - функціонал структуризації, $F(U, \cap)$ - операції структурного об'єднання та розмежування.

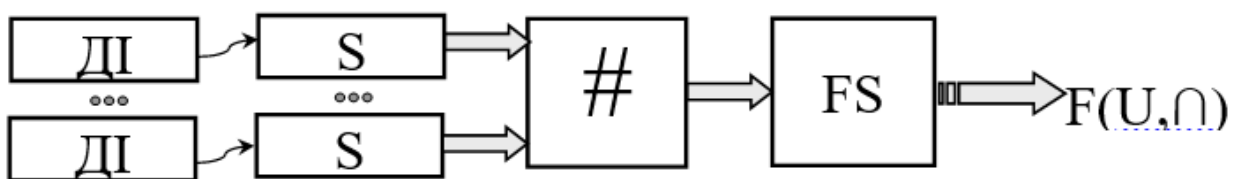


Рис.1.2. Процеси структуризації.

Формування та опрацювання структуризованих даних.

Успішне вирішення проблеми формування та опрацювання структуризованих даних (СД) у розподілених комп'ютерних системах концептуально пов'язано з доцільним застосуванням широкого класу фундаментальних теоретичних засад у галузі інформатики, кібернетики та системотехніки. При цьому необхідно сформулювати основні атрибути положення та інформаційні процедури узагальненої теорії структуризації даних.

Теоретичні та прикладні резерви такого узагальнення дозволяють суттєво вдосконалити математику формування, передавання та цифрового опрацювання даних та досягнути відповідного значного підвищення ефективності функціонування компонентів, алгоритмів та результативності використання опрацьованих даних у розподілених комп'ютерних системах..

Дослідження світового досвіду реалізації засобів формування структуризованих даних, методів аналого-цифрового перетворення потоків технологічних даних, методів оптимізації кодування техніко-економічних даних (ТЕД) та методів організації руху даних в розподілених комп'ютерних системах (РКС) показують, що подальше вдосконалення та оптимізація характеристик апаратно-програмних засобів РКС успішно реалізується завдяки:

- високому рівню паралелізму реєстрації технологічних даних в багатоканальних АЦП, які на рівні аналого-цифрових кодерів реалізують функції зниження надлишковості кодування даних та їх захисту від помилок;
- використанню математичних основ новітніх теоретико-числових базисів (ТЧБ): Крестенсона, Галуа, Уолша та ін. для представлення даних у вигляді фреймів СД, максимально адаптованих до низових рівнів РКС та роботи в умовах інтенсивних промислових заводів;
- вдосконаленню методів кодового представлення ТЕД;
- розширенню сукупності та вдосконаленню методів побудови моделей руху даних, а також ідентифікованих моделей собівартості руху даних.

При цьому на основі комплексу даних рішень на рівні розподілених комп'ютерних систем (РКС) та мереж успішно реалізуються наступні задачі:

- формування СД та організації руху потоків даних у РКС;
- визначення перспективних методів структуризації різних типів даних у вигляді фреймів;
- розробка критеріїв оптимізації процесів формування алфавітно-цифрових, технологіко-економічних та структуризованих даних;
- дослідження засобів формування даних на основі критеріїв оптимізації;
- розробка теоретичних основ синтезованого формування алфавітно-цифрових та структуризованих даних;
- узагальнення теорії побудови багаторівневих інтерактивних моделей руху даних розподілених комп'ютеризованих систем реального часу на основі подальшого вдосконалення принципів формалізації атрибутів граф-алгоритмічних моделей руху даних та ідентифікованих епюр руху даних;
- вдосконалення алгоритму реалізації стратегії проектування РКС на основі законів доцільності та ідентифікованих циклів руху даних;
- розробка атрибутів та програмних засобів побудови багаторівневих граф-алгоритмічних моделей РКС;
- розробка мобільних апаратно-програмних засобів синтезованого вводу СД для низових рівнів РКС;
- вдосконалення структур та алгоритмів опрацювання даних компонентами КС в різних ТЧБ.

Впорядкування, структуризація різних видів даних, побудова моделей складних комп'ютеризованих систем є актуальною задачею, яка направлена на вдосконалення теорії, методології, практики їх проектування та діагностики штатності функціонування в реальному масштабі часу. В методологічному плані вирішення такої задачі потребує врахування проблемної орієнтованості, цілісності та складності, невизначеності, адаптивності, а також універсальності комп'ютеризованої системи. При цьому відповідно визначається ступінь

цілеспрямованості та мета функціонування системи, можливості опису системи однією моделлю, оцінка ентропії, що відображає необхідну кількість керуючої інформації, можливості пристосування системи до впливу зовнішніх факторів, а також опис системи математичними моделями, що мають однакову структуру незалежно від класу об'єктів – джерел інформації.

Аналіз методів структуризації даних, обґрунтування перспективного напрямку розвитку теорії, методології та техніки їх структуризації в сучасних комп'ютеризованих системах є актуальною задачею сьогодні.

Структуризація даних при виконанні функції формування, передавання, цифрового опрацювання, зберігання та використання структуризованих фреймів для управління об'єктами зображена на рис.1.3.

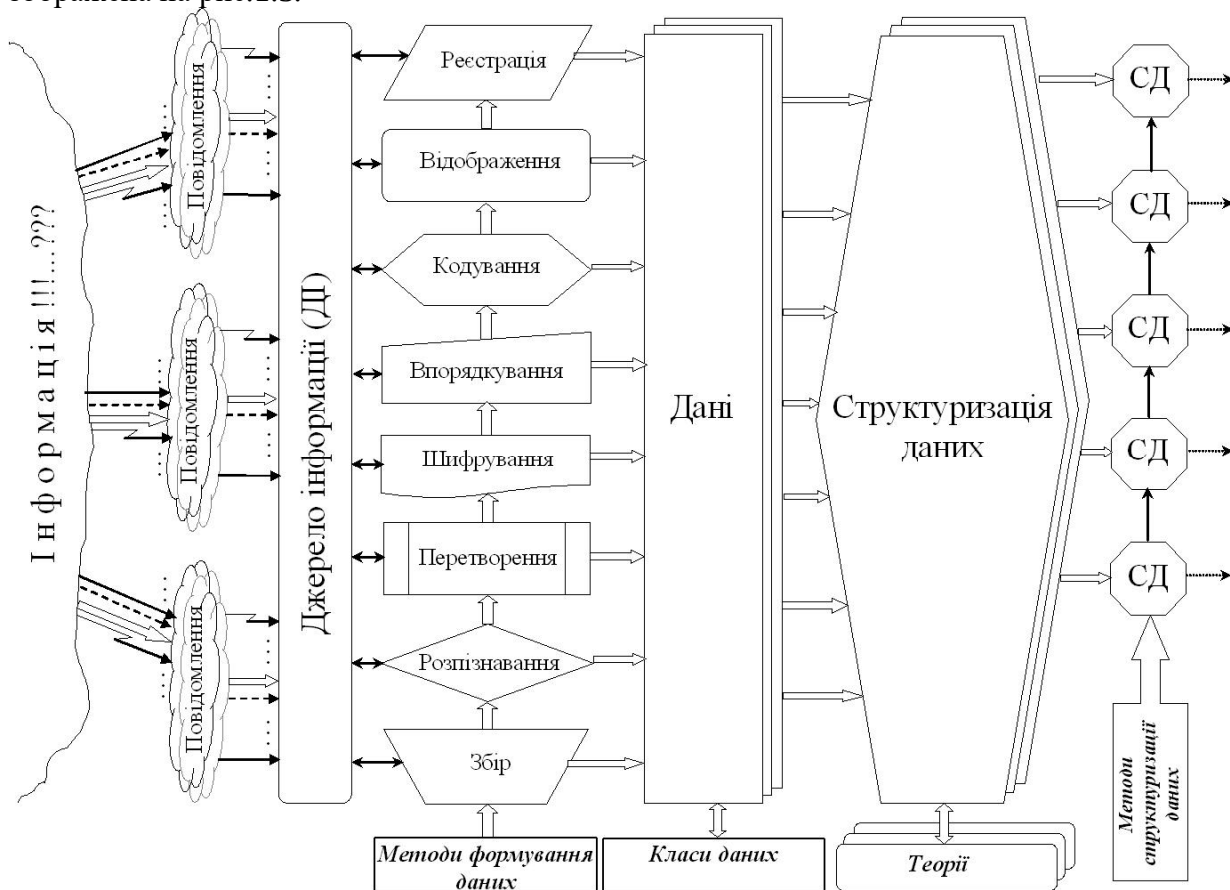


Рис.1.3. Узагальнена схема функцій концепції структуризації даних.

Приведена на рис.1.3 схема продукційної взаємодії функцій структуризації даних показує шлях формування та математико-алгоритмічного опрацювання даних згідно їх проблемної орієнтації. Наприклад, розпізнавання образів, стиснення даних, захист від несанкціонованого доступу, результати розв'язаних задач та ін.

Прикладом успішного застосування основ теорії структуризації даних, яка знаходиться у стадії становлення та розвитку, є формалізація структур та організація опрацювання даних у багаторівневих РКС на основі сімейства матричних моделей руху даних та епюр (згідно функціоналів).

Метод формування та організації руху структуризованих даних багаторівневих РКС полягає у послідовному виконанні функцій наступних алгоритмів:

$$\begin{aligned}
& F_1 \left[\begin{array}{c} \begin{array}{ccc} \odot & \bigcirc & \otimes \end{array} \rightarrow \begin{array}{ccc} \odot & \bigcirc & \otimes \end{array} \rightarrow \begin{array}{c} \otimes \end{array} \\ \vdots & \quad \quad \quad \vdots \\ \begin{array}{ccc} \odot & \bigcirc & \otimes \end{array} \rightarrow \begin{array}{ccc} \odot & \bigcirc & \otimes \end{array} \rightarrow \begin{array}{c} \otimes \end{array} \end{array} \right] \Rightarrow \\
& \Rightarrow F_2 \left\{ \begin{array}{l} E_p = F(T, V, M, S) \\ E_d = F(T, M, V_R, V_W, S) \\ E_{IO} = F(T, M, I, S) \\ E_{SLA} = F(V_R, V_W, P_i, S) \\ E_I = F(T, V_R, V_W, S, M) \\ X_{IO} = F(S_i : X(t) + \dot{O} \ddot{A} \ddot{A}, M_x, D_x, \sigma_x, R_{xx}, R_{xy}, M_{ij}, S(w), K_{ij}, \ddot{E} \ddot{N} \ddot{I}, I_x, \ddot{N} \ddot{A}) \end{array} \right\} \Rightarrow \\
& \Rightarrow F_3 [K_{ed}, D(P - V)] \Rightarrow \\
& \Rightarrow F_4 [БРМ_1, ДД, БРДММРД, \dots, D(M_n + ГАМ)] \Rightarrow \\
& \Rightarrow F_5 \left\{ ДЦРД, ДЕРД, \Delta EРД(t_j), \int \Delta EРД(T) dT, \sum_{i=1}^n \int \Delta EРД(T) dT \right\} \Rightarrow F_6 [G_{KC}] \Rightarrow \\
& F_7 \left[\begin{array}{c} \begin{array}{|c|c|c|c|c|c|c|c|} \hline star & T & N & S & \{X\} & \{M\} & \{L\} & sto \\ \hline \end{array} \\ \begin{array}{|c|c|c|c|c|c|c|c|} \hline \Phi & C_i & C_j & N & S & M & L & ТЕД & \Phi \\ \hline \end{array} \\ \begin{array}{|c|c|c|c|c|c|c|c|} \hline T & N & S & X & M & L & ТЕД & I \\ \hline \end{array} \end{array} \right] \Rightarrow F_8 \left[\begin{array}{c} \begin{array}{|c|c|c|c|c|} \hline \Phi & C & C & Y & \Phi \\ \hline \end{array} \\ \begin{array}{|c|c|c|c|c|} \hline \Phi & C & C & Y & G & \Phi \\ \hline \end{array} \\ \begin{array}{|c|c|c|c|c|} \hline \Phi & C & C & V & Y & \Phi \\ \hline \end{array} \\ \begin{array}{|c|c|c|c|c|} \hline \Phi & C & C & W & Y & \Phi \\ \hline \end{array} \end{array} \right],
\end{aligned}$$

де новизна запропонованих наукових рішень по вдосконаленню методів формування та організації руху потоків даних в багаторівневих РКС полягає у розширенні та вдосконаленні функцій організації руху даних на основі:

- багаторівневої матричної моделі руху даних (1);
- розширення функцій формування СД шляхом сумісного кодування технологічних даних (x(t)) та ТЕД (2) при заданих квазістаціонарних станах ОУ ($S_i :$), а також реалізація апаратно-програмних засобів синтезованого вводу алфавітно-цифрових даних;
- формування мінімально надлишкових та захищених від помилок фреймів структуризованих даних (СД), які формуються, передаються, обробляються, зберігаються та передаються;

• диференціація собівартісних характеристик руху даних в активних вузлах МРД

$$D(P - V) = \sum_{i=1}^m P_i - \sum_{j=1}^n V_j, \text{ де } m - \text{число диференційованих компонентів прибутків, } n - \text{число}$$

диференційованих компонентів витрат в активному вузлі МРД;

- розширення класу МРД до рівня багаторівневих МРД (БРМРД);
- врахування диференціації форм документів (ДД) та даних ММ по відповідних активних вузлах МРД, що відповідають відповідним об'єктам МРД;
- розширення сукупності похідних моделей руху даних шляхом включення в їх склад важливого класу граф-алгоритмічних моделей (ГАМ);

- диференціації циклів руху даних з врахуванням диференціації собівартісних характеристик руху даних в активних вузлах БРДММРД (ДЦРД) та відповідна диференціація епюр руху даних (ДЕРД).

Теоретичне обґрунтування реалізації функціоналів, їх моделювання та аналіз, а також оцінка ефективності запропонованих методів формування та організації руху ідентифіковано-структуризованих даних в РКС є актуальним предметом дослідження сучасних КС.

Створення фундаментальних основ теорії структуризації даних стає потужним інструментом підвищення ефективності та розширення функцій сучасних інформаційних систем та успішного вирішення завдань переходу до інформаційного суспільства.

1.2. Системні характеристики даних та класифікація моделей даних

З давніх часів людство витрачає великі зусилля на вивчення явищ, що відбувається в природі. Первинними в процесі пізнання завжди є результати спостережень. Вони є відправним пунктом до моделі, до абстрактного мислення, а вже від моделі здійснюється перехід до наукової і практичної діяльності. Ця схема пізнання реального світу застосовується для природних об'єктів і явищ та для штучних об'єктів і явищ, показано на рис.1.4.



Рис.1.4. Зв'язок реального світу природних та штучних об'єктів і явищ.

Реальний світ - це об'єкти і явища природи і всесвіту, який сприймає людина у вигляді інформації.

Інформація (Information)- роз'яснення, виклад, набір відомостей - це відомості стосовно осіб, предметів, об'єктів, фактів, подій, явищ і процесів незалежно від форми їх подання. Інформація як відомості про об'єкти або явища відображаються у вигляді конкретних даних, показано на рис.1.5.

Дані (data) - це опрацьована інформація, зафіксована (закодована) у певній формі, придатній для подальшого опрацювання, зберігання та використання, показано на рис.1.5.

Модель даних - це сукупність структур даних і операцій їх обробки.



Рис.1.5. Види інформації.

Сучасна систематизація даних охоплює три їх основні класи:

- фізичні дані;
- логічні дані;
- віртуальні дані.

Фізичні дані – це дані, які представлені в адресному просторі фізичних носіїв (жорстких магнітних носіїв, гнучких магнітних дисків, оптично-лазерних дисків.) а також на твердих носіях (документи, таблиці, графіки).

Логічні дані – використовують в теорії та практиці опису логічних моделей баз даних, а також в програмних продуктах як форматні параметри.

Віртуальні дані – це такий тип даних, які відсутні в фізичному вигляді комп'ютерних систем і можуть формуватися в процесі рішення задач і розрахунків, а також бути в динамічному стані при передаванні в комп'ютерному моделюванні.

Системні характеристики даних представляють: часом зберігання, швидкістю запису, швидкістю зчитування, об'ємом та захистом даних від помилок

$$Ed = F(T, M, Vr, Vw, S); \quad (1.1)$$

де T-час зберігання, M-об'єм, Vr-швидкість запису, Vw-швидкість зчитування, S-захист від помилок та несанкціонованого доступу.

Рівні подання даних. Розрізняють три архітектурні рівні подання даних: а) зовнішнє подання даних; б) концептуальне подання даних; в) внутрішнє (фізичне) подання даних. а) Через зовнішній рівень користувачі та застосування отримують доступ до бази даних. Мета зовнішнього рівня – надати користувачу /застосуванню лише ті дані, які йому потрібні (а отже, до яких дозволений доступ) і в потрібному вигляді. Це індивідуальний рівень користувача, який може бути кінцевий користувач, програміст чи застосування. Кожен з них має свою мову спілкування: для кінцевого користувача- це спеціальна мова запитів, для програміста - одна з мов програмування, розширена командами звернення до СКБД, для застосування- це, як правило, певний стандартний інтерфейс звернення до бази даних через СКБД. Зовнішня модель - це засоби зображення концептуальної моделі предметної області з урахуванням інтересів конкретних користувачів або застосувань. Кожна зовнішня модель подається в СКБД у вигляді зовнішньої схеми. б) На концептуальному рівні здійснюється інтегрований опис предметної області, для якої розробляється БД, незалежно від її сприйняття окремими користувачами та способів реалізації в комп'ютерній системі. Саме через концептуальну схему здійснюється централізоване адміністрування баз даних. Внутрішня модель є відображенням концептуальної моделі предметної області з урахуванням способів зберігання даних і методів доступу до них. Внутрішня модель відображається в СКБД у вигляді внутрішньої схеми. Внутрішня модель- це не модель фізичної пам'яті з характеристиками конкретних пристроїв зберігання даних (циліндри, доріжки, тощо); вона описується у вигляді нескінченної абстрактної лінії пам'яті, яка може структуруватися за допомогою інших абстрактних понять на зразок блоків, кластерів, індексів, тощо. Доступ до фізичної пам'яті надається за допомогою опису відображень внутрішньої моделі на фізичну пам'ять операційної системи. Основними видами моделей даних є: а) модель предметної галузі; б) модель даних; в) модель бази даних. Класифікація моделей даних показано на рис.1.6.

Класифікація моделей даних

Ядром будь-якої бази даних є модель даних. *Модель даних* являє собою безліч структур даних, обмежень цілісності та операцій маніпулювання даними. За допомогою моделі даних можуть бути представлені об'єкти предметної області, взаємозв'язку між ними.

Математична формалізація моделей джерел інформації

В загальному формалізована математична модель джерел інформації визначається як деяке зображення Θ елементів множин фізичних сигналів (повідомлень) X на множину перетворених сигналів (оброблених повідомлень) Y , при чому простір Y повинен бути повним і метричним.

Прообразом кожного з вихідних елементів y_i множини Y може бути як один елемент x_j множини X , або деякої підмножини елементів $X_i = \{x_{ik}\}$, яка в граничному випадку може

бути рівною X . Де X_i – підмножина вхідних інформаційних повідомлень множини X , які визначають вихідні повідомлення y_i з множиною Y . Де $y_i = Op \{x_{i1}, x_{i2}, \dots, x_{im}\}$;

В деяких випадках вихідне повідомлення y_i визначається не тільки вхідними станами ДІ з множинами X , але й внутрішнім станом системи в даний момент або й декількома попередніми станами. Тоді формальний запис перетворення Θ можна подати у вигляді: $y_i = Op \{z_{l-n}, \dots, z_{l-1}, x_{i1}, x_{i2}, \dots, x_{im}\}$; де $\{z_l\}$ – деяка послідовність внутрішніх станів системи, від яких залежить образ y_i ; l – номер поточних станів, n – максимально віддалений від поточного стану системи, який впливає на результати перетворення.

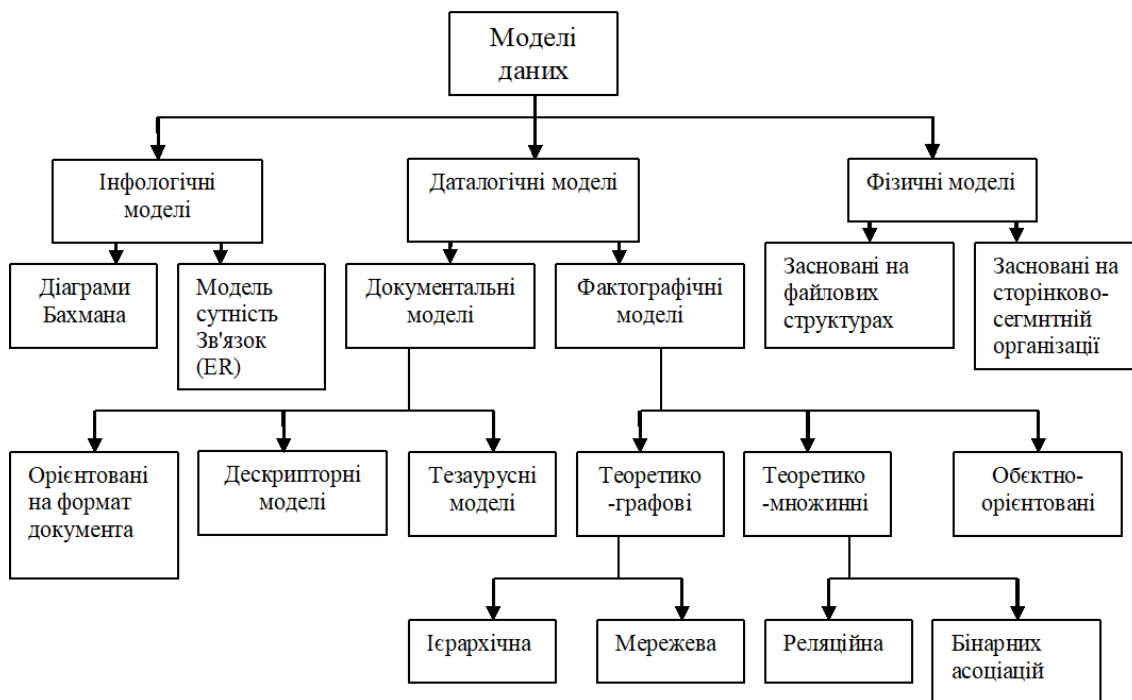


Рис. 1.6. Класифікація моделей даних

Ідентифікація в процесах керування.

Будь-який обчислювальний процес в об'єкті F є відображення (згідно визначеного алгоритму програми A та мети T) деяких вхідних інформацій X (x_i – ТЧБ Фур'є) у вихідні дані Y (y_i – ТЧБ Унітарний, Хаара, Крейга, Радемахера, Грея, Крестенсона, Уолша, Галуа – результат розв'язання задачі), показано на рис.1.7.

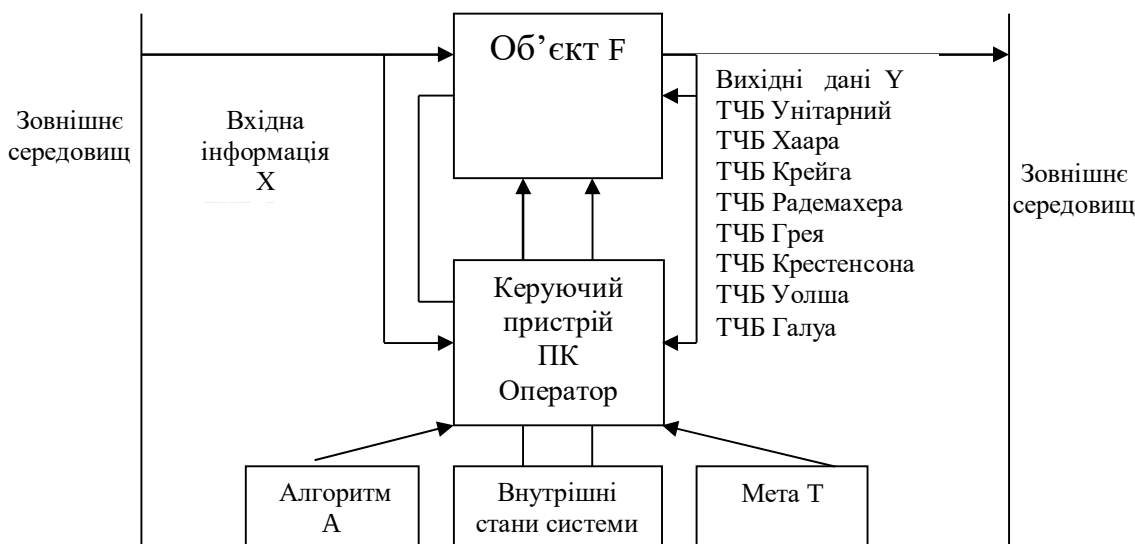


Рис.1.7. Загальна схема керування для опрацювання вихідних даних

Співвідношення складності представляти опрацьовані дані в алгоритми обчислень визначають два класи задач: обчислювальні задачі представлення даних (достатньо прості) і задачі опрацювання даних (не обчислювальні задачі), багатоопераційний процес обчислень (простий алгоритм обробки даних і складне представлення опрацьованих даних).

1.3. Типи баз даних та рівні їх проектування (моделювання)

Бази даних поділяються на логічні бази даних та на фізичні бази даних.

Логічні бази даних (logical data base). База даних в уяві користувачів її структура може відрізнитися від структури фізичної бази даних. На мові DL/1 фірми IBM логічна база даних є деревоподібна сукупність сегментів, побудованою із одної або декількох фізичних баз даних з допомогою з'єднуючих вказівників. При логічному проектуванні дані, виділені на попередньому етапі, набувають форми, прийнятої у СКБД. Схема концептуальної моделі розробляють з використанням тільки тих типів моделей даних, які підтримують обрану СКБД.

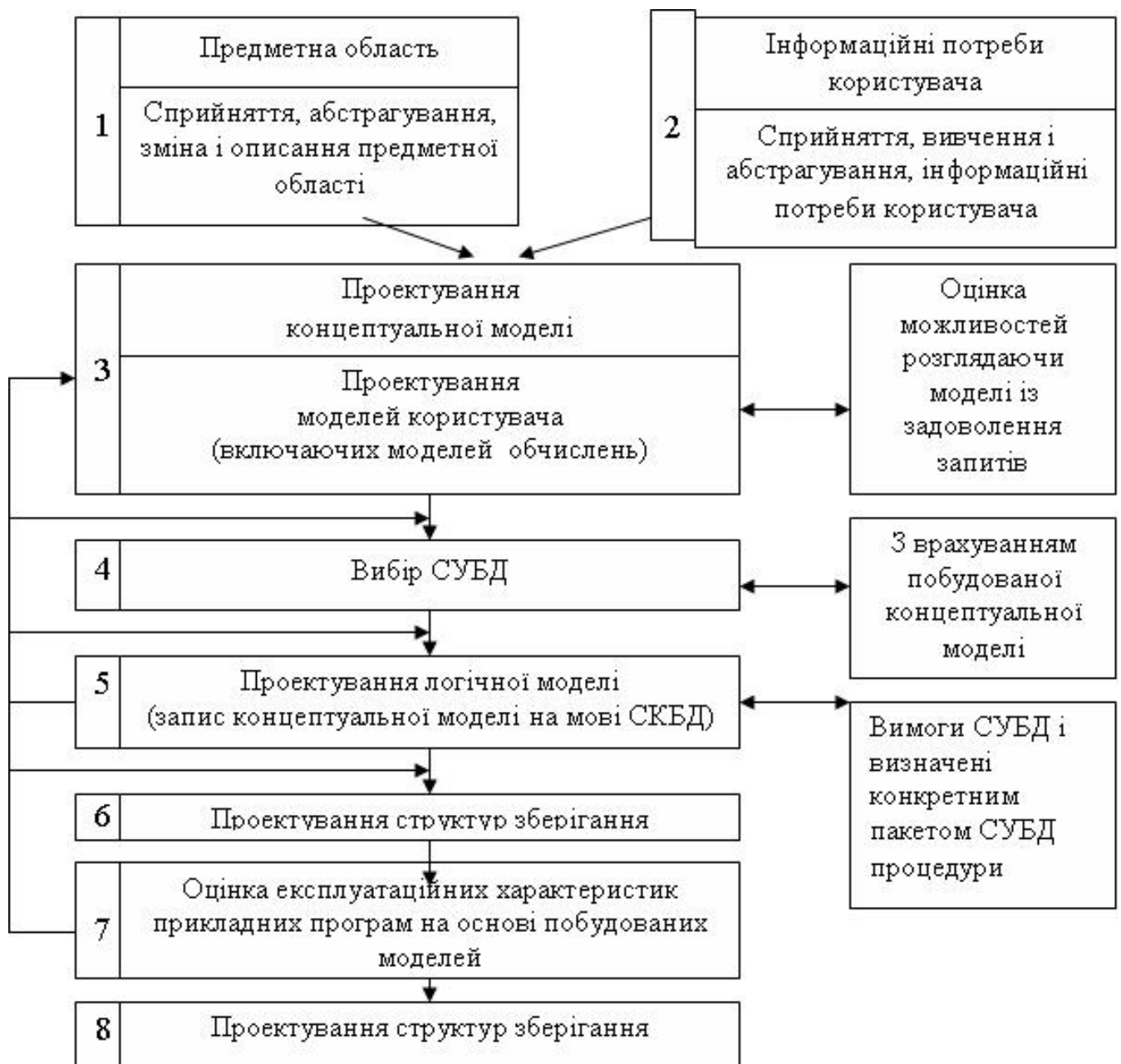


Рис.1.8. Етапи проектування (моделювання) БД

Фізичні бази даних (physical data base). База даних в тому вигляді , як вона зберігається на запам'ятовуючих пристроях , включаючи вказівники та інші засоби підтримки зв'язків між даними. З одної або декількох фізичних баз даних можуть бути отримані різноманітні логічні бази даних. На фізичному етапі проектування вибирають раціональну структуру зберігання даних і методів доступу до них, беручи за основу методи та засоби, які надає СКБД. При описі предметної галузі прийнято подавати у вигляді трирівневої схеми: концептуальне (з точки зору адміністратора), зовнішнє (з точки зору кінцевого користувача і прикладного програміста) та внутрішнє подання (з позицій системного програміста). Проектування даних (баз даних) – це процес відображення досліджуваних явищ реального світу у вигляді даних в пам'яті ПК показані на рис.1.8.

Основний етап проектування БД – інфологічне і датологічне проектування. В основному виділяють логічне і фізичне проектування.

Проектування (моделювання) БД представляє собою багатоетапний процес. Основні етапи цього процесу приведені на рис.1.8.

Проектування (Моделювання) баз даних

Процес створення такої структури БД, яка відповідала б вимогам користувачів, називається проектуванням БД (рис.1.8). Життєвий цикл системи БД є концепцією, в межах якої корисно й зручно розглядати розвиток такої системи.

1.3.1. Класифікація баз даних

Класифікація БД за характером збереженої інформації: –фактографічні (картотеки); – документальні (архіви).

Класифікація БД за технологією фізичного зберігання: – БД у вторинній пам'яті (традиційні);– БД в оперативній пам'яті (in-memory databases); – БД у третинній пам'яті (tertiary databases).

Класифікація БД за змістом:

- географічні;
- наукові;
- історичні;
- мультимедійні.

Класифікація БД за ступенем розподіленості (за способом зберігання даних):

Окремі місця в теорії і практиці займають просторові (spatial).часові, або темпоральні (temporal) і просторово-часові (spatial-temporal) БД.

Класифікація моделей БД за ступенем розподіленості, показана в табл.1.1.

Таблиця.1.1

Класифікація БД за ступенем розподіленості (за способом зберігання даних)

№ п.п	Назва бази даних
1	Централізовані (зосереджені) локальні, просторові (spatial).
2	Розподілені часові, або темпоральні (temporal)
3	просторово-часові (spatial-temporal)

Класифікація БД за структурою організації даних (існуючих). Сам термін database (база даних) появився на початку 1960-х рр., і був введений в користування на симпозиумах, організованих фірмою SDC (System Development Corporation) в 1964 і 1965 рр.

Існує величезна кількість різновидів баз даних, що відрізняються за різними критеріями (наприклад, в «Енциклопедії технологій баз даних» М.Р.Когаловського визначаються понад 50 видів Баз Даних).

Класифікація моделей БД в залежності від способу організації даних, показана в табл.1.2.

Таблиця.1.2.

Класифікація БД за структурою організації даних

№	Назва баз даних
1	Ієрархічні
2	Мережеві
3	Реляційні, (часові темпоральні)
4	Об'єктні
5	Об'єктно-орієнтовані, (з гніздуванням)
6	Об'єктно – реляційні, (постреляційні)
7	Напівструктуровані
8	Розподілені
9	Багатовимірні
10	Куби даних
11	Простори даних
12	Надвеликі бази даних
13	Сховища даних
14	Комп'ютерні бази даних

Класифікація БД за структурою організації даних (розроблених):

- ієрархічно–рекурентна БД в базисі Галуа;
- ієрархічна–реляційна.БД в базисі Галуа;
- лінійно-рекурентна модель БД;
- дворівнева лінійно-рекурентна модель БД;
- багаторівнева лінійно-рекурентна модель БД;
- вертикально-рекурентна модель БД;
- дворівнева вертикально-рекурентна модель БД;
- багаторівнева вертикально-рекурентна модель БД;
- реляційно-рекурентну модель БД,
- об'єктно-реляційно-рекурентну модель сховищ даних глобальних розподілених комп'ютерних систем;
- конгломерати епохальних сховищ даних в базисі Галуа.

1.3.2. Архітектура та характеристики моделей баз даних

Архітектура БД визначає місце розташування БД. Є чотири різновидності архітектур БД: - локальні БД; - архітектура "файл-сервер"; - архітектура "клієнт-сервер"; - багатоланкова архітектура.

Історію розвитку БД можна поділити на чотири періоди: період становлення – 1960 роки: ідея зберігання файлів для зберігання даних спільного використання виникла наприкінці 1950 років. Проте саме у 60-роки з'явився сам термін „бази даних”і було створено кілька систем БД. Цього ж десятиліття з'явилася класифікація БД за структурами даних, що в них використовуються – почали вирізняти системи БД з ієрархічною і мереженою структурами.

Ієрархічні моделі бази даних (hierarchical data base)

Ієрархічна класична структура даних – це структура організації даних має деревоподібний вигляд, де будь-який об'єкт може підпорядковуватися лише одному об'єкту вищого рівня, а йому може підпорядковуватися багато об'єктів нижчого рівня. Такий зв'язок між даними називають „один до багатьох”. За ієрархічним принципом побудовані файлові структури даних на дисках персонального комп'ютера. На рис.1.10 показані зв'язки ієрархічної моделі БД.

Переваги: досягли найвищої ефективності функціонування.

Недоліки: виразові можливості цих систем лишилися відносно низькими.

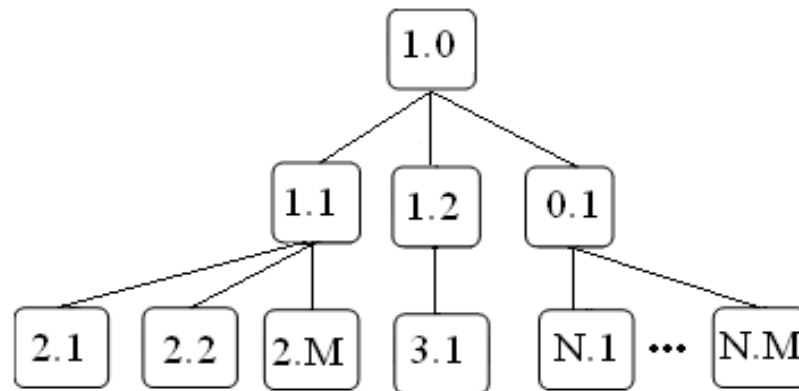


Рис.1.10. Ієрархічна модель БД.

Мережеві бази даних (networks data base)

Мережева класична структура характеризується тим, що будь-який об'єкт одного рівня (одної групи даних) може мати довільні зв'язки з об'єктами іншого рівня. Такі зв'язки називаються “багато до багатьох”. Мережеві структури можна описати у вигляді таблиці, де у першому горизонтальному рядку записують об'єкти одного рівня, а у першому вертикальному – іншого. Така таблиця добре ілюструє зв'язки між об'єктами, але може мати багато порожніх елементів, що призводить до значної надлишковості БД. На рис.1.11 показані зв'язки мережевої моделі БД.

Переваги: значно кращі виразові можливості.

Недоліки: низька ефективність функціонування.

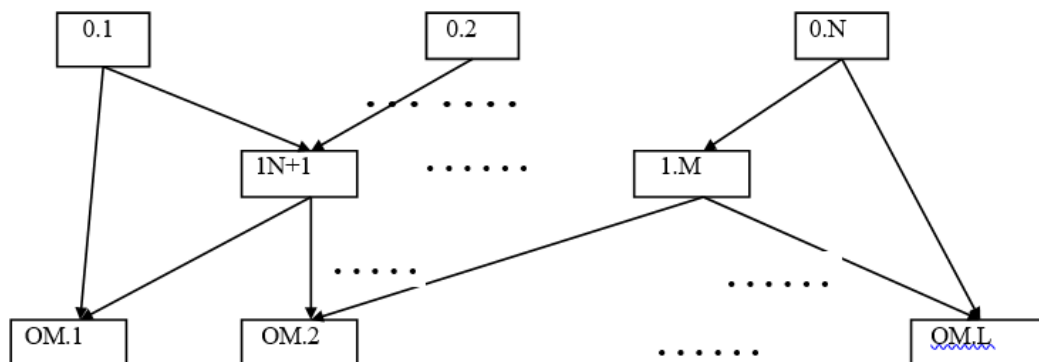


Рис.1.11 Мережева модель БД

Період розвитку – 1970 роки: концепція БД отримала широке розповсюдження завдяки поліпшенню характеристик апаратного забезпечення комп'ютерів. Успішно впроваджувалися системи, орієнтовані на підтримку ієрархічної та мереженої структур даних. Тривала далі робота групи CODASYL DBTG. Була специфікована система мов для БД CODASYL, яка складалася з кількох груп мовних специфікацій.

Період зрілості –1980 роки: реляційна модель отримала повне теоретичне обґрунтування. Були розроблені великі реляційні СКБД – Oracle, Informix та інші. Промислові реляційні системи почали використовуватися в усіх сферах людської діяльності. У такий спосіб реляційні системи практично витіснили зі світового ринку попередні СКБД ієрархічного та мереженого типів.

Реляційні моделі бази даних (relational data base)

Реляційна класична структура є найпоширенішим типом і характеризується поданням даних у вигляді декількох таблиць і зв'язаними між собою кортежів та атрибутів. Один з найпростіших типів зв'язків є „один до одного”. Потрібні дані будуть черпатися з двох таблиць. Отже, для зв'язку між таблицями використовують поле, значення якого не повторюється в різних записах. Це поле називається ключовим. Якщо реляційні таблиці мають спільні поля, то зміни в спільному полі в одній таблиці автоматично відображаються у всіх

таблицях. Мета запровадження реляційних зв'язків – мінімізувати дублювання даних і забезпечити можливість опрацювати (шукати) дані з декількох таблиць, що забезпечує значне зменшення надлишковості БД. На рис.1.12 показана структура реляційної моделі БД.

Переваги: найпростіша структура даних (таблиця або плоский файл), високий рівень мов маніпулювання даними, виразові можливості максимально потужні.

Недоліки: неможливість адекватного відображення семантики предметної галузі, знизило ефективність функціонування.

Постреляційний період –1990 роки: розпочалися інтенсивні дослідження з питань дедуктивних, об'єктно–орієнтованих БД, об'єктно–реляційних БД та впровадження БД в Інтернет.

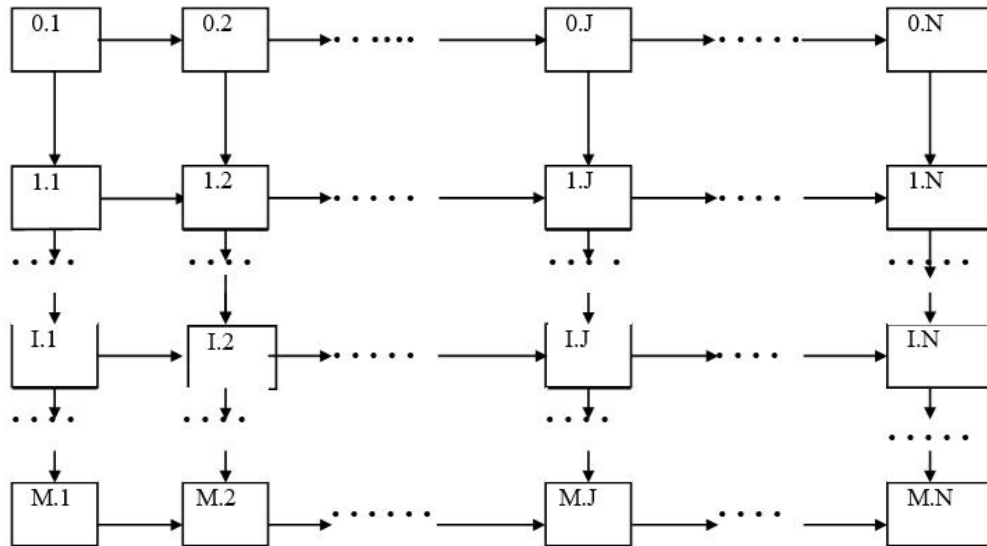


Рис.1.12 Реляційна модель БД

Об'єктні моделі бази даних (Object data base)

Об'єктна база даних — база даних, у якій дані оформлені у вигляді моделей об'єктів, що включають прикладні програми, які управляються зовнішніми подіями. Результатом сполучення можливостей (особливостей) баз даних і можливостей об'єктних мов програмування є Об'єктні системи керування базами даних. ОСКБД дозволяє працювати з об'єктами баз даних також, як з об'єктами в програмуванні на ОЯП. ОСКБД розширює мови програмування, прозора вводячи довгочасні дані, керування паралелізмом, відновлення даних, асоційовані запити й інші можливості. Так у 1991 році з'явилася група ODMG, яка посіла особливе місце в галузі стандартизації об'єктно–орієнтованих СКБД.

Об'єктно - орієнтовані моделі бази даних (Object-orientire data base)

Одним із підходів до забезпечення сумісності систем баз даних з парадигмою об'єктно–орієнтованого проектування зв'язаний з розширенням системи розуміння, що лежить в основі об'єктно–орієнтованих мов програмування, таких як C++ або Java, в напрямку підтримки засобів постійного зберігання даних. В традиційному програмуванні розуміється, що після завершення циклу роботи програми її об'єкти безповоротно втрачаються, в той час як принципові умови роботи будь-якої СУБД полягають в тому, що об'єкти повинні зберігатися необмежено довгий час, поки не будуть змінені або знищені примусово, як це відбувається у файлових системах. „Чисту” об'єктно-орієнтовану модель представлення даних, названу *ODL* (*Object Definition Language-мова визначення об'єктів*), яка в свій час була стандартизована дослідницькою групою Object Data Management Group (ODMG). На рис.1.13 показана структура об'єктно-орієнтованої БД.

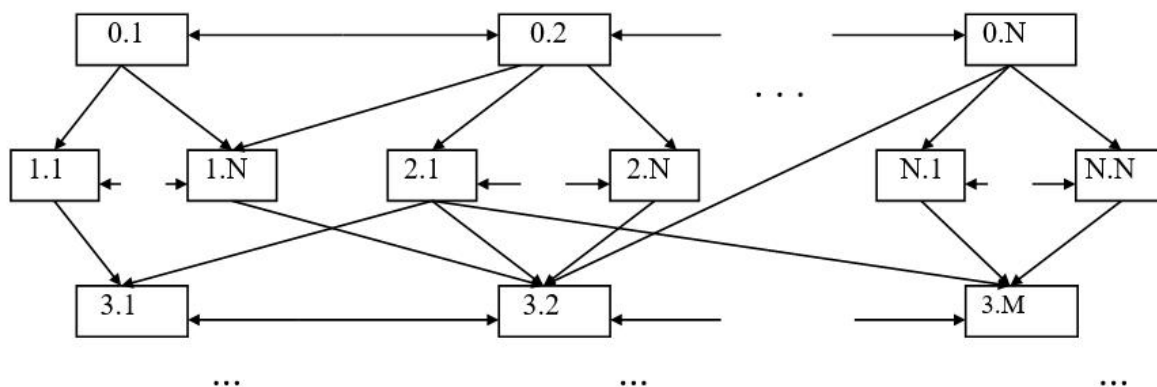


Рис.1.13 Об'єктно – орієнтовані БД

Об'єктно-реляційні моделі бази даних (Object-relational data base)

Ця модель являється частиною самого останнього стандарту SQL, названого SQL-99 (або SQL-1999, також SQL3), і являє собою варіант розширення звичайної реляційної моделі за рахунок формалізації багатьох загальноприйнятих концепцій об'єктно-орієнтованого проектування. Вказаний стандарт служить основою для побудови об'єктно-реляційних систем баз даних, які сьогодні випускаються всіма основними постачальниками комерційних СУБД. Ці системи, однак, значно відрізняються в деталях практичної реалізації початкових концепцій. На рис.1.14 показана структура об'єктно-реляційної БД.

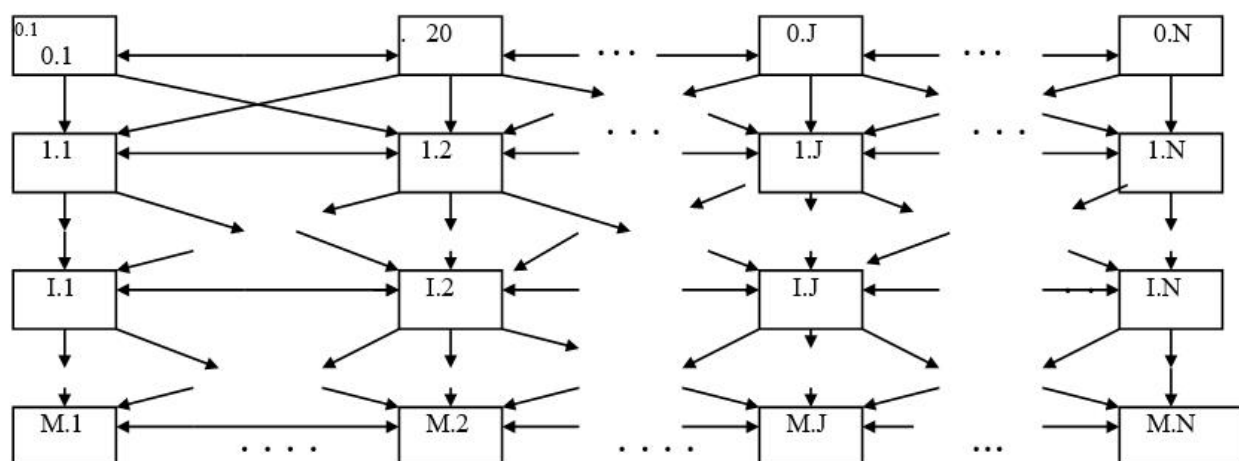


Рис.1.14 Об'єктно–реляційна модель БД

Напівструктуровані моделі бази даних (Semistructured data base)

Одне із останніх досягнень, призначених вирішити велику кількість актуальних проблем технологій СУБД, включаючи, необхідність об'єднання традиційних баз даних з іншими джерелами інформації, такими як Web-сторінки самої різноманітної структури. В цей час як об'єктно-орієнтовані або об'єктно-реляційні системи пропонують використання фіксованих схем для кожного класу або відношень, модель напівструктурованих даних відрізняється набагато більш високою гнучкістю представлення компонентів інформації. Найбільш яскравим практичним втіленням моделі напівструктурованих даних є XML (eXtensible Markup Language-розширювана мова розмітки). XML - це специфікація опису „документів”, що представляють собою набір вкладених елементів даних, функції яких визначаються відповідними тегами. Підтримка XML є найважливішим компонентом систем, здійснюючих посередницькі функції між різнорідними джерелами даних, служить цілям гнучкого представлення інформації в базах даних.

Розподілені моделі бази даних (Distributed data base)

РпБД- множина логічно взаємозалежних БД розподілених у комп'ютерній мережі. Сукупність баз даних, фізично розподілена за взаємо зв'язаними ресурсами обчислювальної

системи і доступна для спільного використання в різних додатках. Розподілена система керування базами даних (РСКБД) - це програмне забезпечення, яке керує РБД і надає такі механізми доступу до них, що їх застосування дає користувачу можливість працювати з РБД як з однією цілісною базою даних. Розподілені системи баз даних (РСБД) – це РпБД разом із ПК РСКБД. Архітектура розподілених БД в ГРКС представлена на рис.1.15.

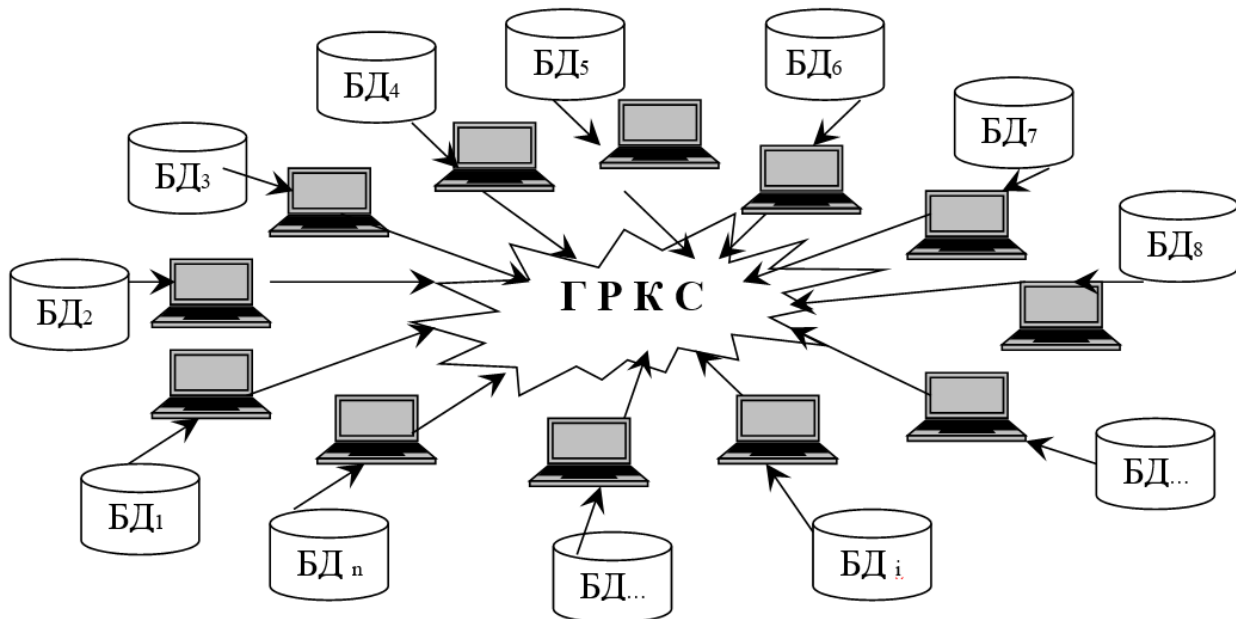


Рис. 1.15. Архітектура розподілених моделей БД в ГРКС

Для розподілених баз даних відома оцінка коефіцієнт ефективності K_e .

$$K_e = \frac{\sum_{i=1}^{N_{ik}} V_i + \sum_{j=1}^{N_{ij}} V_j + \sum_{k=1}^{N_{ikn}} V_k}{V_{ISP}} \quad (1.2)$$

який визначається відношенням пропускної здатності зовнішнього каналу (від ІЦЦ до Інтернет-провайдерів) пропускної здатності приєднаних каналів навантаження ІЦЦ-ІЦП, ІЦ, ІК; V_i - швидкість модему абонентів кбіт/с, i -інтернет-центру кбіт/с; N_k -кількість –абонентів, комутованих ліній(модемів); V_j -швидкість модему(БЗ) j -інтернет-центру кбіт/с; N_{iij} - кількість Інтернет-центрів безпосереднього підключення; V_k -швидкість модему (по БЗ) k -інтернет-центру кбіт/с; N_{iij} -кількість проміжних Інтернет-центрів; V_{ISP} - швидкість каналу до Інтернет-провайдера, кбіт/с. Для доступу до розподілених баз даних ресурсів комерційної інформації, можна здійснити через WEB-портал. Розподіл пріоритетів між розділами і сторінками сайту, зробить їх основними точками входу на сайт, що дозволить потенційному користувачеві швидко знайти необхідну йому інформацію. Архітектура інтернет-сайту складається з клієнтської частини, адміністративної частини та програмної частини. Архітектура інтернет-сайту представлена на рис.1.16.



Рис.1.16.Архітектура інтернет-порталу

Багатовимірні моделі бази даних (Multidimesenion data base)

Бази даних, що підтримує багатовимірну модель даних на концептуальному рівні та призначена для інтерактивного аналітичного опрацювання агрегатних історичних і прогнозованих даних. Основні поняття багатовимірної моделі даних: гіперкуб даних – rel; вимір - V; атрибут - A; комірка - X; значення – rel (V, A). Гіперкуб даних містить один або більше вимірів і є впорядкованим набором комірок. Кожна комірка визначається одним і лише одним набором значень вимірів – атрибутів. Комірка може містити дані – значення або бути порожньою.

Куби даних (data cubes)

Сучасне досягнення в галузі технологій інтеграції інформації , баз даних , реалізоване в ряді комерційних СУБД , зв'язані з підтримкою моделі кубів даних), де інформація інтерпретується в контексті багатомірного простору.

Ранні підходи до проблем інтеграції інформації передбачали утворення „федерацій,, , де кожна система баз даних могла генерувати запити, „зрозумілі,, іншим системам. Більш сучасні рішення зв'язані із застосування сховищ даних , де дані транслюються в глобальну схему і копіюються в центральні сховища. Альтернативна архітектура допускає використання механізму медіаторів-віртуальних сховищ, які надають можливості активізації запитів , звернених до глобальної схеми: дані не копіюються, а запити транслюються з врахуванням особливостей конкретних джерел інформації. Багато корпорацій зайняті збором багатомірних даних, для використання застосувань в підтримці прийняття рішень (decision-support), які дозволять, наприклад, проаналізувати господарську діяльність підприємства на основі інформації про продаж вироблених товарів, включно: а) дату і час оборудки; б) найменування пункту продажі; в) найменування товару , його колір, артикул, розмір.

Простори даних (Spasiousness data base)

Простори даних – нова абстракція керування даними . У випадку роботи у всесвітній мережі з величезною кількістю даних ресурсів. Прикладами таких задач є міжнародні транснаціональні корпорації, туристичний бізнес – збирання інформації про місця відпочинку її інтеграції та збирання у внутрішніх БД, геоінформаційні системи – на сьогодні ще не розроблені єдиним стандартом подання такої інформації.

Надвеликі бази даних (Very Large Database, VLDB)

Надвеликі бази даних— це база даних, яка займає надзвичайно великий об'єм на пристрої фізичного зберігання. Термін має на увазі максимально можливі об'єми БД, які визначаються останніми досягненнями в технологіях фізичного зберігання даних і в технологіях програмного оперування даними. Конкретне визначення поняття «надзвичайний великий об'єм» змінюється у часі; в теперішній час вважається, що це є об'єм, вимірюваний за меншою мірою терабайтами, а в останній час — петабайтами. Надвеликі бази і склади даних вимагають особливих підходів до логічного і системно-технічного проектуванню, звичайно викананому в рамках самостійного проекту, зміст якого в тому, щоб знайти таке системотехнічне рішення, яке дозволило б доцільно працювати з такими великими об'ємами. Таке рішення можливе при належності трьох умов: спеціального рішення для дискової підсистеми, спеціальних версій операційного середовища и спеціальних механізмів звертання СКБД до даних.

Сховища даних (Data Warehouse, DW)

Сховище даних – новий підхід до керування базами даних, які вміщують структуровані і згруповані дані, та перспективний напрям розвитку засобів аналізу даних і прийняття рішень менеджерами.

Сховище даних використовують для оброблення даних , які:

- а) надходять в різних форматах;
- б) можуть розміщатися в різних місцях ;
- в) відображають поточний стан інформації;
- г) не змінюються

Для опрацювання та зберігання даних великих обсягів використовуються реляційні БД (наприклад Access). Такі бази складаються з кількох пов'язаних між собою таблиць, данні яких потрібні для побудови запитів. Однак такий спосіб зберігання даних досить складно змінювати користувачеві-непрограмісту, а виконати аналіз наявної інформації він не в змозі. Ці дані аналізують і перетворюють у предметно-орієнтовані, інтегровані, розташовують із дотримання хронології. Дані можна бути успадкованими (наприклад, із реляційних БД), оперативними та зовнішніми. Вони можуть мати різні формати; тому спочатку за допомогою відповідних програм їх уніфікують, після чого переносять у сховище даних. Кінцевий користувач одержує дані зі сховища безпосередньо або за допомогою тематично підібраних інформаційних вітрин- сховищ даних, які містять зменшену (зрізану) частину даних. Вони створюються з метою прискорення оброблення запитів для вузького кола користувачів організації. Окрім функції збереження і впорядкування даних, сховища містять складну систему керування (перевірка достовірності, управління апаратним і програмним забезпеченням, даними і метаданими, персоналом і процедурами). Існує два основних методи оброблення даних : операційний та аналітичний. Операційне опрацювання даних потрібне для щоденної підтримки роботи підприємства. Воно здійснюється системами оброблення транзакцій у реальному часі (On-Line Transaction Processing, OLTP). Дані, що обробляються, після проведення розрахунків надалі не змінюються. Аналітичне оброблення даних ґрунтується на розробленні загальної стратегії підприємства. Воно проводиться за допомогою систем прийняття рішень (On-Line Analitical Processing, OLAP).

Модель сховища даних може бути багатовимірною або реляційною. Реляційна модель (ROLAP) використовує технологію багатовимірних БД (Multi Dimentional Database, MDD) і зберігає дані у вигляді сукупності логічно впорядкованих масивів, що значно прискорює швидкість оброблення запитів, але через надмірність даних вона суттєво збільшує обсяг файлу.

Багатовимірна модель сховища даних складається з таблиць–фактів, які, у свою чергу, містять певні дані у вигляді елементів, ієрархій та атрибутів.

Сховища містять засоби візуалізації даних, що включають: оперативну аналітичну обробку(OLAP) – інтерактивну технологію маніпуляції даними і виведення на екран агрегованих значень, геоінформаційні системи; гістограми, кольорове кодування секторні діаграми, деревовидні карти, тривимірні тренди, діаграми розсіювання. Такі фірми, як Informix, Oracle, Sybase, постачають відповідні інструментальні засоби для конструювання сховищ даних, а також інформаційних вітрин.

Комп'ютерні бази даних (Computers data base)

Комп'ютерні бази даних діляться на бази даних з доступом в режимі online, offline і через Internet (рис.1.17.).

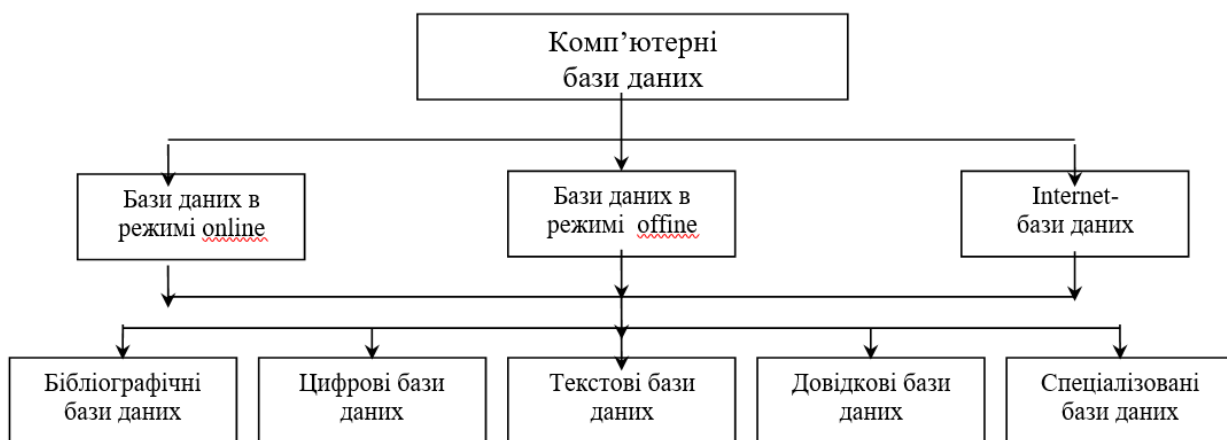


Рис.1.17. Класифікація комп'ютерних баз даних

Бази даних з доступом в режимі online (online databases) зберігаються в центральному банку даних. Доступ до них здійснюється посередністю комп'ютера (або іншого термінала)

через телекомунікаційну мережу. Доступ, пошук і аналіз Internet-баз даних (internet databases) здійснюється посередністю Internet. А також доступ, пошук і аналіз баз даних глобальних комп'ютерних систем здійснюється посередністю глобальних комп'ютерних систем. Відомості із них можна завантажувати і зберігати на комп'ютері або допоміжним запам'ятовуючий пристрій. Описані види баз даних в свою чергу поділяються на бібліографічні, цифрові, текстові, довідкові і спеціалізовані.

Методи оптимізації і підвищення ефективності БД.

Основними характеристиками оптимізації і підвищення ефективності є: здешевлення системи СУБД; збільшення швидкості пошуку доступу до даних; зменшення або стиснення об'єму інформації; розробка різноманітних типів третинних пристроїв зберігання (tertiary storage devises) даних;

- тенденції росту системи ;
- підвищення продуктивності системи ;
- паралельні обчислення;
- розвиток системи клієнт/сервер і багаторівневої архітектури;
- підвищення надійності захисту від помилок;
- захист від несанкціонованого доступу до БД тощо.

Викладені класи та характеристики БД, показують їх широкі можливості у сучасних інформаційних системах. В той ж час по замовчуванню всі відомі БД застосовують кодування на основі двійкової системи числення базису Радемахера .В цьому плані потенційні можливості БД недовикористовуються у зв'язку з наступними функціональними обмеженнями кодів базису Радемахера.:

- незахищеність даних від помилок;
- велика надлишковість кодування службової інформації;
- незахищеність від не санкціонованого доступу;
- використання позиційної системи числення приводить різної ваги впливів помилок в окремих розрядів двійкових чисел, що може привести до катастрофи БД;
- низька швидкодія виконання логічних та арифметичних операцій в наслідок наявності наскрізних переносів.

Проведені функціональні обмеження організації відомих БД на основі кодування в ТЧБ Радемахера визначає актуальність розробки теорії та технології побудови БД в інших ТЧБ. Особливо це стосується ТЧБ Галуа, в якому досягається найвищий рівень компактності функціонального кодування даних з внутрішнім системним захистом від помилок та не санкціонованого доступу.

Визначені систематичні характеристики даних, які включають час зберігання даних та їх об'єм, швидкість запиту та зчитування, а також захист від помилок і несанкціонованого доступу ,що дозволяє конкретизувати місце моделей теоретико-графових та теоретико-множинних баз даних серед інших моделей.

Описані існуючі типи баз даних та рівні їх проектування, що дозволяє визначити перспективи удосконалення та підвищення ефективності баз даних глобальних комп'ютерних систем.

Виконана класифікація баз даних, яка дозволила встановити взаємозв'язок між розподіленими базами даних і надвеликими комп'ютерними базами даних.

Досліджені архітектурні характеристики моделей баз даних які підтверджують перспективний розвиток об'єктно-орієнтованих та об'єктно-реляційних баз даних. виконана оцінка коефіцієнта ефективності розподілених баз даних яка враховує пропускну здатність інформаційних каналів та швидкість модемів комунікаційних центрів та абонентів.

Обґрунтовані перспективи зберігання даних у конгломератів епохальних моделей сховищ даних в базисі Галуа, які можуть бути застосовані у глобальних розподілених комп'ютерних системах.

РОЗДІЛ 2

ТЕОРІЯ ПОЛІВ ГАЛУА

2.1. Визначення поля.

Зупинимось детальніше на аксіомах, що визначають поняття поля.

Полем називається сукупність деяких елементів, для яких введено два закони композиції, що підкоряються наступним аксіомам:

Аксіома I. Всі елементи поля утворюють абелеву групу відносно першого закону композиції (означатимемо його знаком $+$ і називати складанням).

Аксіома II. Усі елементи поля, за винятком одиниці першої групи (якою ми називатимемо нулем), утворюють абелеву групу відносно другого закону композиції (який ми називатимемо множенням).

Аксіома III. Добуток елементів поля дорівнює нулю тоді і тільки тоді, якщо один з них дорівнює нулю.

Аксіома IV. Має місце дистрибутивний закон: $a(b + c) = ab + ac$.

2.1.1. Типи полів.

Сукупності 1) усіх дійсних чисел, 2) усіх комплексних чисел, 3) усіх чисел алгебри, тобто чисел, що задовольняють рівнянням алгебри з раціональними коефіцієнтами, є також полями.

Сукупності раціональних функцій від однієї або декількох змінних, коефіцієнти яких є раціональними числами (чи взагалі належать якому-небудь заданому полю), називається полем раціональних функцій.

Нехай дано одне або декілька рівнянь алгебри з раціональними коефіцієнтами. Сукупність раціональних функцій від коренів цих рівнянь складає поле, що називається полем алгебраїчних чисел, що і становитиме предмет даного дослідження.

Сукупність раціональних функцій від декількох змінних, зв'язаних однією або декількома залежностями алгебри, називається полем функцій алгебри.

Існують поля, що складаються з кінцевого числа елементів. Прикладом таких полів може служити сукупність класів порівнянь по простому модулю. Причина, чому сукупність класів порівнянь по складеному модулю не є полем, полягає в тому, що аксіома III вимагає, щоб добуток двох елементів поля дорівнював нулю тоді і тільки тоді, якщо принаймні один з множників дорівнює нулю. В той же час порівняння $x \cdot y = 0 \pmod{mn}$ може виконуватись, якщо ми покладемо $x = m$, $y = n$, причому ні m ні n не порівнянні з 0 при модулю mn .

Кінцеві поля найзагальнішого вигляду можна отримати при розгляді раціональних функцій від коренів поліномів, незвідних по простому модулю p , тобто незрівняних по модулю p з добутком поліномів нижчого степеня, і при цьому відкидати кратності p . Вони носять назву полів Галуа або полів порівнянь по подвійному модулю.

Гензель (K. Hensel) ввів у розгляд поля, природа яких абсолютно відмінна від природи числових і функціональних полів, так звані поля p -адичних чисел. Елементами поля p -адичних чисел є нескінченні ряди $a_0 + a_1p + a_2p^2 + \dots$, розташовані по зростаючих степенях простого числа p з цілими раціональними коефіцієнтами. Такі ряди завжди розбігаються. Проте над цими рядами можна виконувати формальні дії додавання і множення, які задовольняють усім аксіомам поля, p -адичні числа служать хорошим інструментом при вивченні властивостей звичайних чисел алгебри.

2.1.2. Властивості полів чисел алгебри

Поля чисел алгебри розглядаються як сукупності раціональних функцій від одного або декількох коренів рівнянь алгебри. Можна показати, що будь-яке таке поле може бути утворене коренем одного рівняння. Іншими словами, справджується наступна теорема:

Теорему. У всякому полі, утвореному коренями $a_1, a_2, \dots, a_m$ рівнянь алгебри $f_1(x)=0, f_2(x)=0, \dots, f_m(x)=0$ з раціональними коефіцієнтами (які можуть частково або повністю співпадати), існує примітивний елемент, тобто елемент, через який можуть бути раціонально виражені усі елементи поля.

Наслідок. Якщо $x, y, z, \dots$ - величини, що утворюють поле, то примітивну величину поля можна підшукати у вигляді $c_1x + c_2y + c_3z + \dots$, де $c_1, c_2, c_3, \dots$ спеціально підібрані раціональні числа.

Теорема. Будь-яка величина поля чисел алгебри задовольняє рівнянню алгебри з раціональними коефіцієнтами.

Степінь рівняння, що є незвідним і якому задовольняє примітивна величина поля, не залежить від вибору примітивної величини і носить назву степеня поля.

Теорема. Якщо $\theta_1, \theta_2, \dots, \theta_n$ - корені незвідного полінома $f(x)$, а $R(x)$ - раціональна функція (знаменник якої не ділиться на $f(x)$), то поліном:

$(t-R(\theta_1))(t-R(\theta_2)) \dots (t-R(\theta_n))$ є степенем незвідного полінома.

Теорема. Якщо 0 - примітивний елемент поля степеня n , то всякий елемент цього поля можна представити у вигляді полінома степеня не вище $n-1$:

$$a_0 + a_1\theta + \dots + a_{n-1}\theta^{n-1},$$

де коефіцієнти $a_1, a_2, \dots, a_{n-1}$ - раціональні числа. Це представлення єдине.

Поняття незвідності можна узагальнити на випадок, коли коефіцієнти поліномів не є раціональними числами, а належать до заданого поля. Поліном зводиться в полі K , якщо його можна розкласти на добуток поліномів нижчого степеня, коефіцієнти яких є величини поля K .

Приклад. Поліном x^4+1 є незвідним в раціональному полі, але в полі $K(\sqrt{2})$ (тобто поля раціональних функцій від $\sqrt{2}$) зводиться. Дійсно,

$$X^4+1=(x^2+\sqrt{2}*x+1)(x^2-\sqrt{2}*x+1)$$

Всі властивості, що мають місце для поліномів, що не зводяться в раціональному полі, зберігають свою силу й для поліномів, що не зводяться в поле K .

Проте ті практичні методи, які ми запропонували для розпознавання поліномів, що наводилися і не наводилися, тут вже не можуть бути проведені. Основна причина цього полягає в тому, що кожне ціле число (тут ми не будемо останавливатися на цьому понятті) алгебри може бути представлене у вигляді твору двох цілих чисел незліченним числом способів, через існування так званих одиниць алгебри.

2.2. Група Галуа. Співвідношень між коренями поліномів.

Нехай рівняння (без кратних коренів): $F(x)=x^n+a_1x^{n-1}+\dots+a_{n-1}x+a_n=0$ має корені $x_1, x_2, \dots, x_n$. Між цими коренями матимуть місце співвідношення, що мають форму поліномів з коефіцієнтами, раціонально залежними від коефіцієнтів $a_1, a_2, \dots, a_n$, а також від заданих величин, які утворюють поле, що називається областю раціональності. Прикладом можуть служити співвідношення між елементарними, симетричними функціями від коренів та з коефіцієнтами:

$$\begin{cases} x_1 + x_2 + \dots + x_n = -a_1, \\ x_1x_2 + x_1x_3 + \dots + x_{n-1}x_n = a_2 \\ x_2x_3 \dots x_n + x_1x_3 \dots x_n + \dots + x_1x_2 \dots x_{n-1} = (-1)^{n-1}a_{n-1} \\ x_1x_2 \dots x_{n-1}x_n = (-1)^na_n \end{cases} \quad (2.1)$$

Між коренями $x_1, x_2, \dots, x_n$ можна виконувати будь-які перестановки, при цьому співвідношення (1.1) буде залишатися справедливими. Однак існують рівняння, між коренями яких дане співвідношення не виконується. Прикладом може служити рівняння: $x^4 + x^3 + x^2 + x$

$+ 1 = 0$, що має, як відомо, корінь 5-го степеня з одиниці: $x_1=\varepsilon, x_2=\varepsilon^2, x_3=\varepsilon^3, x_4=\varepsilon^4 (\varepsilon=e^{\frac{2\pi}{5}})$

Між цими коренями виконуються наступні співвідношення:

$$x_1x_4=1, x_1^2x_3=1, x_1^3x_2=1, x_1^4x_4=1 \quad (2.2)$$

Не всяка підстановка між коренями буде зберігати ці співвідношення в силі. Наприклад, підстановка (x_1, x_2) порушує відношення, переводячи його ліву частину в $x_2 x_4$ тобто величину, рівну ε , а не одиниці. Перевіривши кожну з 24 підстановок симетричної групи 4-го степеня, можна переконатися, що співвідношень (2.2) не порушують такі заміни: $(x_1x_2x_4x_3)$ (x_1x_4) (x_2x_3) $(x_1x_2x_4x_2)$.

Ці підстановки утворюють циклічну групу. Нехай в загальному вигляді між коренями рівняння мають місце співвідношення:

$$\varphi_i(x_1x_2, \dots, x_n)=0 \quad (i=1.2.\dots) \quad (2.3)$$

Відносно кожної підстановки симетричної групи між $x_1 x_2, \dots, x_n$ χ завжди можна сказати (принаймні теоретично), чи порушує вона хоч одне з цих співвідношень, або зберігає всі їх у силі (тобто переводить кожне співвідношення (2.3) в яке-небудь з цих же співвідношення). Справедлива така

Теорема. Сукупність усіх підстановок, що не порушують всіх можливих співвідношень між коренями алгебраїчного доповнення, утворює групу.

Група підстановок, що не порушує всіх можливих співвідношень між коренями рівняння $f(x) = 0$, носить назву групи Галуа або просто групи цього рівняння.

З усіх співвідношень (2.3) розглянемо спочатку співвідношення, в які входить тільки один з коренів нашого рівняння, наприклад x_1 . Нехай $\Phi(x_1)$ буде одне з таких співвідношень і $f_1(x) = 0$ буде незвідне рівняння, якому задовольняє x_1 . Тоді $\Phi(x) = 0$ і $f_1 = 0$ мають спільні корені, а тому $\Phi(x)$ має ділитися на $f_1(x)$. Таким чином $\Phi(x) = 0$ має місце для всіх коренів незвідного множника полінома $f(x)$. З іншого боку, $f_1(x_1) = 0$ само по собі є відношення, яке порушується від всіх підстановок. Якщо, зокрема, рівняння (2.1) незвідне, то співвідношення, в які входить один корінь, не накладають ніякого обмеження на його групу Галуа. Якщо ж рівняння (2.1) зводиться, то відразу видно, що його група Галуа інтранзитивна.

Таким чином, ліва частина всякого співвідношення $\Phi(x_1) = 0$, має тільки один корінь x_1 , і повинна ділитися на незвідний поліном $f_1(x)$, якому задовольняє x_1 . Це можна записати так: $\Phi(x) = 0 \pmod{f_1(x)}$, де x тепер треба вважати змінною величиною. Поліном $f_1(x)$ називається першим основним модулем.

Розглянемо співвідношення типу (2.3), в які входять два кореня, наприклад x_1 і x_2 . ліву частину рівняння потрібно розкласти на незвідні множники в полі $K(x_1)$ (тобто у полі раціональних функцій від x_1) і позначити той незвідний поліном, якому задовольняє x_2 через $f_2(x_1, x)$. Нехай $\Phi(x_1 x_2) = 0$ з будь-яким співвідношенням між коренями x_1 і x_2 . Розглянемо поліном $\Phi(x_1, x)$, який можна вважати поліномом від змінної x з коефіцієнтами з поля $K(x_1)$. Тоді поліном $\Phi(x_1, x)$ повинен ділитися на $f_2(x_1, x)$:

$$\Phi(X_1, X) = 0 \pmod{f_2(X_1, X)}. \quad (2.4)$$

Якщо $\psi(x_1, x) = 0 \pmod{f_2(x_1, x)}$, то $\psi(x_1 x_2) = 0$, так як $f(x_1, x_2) = 0$.

Нехай степінь полінома $f_1(x) \in n_1$ і степінь полінома $f_2(x_1 x) \in n_2$. Далі можна вважати, що в $f_2(x_1 x)$ величина x_1 входить як поліном. Вважаючи $\xi_1 \xi_2$ змінними, потрібно розділити $\Phi(\xi_1, \xi_2)$ на $f_2(\xi_1, \xi_2)$. Це можна виконати, не вводячи дрібних функцій від ξ_1 (коефіцієнтами при $\xi_2^{n_2}$ в поліномі $f_2(\xi_1, \xi_2)$ служать одиниці. Тоді можна отримати:

$$\Phi(\xi_1, \xi_2) = f_2(\xi_1, \xi_2) * q_2(\xi_1, \xi_2) + r(\xi_1, \xi_2) \quad (2.5)$$

де поліном $r(\xi_1, \xi_2)$ має степінь, не вищий n_1-1 щодо ξ_2 . Тому всі коефіцієнти при ступенях ξ_2 полінома $r(\xi_1, \xi_2)$, маючи з незвідним поліномом $f(\xi_1)$ спільні корені, діляться на нього, і таким чином поліном $r(\xi_1, \xi_2)$ може бути представлений в формі $f(\xi_1) \cdot q_1(\xi_1, \xi_2)$, де $q_1(\xi_1, \xi_2)$ - деякий поліном, так що формула (2.6) набуває вигляду:

$$\Phi(\xi_1, \xi_2) = f_1(\xi_1) * q_1(\xi_1, \xi_2) + f_2(\xi_1, \xi_2) * q_1(\xi_1, \xi_2) \quad (2.6)$$

де $q_1(\xi_1, \xi_2)$ і $q_2(\xi_1, \xi_2)$ - деякі поліноми від ξ_1 і ξ_2

Якщо замінити x_1 на будь-який інший корінь x_i полінома $f_1(x_1)$, то $f_2(x_1, \xi_2)$ є той поліном, що не зводиться в полі $K(x_1)$. Нехай

$$f(x_2) = (\xi_2 - x_1) f_2(x_1, \xi_2) f_2^{(1)}(x_1, \xi_2) \dots f_2^{(k)}(x_1, \xi_2). \quad (2.7)$$

Тому завжди можна в $f_2(x_1, x_2) = 0$ (а отже і в будь-якому співвідношенні між x_1 і x_2) замінити x_1 на будь-який корінь полінома $f_1(x)$, а x_2 - на якийсь інший корінь, відмінний від x_1 , без того, аби співвідношення порушилося.

Нехай $f_3(x_1, x_2, \zeta_3)$ буде поліномом, що не зводиться в полі $K(x_1, x_2)$, і якому задовольняє $\zeta_3 = x_3$, і хай його степінь буде n_3 . Якщо розділити поліном $\Phi(x_1, x_2, \zeta_3)$ на $f_3(x_1, x_2, \zeta_3)$, то можна отримати:

$$\Phi(x_1, x_2, \zeta_3) = f_3(x_1, x_2, \zeta_3) \cdot q_3(x_1, x_2, \zeta_3) + r(x_1, x_2, \zeta_3). \quad (2.8)$$

Залишок $r(x_1, x_2, \dots)$, будучи не вище степеня $n_3 - 1$, має спільний з $f(x_1, x_2, \zeta_3)$ корінь $\zeta_3 = x_3$. Тому якщо розділити $\Phi(\zeta_1, \zeta_2, \zeta_3)$ на $f(\zeta_1, \zeta_2, \zeta_3)$ (при змінних $\zeta_1, \zeta_2, \zeta_3$, але по змінній ζ_3), то залишок $r(\zeta_1, \zeta_2, \zeta_3)$ при довільному ζ_3 перетворюватиметься на нуль, якщо покласти $\zeta_1 = x_1, \zeta_2 = x_2$, а тому рівність $r(\zeta_1, \zeta_2, \zeta_3) = 0$ є співвідношенням між x_1 і x_2 з ζ_3 у вигляді параметра, і поліном $r(\zeta_1, \zeta_2, \zeta_3)$ може бути представлений у формі (2.9), де ζ_3 може входити в поліноми q_1 і q_2 у вигляді параметра. Підставляючи в 2.10), можна отримати:

$$\Phi(\zeta_1, \zeta_2, \zeta_3) = f_3(\zeta_1, \zeta_2, \zeta_3) q_3(\zeta_1, \zeta_2, \zeta_3) + f_1(\zeta_1) q_1(\zeta_1, \zeta_2, \zeta_3) + f_2(\zeta_1, \zeta_2) q_2(\zeta_1, \zeta_2, \zeta_3) \quad (2.10)$$

Поліном $f_3(x_1, x_2, \zeta_3)$ є незвідним в полі $K(x_1, x_2)$ дільником полінома $f(\zeta_3)$:

$$f(\zeta_3) = f_3(x_1, x_2, \zeta_3) (\zeta_3 - x_2) (\zeta_3 - x_1) F(x_1, x_2, \zeta_3) \quad (2.11)$$

Ця рівність при довільному ζ_3 є співвідношенням між x_1 і x_2 , а тому в ньому можна замінити корені x_1, x_2 іншими коренями x_i, x_j , якщо при цьому не порушуються співвідношення типу $\Phi(x_1, x_2) = 0$. Тому поліном $f(x_i, x_j, \zeta_3)$ має коренями (відносно ζ_3) деякі з коренів $x_1, x_2, \dots, x_n$, причому відмінні від x_i і x_j . Дійсно, переводячи в співвідношенні (2.11) x_1 в x_i і x_2 в x_j , можна отримати:

$$f(\zeta_3) = f_3(x_i, x_j, \zeta_3) (\zeta_3 - x_i) (\zeta_3 - x_j) F(x_i, x_j, \zeta_3) \quad (2.12).$$

Тут ліва, а тому і права, частина має коренями. $x_1, x_2, \dots, x_n$, причому x_i і x_j - корені множників $\zeta_3 - x_i$ і $\zeta_3 - x_j$ правої частини.

Позначаючи через $f_k(x_1, x_2, \dots, x_{k-1}, \zeta_k)$, що є незвідним у полі $K(x_1, x_2, \dots, x_{k-1})$, дільник полінома $f(\zeta_k)$, що має коренем $\zeta_k = x_k$ і продовжуючи міркування для значень $k = 4, 5, \dots, n$, можна переконатися, що ліва частина будь-якого співвідношення $\Phi(x_1, x_2, \dots, x_n) = 0$ може бути представлена в наступній формі:

$$\Phi(\zeta_1, \zeta_2, \dots, \zeta_n) = f_1(\zeta_1) q_1(\zeta_1, \zeta_2, \dots, \zeta_n) + f_1(\zeta_1, \zeta_2) q_2(\zeta_1, \zeta_2, \dots, \zeta_n) + \dots + f_n(\zeta_1, \zeta_2, \dots, \zeta_n) q_n(\zeta_1, \zeta_2, \dots, \zeta_n) \quad (2.13)$$

де $q_i(\zeta_1, \zeta_2, \dots, \zeta_n)$ ($i = 1, 2, \dots, n$) - поліноми з раціональними коефіцієнтами. Таким чином, кожна дана підстановка між коренями $x_1, x_2, \dots, x_n$ зберігає в силі будь-які співвідношення між $x_1, x_2, \dots, x_n$ тоді і лише тоді, якщо вона зберігає в силі співвідношення

$$f_1(x) = 0, f_2(x_1, x_2) = 0, f_3(x_1, x_2, x_3) = 0, \dots, f_n(x_1, x_2, \dots, x_n) = 0 \quad (2.14)$$

Ліві частини співвідношень (2.14) носять назву основних модулів рівняння (2.14). Вони характеризують групу Галуа в тому сенсі, що кожна підстановка належить до групи Галуа тоді і лише тоді, якщо вона зберігає в силі всі співвідношення (2.14).

Крім того, продовжуючи міркування, можна переконатися, що в групі Галуа існуватиме підстановка, що переводить x_1 у будь-який корінь x_{a1} першого модуля $f_1(\zeta_1)$. Дійсно, підставляючи в $f_2(\zeta_1, \zeta_2) = x_{a1}$ можна отримати поліном, що має коренем якісь з величин $x_1, x_2, \dots, x_n$, відмінні від x_{a1}

2.3. Основні властивості групи Галуа.

Теорема. Для того, щоб група рівняння $a_0 x^n + a_1 x^{n-1} + \dots + a_{n-1} x + a_n = 0$ була транзитивна, необхідно і достатньо, щоб рівняння було незвідним.

Цю теорему потрібно узагальнити на випадок кратної транзитивності.

Теорема. Для того, щоб група Галуа рівняння $a_0 x^n + a_1 x^{n-1} + \dots + a_{n-1} x + a_n = 0$ була k раз транзитивна, необхідно і достатньо, щоб перші його k основних модулів збігалися з модулями Коші.

Зв'язок між порядком групи Галуа і степенями основних модулів може бути сформульований у вигляді

Теорема. Порядок групи Галуа дорівнює добутку $n_1 \cdot n_2 \dots n_n$ степенів основних модулів.

Якщо група Галуа є симетрична, то вона n раз транзитивна, і тому всі n основних модулів через співпадають з модулями Коші. Степені цих модулів рівні, відповідно, $n, n-1, n-2, \dots, 3, 2, 1$, і ще раз можна прийти до того, що порядок симетричної групи рівний n .

Якщо група Галуа є знакозмінною, то вона $n-2$ рази транзитивна, а тому перші $n-2$ основних модуля повинні збігатися з модулями Коші. Степень $(n-1)$ -го модуля Коші дорівнює двом. Якщо він не зводиться, то матимем симетричну групу (n -й модуль у всіх випадках першого степеня). Для випадку знакозмінної групи залишається одна можливість, саме: $(n-1)$ -й модуль повинен розкладатися на лінійні множники. Тому степені послідовних основних модулів рівняння із знакозмінною групою Галуа такі: $n, n-1, n-2, \dots, 3, 1, 1$.

Наступна теорема є однією з найважливіших в теорії Галуа відносно симетричних функцій:

Теорема. Функція від коренів $x_1, x_2, \dots, x_n$ рівняння $a_0 x^n + a_1 x^{n-1} + \dots + a_{n-1} x + a_n = 0$, що не змінює своєї величини при всіх підстановках групи Галуа, раціонально виражається через коефіцієнти цього рівняння.

Приклад. Розглянемо рівняння:

$$f(x) = x^{p-1} + x^{p-2} + \dots + x + 1 = 0. \quad (2.15)$$

Воно є незвідним. З іншого боку, позначаючи через x_1 його корінь, можна виразити через нього інші корені так: $x_2 = x_1^2, x_3 = x_1^3, \dots, x_{p-1} = x_1^{p-1}$.

Його перший основний модуль є $f(\xi_1)$, тоді як всі модулі, починаючи з другого, є лінійними поліномами. Дійсно, в полі $k(x_1)$ поліном $f(\xi)$ розкладається на лінійні множники: $f(\xi) = (\xi - x_1)(\xi - x_1^2) \dots (\xi - x_1^{p-1})$.

Тому основними модулями рівняння (2.15) є наступні поліноми:

$$f_1 = \xi_1^{p-1} + \xi_2^{p-2} + \dots + \xi_{p-1}^1, f_2 = \xi_2^{p-1} - \xi_1^2, \dots, f_{p-1} = \xi_{p-1}^{p-1} - \xi_1^{p-1}.$$

Таким чином група Галуа рівняння (2.15) має порядок $p-1$ і вона є циклічною. Позначаючи через g - первісний корінь порівняння $x^{p-1} - 1 = 0 \pmod{p}$, через T - підстановку, що переводить x_1 в $x_1^g, x_1^{g^2}, \dots$, звернемо увагу на те, що T^k переводить x_1 в $x_1^{g^k}$ ($k = 1, 2, \dots, p-1$). Але оскільки g^{p-1} є найменшим степенем g , порівнянним з одиницею по модулю p , то звідси слідує, що T^{p-1} є найменший степінь підстановки $T=1$. Тому всі підстановки $1, T, T^2, \dots, T^{p-2}$ різні і таким чином вичерпують всю групу Галуа, порядок якої рівний теж $p-1$.

Якщо всі корені якого-небудь рівняння можуть бути раціонально виражені через один корінь, то рівняння називається нормальним. Не важко переконатися, що рівняння нормальне тоді і лише тоді, якщо всі його основні модулі, починаючи з другого, лінійні.

2.4. Підстановки групи Галуа.

Нехай, наприклад, задані корені x_1, x_2, x_3, x_4 . Якщо корені x_1, x_2, x_3, x_4 піддати підстановці (x_1, x_3) , то величина $z_1 = x_1 x_2 + x_3 x_4$, перейде у $z_2 = x_3 x_2 + x_1 x_4$. шляхом заміни кореня x_1 на корінь x_3 і кореня x_3 - на корінь x_1 .

Нехай a -величина поля, і хай $\varphi_1(x_1, x_2, \dots, x_n)$ і $\varphi_2(x_1, x_2, \dots, x_n)$ - два її різні вирази через корені. Тоді рівність $\varphi_1(x_1, x_2, \dots, x_n) + \varphi_2(x_1, x_2, \dots, x_n)$ можна розглядати як співвідношення між коренями рівняння. Якщо застосувати до цього співвідношення будь-яку підстановку групи Галуа, то це воно не повинно порушитися, а тому і в правій, і в лівій частині отримається однакова величина.

2.4.1 Автоморфізм нормального поля.

Щоб підійти до нового визначення групи Галуа, не залежного від вихідного рівняння, потрібно ввести поняття автоморфізму поля. Нехай кожен елемент поля переходить в якийсь інший, визначений елемент того ж поля. Автоморфізмом поля можна вважати відображення,

яке переводить величини α, β нашого поля відповідно в α_1, β_1 так, що сума $\alpha + \beta$ переводиться в $\alpha_1 + \beta_1$ і добуток $\alpha \beta$ - в $\alpha_1 \beta_1$.

З цього визначення витікають наступні властивості автоморфізму:

1°. Різниця переходить в різницю. Дійсно, нехай $\alpha \rightarrow \alpha_1, \beta \rightarrow \beta_1$, і нехай $\alpha - \beta \rightarrow x$. Тоді $\alpha - \beta + \beta$ переходить в $x \rightarrow \beta_1$, оскільки $\alpha \rightarrow \alpha_1$, то повинно мати місце $\alpha_1 = x + \beta_1, x = \alpha_1 - \beta_1$, тобто $\alpha - \beta \rightarrow \alpha_1 - \beta_1$.

2°. Ділене переходить в ділене. Доведення аналогічне.

3°. Нуль переходить в нуль, одиниця - в одиницю. Дійсно, $\alpha \rightarrow \alpha_1$ то $0 = \alpha - \alpha \rightarrow \alpha_1 - \alpha_1 = 0$, і $1 = \frac{\alpha}{\alpha} \rightarrow \frac{\alpha_1}{\alpha_1} = 1$.

4°. Всі раціональні числа залишаються на місцях.

5°. Якщо $\alpha \rightarrow \alpha_1$ і $f(x)$ - раціональна функція від x з раціональними коефіцієнтами, то $f(\alpha) \rightarrow f(\alpha_1)$.

6°. Корені рівнянь алгебри з раціональними коефіцієнтами переходять в корені тих же рівнянь. Насправді, якщо $\alpha \rightarrow \alpha_1$, то $f(\alpha) \rightarrow f(\alpha_1)$. Якщо при цьому $f(\alpha) = 0$, то в силу $0 \rightarrow 0$ повинно бути також $f(\alpha_1) = 0$.

7°. Якщо $x_1 \rightarrow x_1', x_2 \rightarrow x_2', x_n \rightarrow x_n'$ і притому $\Phi(x_1, x_2, \dots, x_n) = 0$, то і $\Phi(x_1', x_2', \dots, x_n') = 0$.

Таким чином якщо в даному полі лежать всі корені якого-небудь рівняння, то автоморфізм поля виробляє над цими коренями підстановки його групи Галуа. Розширюючи підстановки групи Галуа, можна отримати з них автоморфізми поля $K(x_1, x_2, \dots, x_n)$.

2.4.2. Нормальні поля. Теорема Лагранжа.

Аби можна було говорити про групу Галуа як про групу автоморфізму поля, необхідно, аби кожна підстановка групи Галуа не виводила величин даного поля з його границь. Тому поле, яке містить один корінь якого-небудь рівняння, повинно містити і всі інші його корені. Поле такого роду називається нормальним. Тоді групою Галуа нормального поля є сукупність всього його автоморфізму. Справедливі такі теореми.

Теорема. Примітивний елемент ζ нормального поля задовольняє незвідному рівнянню з раціональними коефіцієнтами, степінь якого дорівнює порядку групи поля.

Теорема. Нормальне поле містить елементи, що належать до кожної заданої підгрупи його групи Галуа.

Теорема Лагранжа. Якщо α належить до підгрупи групи Галуа нормального поля K , а β не змінюється від підстановок групи, то β раціонально виражається через α .

РОЗДІЛ 3

ГРУПА ГАЛУА

3.1. Початкові визначення групи Галуа.

3.1.1. Корені полінома. Розкладання полінома на лінійні множники.

У курсі алгебри доводяться наступні теореми:

Теорема. Будь-яке рівняння

$$a_0 x^n + a_1 x^{n-1} + \dots + a_{n-1} x + a_n = 0 \quad (3.1)$$

з дійсними або комплексними коефіцієнтами має принаймні один корінь, тобто таке значення змінної x , яке, будучи підставлено в рівняння (3.1), перетворює його ліву частину в нуль.

Теорема (Безу). Якщо рівняння $f(x) = 0$ має корінь число a , то поліном $f(x)$ ділиться на $x - a$.

Звідси випливає наступна властивість поліномів n -ної степені: вони не можуть мати більш як n різних коренів.

Якщо деякі з величин $a_1, a_2, \dots, a_n$ будуть рівними один одному, то говорять, що поліном $f(x)$ має кратні корені. Корінь a називається k -кратним, якщо $f(x)$ ділиться на $(x - a)^k$, але не ділиться на $(x - a)^{k+1}$. Для розпізнавання кратного кореня існує наступний критерій.

Теорема. Якщо a є для полінома $f(x)$ коренем k кратності, то для його похідної $f'(x)$ він є коренем $(k-1)$ кратності.

Розглянемо найбільший спільний дільник $f_1(x)$ поліномів $f(x)$ і $f'(x)$. Він містить тільки такі множники $x - a$, які входять в $f(x)$, і притому у степені, вищому, ніж перший. Якщо $x - a$ входить в $f(x)$ в k -тій степені, то в $f_1(x)$ він увійде в $(k - 1)$ степені. Якщо $f(x)$ не має кратного кореня, то $f_1(x)$ не матиме множників, лінійних відносно x , і тому дорівнюватиме сталому числу.

Нехай $x - a, x - a', x - a'', \dots$ будуть множники, що входять в $f(x)$ першого степеня; $x - \beta, x - \beta', x - \beta'', \dots$ в другій степені, і т. д.; нарешті, $x - \omega, x - \omega', x - \omega'', \dots$ множники, що входять в $f'(x)$ в самому вищому, k -тому степені ($k \leq n$). Введемо позначення:

$$X_1 = (x - a)(x - a')(x - a'')\dots, \quad X_2 = (x - \beta)(x - \beta')(x - \beta'')\dots,$$

$$X_k = (x - \omega)(x - \omega')(x - \omega'')\dots$$

Тоді поліном $f(x)$ можна представити так: $f(x) = a_0 X_1 \cdot X_2^2 \cdot X_3^3 \dots X_k^k$. Поліном $f_1(x)$ виразиться в наступному виді: $f_1(x) = X_2 \cdot X_3^2 \dots X_k^{k-1}$. Якщо ввести в розгляд $f_2(x)$, найбільший спільний дільник полінома $f_1(x)$ і його похідної $f'_1(x)$, то: $f_2(x) = X_3 \dots X_k^{k-2}$.

Якщо продовжити цей процес до отримання полінома, не залежного від x , то можна отримати: $f_{k-1}(x) = X_k$, $f_k(x) = \text{const}$.

Уміючи знаходити поліноми $f_1(x), f_2(x), \dots, f_k(x)$, можна виділити в поліномі $f(x)$ його множники $X_1, X_2, \dots, X_k$. Насправді, з отриманих співвідношень ми матимемо:

$$\frac{f(x)}{f_1(x)} = \varphi_1(x) = a_0 \cdot X_1 X_2 X_3 \dots X_{k-1} X_k, \quad \frac{f_1(x)}{f_2(x)} = \varphi_2(x) = X_2 X_3 \dots X_{k-1} X_k, \dots,$$

$$\frac{f_{k-2}(x)}{f_{k-1}(x)} = \varphi_{k-1}(x) = X_{k-1} X_k, \quad f_{k-2}(x) = \varphi_k(x) = X_k$$

звідки

$$X_1 = \frac{\varphi_1(x)}{a_0 \varphi_2(x)}, \quad X_2 = \frac{\varphi_2(x)}{\varphi_3(x)}, \dots, X_{k-1} = \frac{\varphi_{k-1}(x)}{\varphi_k(x)}, \quad X_k = \varphi_k(x)$$

Праві частини цих формул є цілими раціональними функціями від коренів $a_1, a_2, \dots, a_n$, що мають при цьому наступну властивість: вони не змінюються, якщо самим довільним чином переставляти один з одним корені $a_1, a_2, \dots, a_n$. Функції, що мають таку властивість, носять назву симетричних. Функції, представлені у формулах (3.5), носять, крім того, особливу назву елементарно-симетричних функцій.

Неважко переконатися, користуючись формулами (3.5), що всяка раціональна функція від коефіцієнтів $a_1, a_2, \dots, a_n$ може бути виражена як симетрична функція від коренів $a_1, a_2, \dots, a_n$. Дещо важче довести зворотну до неї теорему

Теорема. Всяка раціональна симетрична функція від коренів $a_1, a_2, \dots, a_n$, може бути представлена як раціональна функція від коефіцієнтів $a_1, a_2, \dots, a_n$. Якщо при цьому її коефіцієнти при степенях $a_1, a_2, \dots, a_n$ цілі числа, то і через $a_1, a_2, \dots, a_n$ вона виражається з цілими коефіцієнтами.

Приклад. $f(x) = x^3 + ax^2 + bx + c = (x - \alpha)(x - \beta)(x - \gamma), V = \alpha^3 + \beta^2 + \gamma^2$.

Модулі Коші наступні:

$$X = x^3 + ax^2 + bx + c, X_1 = x^2 + (a + a)x + (a^2 + aa + b), X_2 = x + (a + \beta + c).$$

Замінімо у виразі V букву γ на x і розділимо його на X_2 . Отримаємо в залишку $a^2\beta^2 + (a + \beta + a)^2$. Замінімо в цьому виразі β на x і розділимо його на X_2 . Отримаємо в залишку: $a^2 - 2b$. Сюди a не входить, так що це і є шуканий вираз V через a, β, γ : $a^2 + \beta^2 + \gamma^2 = a^2 - 2b$.

3.1.4. Результат.

Задано два поліноми:

$$f(x) = (x - x_1)(x - x_2) \dots (x - x_n) \quad (3.6)$$

$$g(x) = (x - y_1)(x - y_2) \dots (x - y_m) \quad (3.7)$$

Виведемо умову того, щоб вони мали принаймні один загальний корінь. Для цього необхідно і достатньо, щоб вираз:

$$R(f, g) = f(y_1)f(y_2) \dots f(y_m) \quad (3.8)$$

перетворювався в нуль.

Вираз (3.8) носить назву результанта поліномів $f(x)$ і $g(x)$. Будучи симетричною функцією від кореня полінома (6.7), результат може бути представлений, як ціла раціональна функція від коефіцієнтів цього полінома, у вираз якого входять також коефіцієнти полінома (3.6).

У виразі (3.8) коефіцієнти обох поліномів грають однакову роль. Щоб переконатися в цьому, перепишемо його так:

$$\begin{aligned} R(f, g) &= (y_1 - x_1)(y_1 - x_2) \dots (y_1 - x_n) \\ &\quad (y_2 - x_1)(y_2 - x_2) \dots (y_2 - x_n) \\ &\quad \dots \dots \dots \\ &\quad (y_m - x_1)(y_m - x_2) \dots (y_m - x_n) \end{aligned}$$

Збираючи множники по стовпцях і користуючись формулою (3.7), можна отримати:

$$R(f, g) = (-1)^{mn} g(x_1)g(x_2) \dots g(x_n), \quad (3.9)$$

звідки:

$$R(f, g) = (-1)^{mn} R(g, f) \quad (3.10)$$

Результанти знаходять застосування головним чином в теорії виключення невідомих з систем рівнянь.

3.1.5. Дискримінант.

Дискримінантом називається узятий з відомим знаком результат від полінома і його похідної :

$$R(f, f') = f'(x_1) f'(x_2) \dots f'(x_n) \quad (3.11)$$

Оскільки поліном має зі своєю похідною загальний корінь тоді і тільки тоді, якщо він має кратний корінь, то перетворення на нуль дискримінанта необхідно і достатньо для того, щоб поліном мав кратні корені.

Перетворимо вираз (6.11). Для цього звернемо увагу на те, що:

$$f'(x^i) = \lim_{x \rightarrow x_i} \frac{f(x) - f(x_i)}{x - x_i} = \lim_{x \rightarrow x_{n \setminus i}} \frac{(x - x_1)(x - x_2) \dots (x - x_n)}{(x_i - x_1) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_n)}$$

Підставляючи цей вираз у формулу (3.11), можна отримати добуток всіх різниць між коренями $x_1, x_2, \dots, x_n$, причому кожна різниця входить двічі, і при цьому з протилежними знаками. Таким чином можна отримати:

$$R(f, f') = (-1)^{\frac{n(n-1)}{2}} \cdot \prod_{i>j} (x_i - x_j)^2 \quad (3.12)$$

Вираз $\prod_{i>j} (x_i - x_j)$ залишається незмінним при парних і змінює знак при непарних підстановках. Його квадрат якраз і називатиметься дискримінантом і буде позначатися через $D(f)$, так що: $R(f, f') = (-1)^{\frac{n(n-1)}{2}} \cdot D(f)$

Якщо розглянути вираз дискримінанта від добутку двох або декількох поліномів ($F(x) = f(x) g(x)$), то:

$$\begin{aligned} D(F) &= \pm R(F, F') = \pm \prod_{i=1}^n [f(x_i) g(x_i)] \cdot \prod_{i=1}^n [f(y_i) g(y_i)] = \\ &= \pm \prod_{i=1}^n [f(x_i) g'(x_i) + f'(x_i) g(x_i)] \cdot \prod_{i=1}^n [f(y_i) g'(y_i) + f'(y_i) g(y_i)] \end{aligned}$$

Тоді в силу $f(x_i) = 0$ і $g(y_i) = 0$, можна отримати:

$$\begin{aligned} D(F) &= \pm \prod_{i=1}^n g(x_i) \cdot \prod_{i=1}^n f'(x_i) \cdot \prod_{i=1}^m f(y_i) \prod_{i=1}^m g'(y_i) = \\ &= \pm R(f, f') \cdot R(g, g') \cdot R(f, g) \cdot R(g, f) \end{aligned} \quad (3.13)$$

або

$$D(f \cdot g) = \pm D(f) D(g) R^2(f, g) \quad (3.14)$$

У цій формулі кожен з дискримінантів і результатів правої частини є ціла раціональна функція з цілими коефіцієнтами від коефіцієнтів поліномів $f(x)$ і $g(x)$. Звідси випливає, що кожен з множників правої частини є дільник числа $D(f \cdot g)$.

Формула (3.14) допускає просте узагальнення на випадок багатьох поліномів. Вважаючи $F(x) = f_1(x) f_2(x) \dots f_k(x)$, можна отримати:

$$D(f_1, f_2, \dots, f_k) = \pm \prod_{i=1}^k D(f_i) \cdot \prod_{i>j} R^2(f_i, f_j) \quad (3.15)$$

3.2. Звідні та незвідні поліноми.

Розглянемо спочатку поліноми з раціональними коефіцієнтами. Поліном називається звідним, якщо його можна представити як добуток поліномів з раціональними коефіцієнтами, кожен з яких був би нижчої степені, ніж початковий. Якщо ж це неможливо, то поліном називається незвідним.

Для знаходження методів, що дозволяють за допомогою кінцевого числа дій розрізняти звідні та незвідні поліноми, вигідно перейти до поліномів з цілими коефіцієнтами. Цього можна легко досягнути шляхом множення поліномів на спільний знаменник усіх коефіцієнтів. Крім того, зручніше мати справу з поліномами, у яких коефіцієнт при старшому членові дорівнює одиниці. Щоб перейти від довільного цілочисельного полінома

$$A_0 + A_1y + \dots + A_{n-1}y^{n-1} + A_ny^n$$

до цілочисельного полінома із одиничним старшим коефіцієнтом, треба зробити підстановку $y = \frac{x}{A_n}$, а потім помножити цей поліном на A_n^{n-1} . Отримаємо поліном типу:

$$a_0 + a_1x + \dots + a_{n-1}x^{n-1} + x^n \quad (3.16)$$

Ця рівність показує, що кожне значення $\varphi(x_1)$ шуканого полінома $\varphi(x_1)$ є дільником числа $f(x_1)$. Але оскільки кожне ціле число має лише скінченне число множників, то надається скінченне число різних припущень відносно величини $\varphi(x_1)$. Комбінуючи ці припущення для усіх $i = 0, 1, 2, \dots, n$ (зважаючи, що кожне передбачуване значення $\varphi(x_1)$ може мати або позитивний, або негативний знак, що ще збільшує число комбінацій), можна отримати деяке (практично досить велике) число гіпотез відносно значень $\varphi(x_0), \varphi(x_1), \dots, \varphi(x_n)$. Кожну з цих гіпотез необхідно перевірити, тобто: 1) користуючись формулою Лагранжа, побудувати поліном $\varphi(x)$, що дає випробовувані значення для $\varphi(x_0), \varphi(x_1), \dots, \varphi(x_n)$; 2) поліном $f(x)$ повинен ділитися на побудований нами поліном $\varphi(x)$.

Якщо виявиться, що ні при якій з гіпотез поліноми $\varphi(x)$ не задовольняють цим вимогам, то поліном $f(x)$ не зв'одиться.

Описаний метод вимагає великих викладень, оскільки число випробувань, рівне $2^{n+1} * \omega_0 * \omega_1 * \dots * \omega_n$ де ω_i - число множників числа $f(x_i)$ (включаючи само $f(x_i)$ і одиницю), може виявитися на практиці дуже великим. Тому існує багато різноманітних методів, які дозволяють за тими або іншими ознаками відразу виключати деякі з гіпотез. Для прикладу можна навести метод, що полегшує перевірку гіпотез у тому випадку, якщо як $x_0, x_1, x_2, \dots, x_n$ послідовні числа натурального ряду (або взагалі якщо вони складають арифметичну прогресію). В цьому випадку немає необхідності будувати для кожної гіпотези відповідний поліном $\varphi(x)$, а досить перекопатися, що різниці u -го порядку від послідовності $\varphi(x_0), \varphi(x_1), \dots, \varphi(x_n)$ мають постійне значення. Насправді, перші різниці полінома u -го степеня $\varphi(x)$ є значеннями полінома: $\Delta\varphi(x) = \varphi(x+1) - \varphi(x)$, степінь якого на одиницю менший. Другі різниці є значеннями полінома $(u-2)$ -ого степеня, і т. д. Нарешті різниці u -го порядку є значеннями полінома u -ого степеня, тобто постійної величини.

Для знаходження раціональних дільників полінома можна запропонувати інший метод. Якщо поліном $f(x)$ має множник u -ого степеня з раціональними коефіцієнтами, то, позначивши його корені, які є також коренями полінома $f(x)$, через $x_1, x_2, \dots, x_u$, потрібно звернути увагу на те, що усі елементарно-симетричні функції $x_1, x_2, \dots, x_u$ є раціональні числа. Нехай $\xi_1 = \psi(x_1, x_2, \dots, x_u)$ - одна з таких елементарно-симетричних функцій. Здійснюючи над величинами $x_1, x_2, \dots, x_n$ будь-які перестановки симетричної групи порядку $n!$ (тобто замінюючи їх через

$x_1, x_2, \dots, x_u, x_{u+1}, \dots, x_n$), можна отримати $s = C_n^u = \frac{n!}{(n-u)!u!}$ різних функцій $\xi_1,$

$\xi_2, \dots, \xi_n$. Коефіцієнти полінома $F(\xi) = (\xi - \xi_1)(\xi - \xi_2) \dots (\xi - \xi_n)$ є симетричними функціями від $x_1, x_2, \dots, x_n$ і тому раціонально виражаються через коефіцієнти

полінома $f(x)$. Знайшовши ці коефіцієнти, можна звести задачу до знаходження раціонального кореня полінома $F(\xi)$. Насправді, знаючи раціональні корені для кожного з рівнянь, які відповідають усім елементарно-симетричним функціям від кореня шуканого полінома, відразу отримується і самий поліном.

3.2.1 Критерій незвідності Ейзенштейна.

Використання загального методу для знаходження раціональних множників полінома на практиці є доволі складним. В зв'язку з цим було запропоновано декілька нових критеріїв, що дозволяють в деяких випадках показувати, що цей поліном є незвідним. Приведемо один з таких критеріїв, який належить Ейзенштейну (Eisenstein) :

Теорема. Якщо усі коефіцієнти полінома

$$f(x) = a_0 + a_1x + \dots + a_{n-1}x^{n-1} + x^n$$

окрім останнього $a_n = 1$, діляться на яке-небудь просте число p , але при цьому перший коефіцієнт a_0 не ділиться на p^2 , то поліном $f(x)$ незвідний:

Приклад. Розглянемо поліном

$$\frac{x^p - 1}{x - 1} = x^{p-1} + x^{p-2} + \dots + x + 1 \quad (3.17)$$

де p яке-небудь просте число. Цей поліном має корінь так званий корінь з одиниці і називається поліномом ділення круга. Щоб довести його незвідність, зробимо підстановку $x = z + 1$. Отримаємо:

$$\frac{(z+1)^p - 1}{z} = z^{p-1} + C_p^1 z^{p-2} + \dots + C_p^{p-2} z + C_p^{p-1} \quad (3.18)$$

Усі коефіцієнти $C_p^i = \frac{p(p-1)\dots(p-i+1)}{1 \cdot 2 \cdot 3 \dots i}$ ($i = 1, 2, \dots, p-1$) діляться на p , причому

останній коефіцієнт $C_p^{p-1} = p$ ділиться на p , але не на p^2 . Через це поліном (6.18) не зводиться, звідки також слідує незвідність полінома (3.17). Доведемо незвідність загальнішого полінома:

$$\frac{x^{p^{n-1}} - 1}{x^{p^{n-1}-1}} = x^{p^{n-1}(p-1)} + x^{p^{n-1}(p-2)} + \dots + x^{p^{n-1}} + 1 \quad (3.19)$$

Роблячи знову підстановку $x = z + 1$, отримаємо:

$$\frac{z^{p^{n-1}} + C_p^1 n z^{p^{n-2}} + \dots + C_{p^n}^{p^n-2} z + C_{p^n}^{p^n-1}}{z^{p^{n-1}-1} + C_{p^{n-1}}^1 z^{p^{n-1}-2} + \dots + C_{p^{n-1}}^{p^{n-1}-2} z + C_{p^{n-1}}^{p^{n-1}-1}}$$

Поліноми, що стоять в чисельнику і знаменнику цього виразу, мають коефіцієнти, що діляться на p , окрім старших коефіцієнтів. В той же час, поліном-чисельник ділиться на поліном-знаменник і, як неважко побачити з самого процесу ділення, в поліномі-частці теж усі коефіцієнти, окрім старшого, діляться на p :

$$C_{p^n}^{p^{n-1}} : C_{p^{n-1}}^{p^{n-1}-1} = p^n : p^{n-1} = p$$

Вільний член цього полінома ми отримаємо, якщо в нашій дробовій частині покладемо $z = 0$. Звідси слідує, що цей поліном задовольняє умовам критерію Ейзенштейна і тому є незвідним.

Теорема. Нехай $f(x)$ є незвідний поліном. Якщо відомо, що поліном $F(x)$ має з $f(x)$ хоча б один загальний корінь, то звідси випливає, що $F(x)$ ділиться на $f(x)$. При цьому вважаємо, що коефіцієнти обох поліномів є раціональні числа.

Наслідок. Існує один і тільки один звідний поліном з раціональними коефіцієнтами, якому задовольняє задане алгебраїчне число.

РОЗДІЛ 4

ПОЛЯ ГАЛУА

4.1. Скінченні комутативні поля (поля Галуа).

Серед простих полів характеристики p вже зустрічалися поля зі скінченною кількістю елементів. Такі скінченні поля називаються полями Галуа в честь їх першого дослідника Евариста Галуа.

Нехай Δ – поле Галуа і q – число його елементів.

Характеристика поля Δ не може бути рівною нулю, тому що інакше в Δ містилося б порожнє поле Π з характеристикою нуль, яке складається з нескінченного числа елементів. Нехай p – характеристика даного скінченного поля. Порожнє поле Π ізоморфне тоді до кільця класів лишків кільця цілих чисел по модулю p і тому містить p елементів.

Так як в Δ взагалі є лише скінченне число елементів, то в цьому полі існує найбільша система з лінійно незалежних над Π елементів $\alpha_1, \dots, \alpha_n$. Тоді n – ступінь розширення $(\Delta : \Pi)$ і кожен елемент з Δ набуває виду:

$$c_1\alpha_1 + \dots + c_n\alpha_n, \quad (4.1)$$

де коефіцієнти c_i з поля Π однозначно визначені.

Для кожного коефіцієнта $c_i \in p$ можливих значень; звідси слідує, що ми маємо точно p^n виразів виду (4.1). Оскільки ці вирази і дають елементи поля, про які йде мова, то отримуємо рівність $q = p^n$.

Отже, число елементів скінченного поля являється степенем характеристики p ; показник цього степеня дорівнює степеню розширення $(\Delta : \Pi)$.

Будь-яка структура після відкидання нуля перетворюється в деяку мультиплікативну групу. У випадку поля Галуа ця група абелева має порядок $q-1$; отже $\alpha^{q-1} = 1$ для кожного $\alpha \neq 0$.

В такому випадку рівняння $\alpha^q - \alpha = 0$ має корінь $\alpha = 0$. Звідси, всі елементи поля являються коренями многочлена $x^q - x$. Якщо $\alpha_1, \dots, \alpha_q$ – елемент поля, то $x^q - x$ ділиться на

$$\prod_i^q (x - \alpha_i).$$

В силу рівності степенів виходить що

$$x^q - x = \prod_i^q (x - \alpha_i).$$

Тому Δ складається з усіх коренів одного-єдиного многочлена $x^q - x$, що приєднуються до поля Π . Цими умовами поле Δ визначається однозначно з точністю до ізоморфізму. Звідси, при заданих числах p і n всі поля з p^n елементів ізоморфні.

Тепер потрібно показати, що для кожного $n > 0$ та для кожного p дійсно існує поле з $q = p^n$ елементів.

При розгляді простого поля Π характеристики p можна побудувати над Π поле, в якому многочлен $x^q - x$ повністю розкладається на лінійні множники. В цьому полі розглядається множина коренів многочлена $x^q - x$. Останнє являється полем тому, що з $x^{p^n} = x$ і $y^{p^n} = y$

випливає, що $(x - y)^{p^n} = x^{p^n} - y^{p^n}$, а у випадку $y \neq 0$ $\left(\frac{x}{y}\right)^{p^n} = \frac{x^{p^n}}{y^{p^n}}$.

Так що різниця і відношення двох коренів розглянутого многочлена знову являються його коренями.

Многочлен $x^q - x$ має тільки прості корені тому, що похідна, зважаючи на порівняння $q \equiv 0(p)$, дорівнює $qx^{q-1} - 1 = -1$, а -1 не є нуль. Отже, множина коренів являється множиною елементів поля з q елементів.

Отже, для кожного степеня простого числа $q = p^n (n > 0)$ існує одне, і з точністю до ізоморфізму тільки одне поле Галуа з q елементів. Ці елементи являються коренями многочлена $x^q - x$.

Поле Галуа з p^n елементів далі буде позначатися через $GF(p^n)$.

Нехай $q - 1 = h$ та відмітимо, що всі відмінні від нуля елементи поля Галуа являються коренями многочлена $x^h - 1$, тобто коренями h -ої степені з одиниці. Так як h і p взаємно прості, то для цих коренів з одиниці справджуються вищенаведені міркування.

Всі відмінні від нуля елементи поля являються степенями деякого примітивного поля h -го степені з одиниці. Це означає, що мультиплікативна група поля Галуа циклічна.

Якщо ζ – примітивний корінь h -ої степені з одиниці в $\Delta = GF(p^n)$, то всі ненульові елементи з Δ являються степенями елемента ζ . Звідси слідує, що $\Delta = \Pi(\zeta)$ і Δ являється простим розширенням поля Π . Степінь елемента ζ над Π рівна, зазвичай, степеню розширення n .

Цією теоремою будова скінченних полів описується повністю.

В подальшому знадобиться наступна

Теорема. Поле Галуа з характеристикою p містить разом з кожним своїм елементом a рівно один корінь p -го степеня з a .

Теореми справедливі для полів $GF(p^n)$, в частковому випадку $n = 1$ стають теоремами про кільце класів лишків $Z/(p)$ і співпадають з теоремами, відомими з елементарної теорії чисел. А саме:

1. Порівняння по модулю p має найбільше число коренів по модулю p таке, яке його степінь.

2. Теорема Ферма:

$$a^{p-1} \equiv 1 \pmod{p} \text{ для } a \not\equiv 0 \pmod{p}.$$

3. Існує «первісний корінь ζ по модулю p » – таке число, що будь-яке число b , взаємно просте з p , порівняне за модулем p з деяким степенем числа ζ . (Інакше кажучи: група класів лишків за модулем p , відмінних від нуля, являється циклічною.)

4. Добуток всіх відмінних від нуля елементів $a_1 \ a_2 \ \dots \ a_h$ поля $GF(p^n)$ рівне -1 , так як

$$x^h - 1 = \prod_i (x - \alpha_i).$$

Для $n = 1$ це дає теорему Вільсона: $(p-1)! \equiv -1 \pmod{p}$.

4.2. Сепарабельні і несепарабельні розширення.

Нехай Δ – поле. Потрібно в'яснити, чи може незвідний в $\Delta[x]$ многочлен мати кратні корені?

Для того, щоб $f(x)$ мав кратні корені, многочлени $f(x)$ і $f'(x)$ повинні мати відмінний від константи множник, який можна вирахувати вже в $\Delta[x]$. Якщо многочлен $f(x)$ не можна розкласти, то ні з яким многочленом меншого степеня $f(x)$ він не може мати несталих спільних множників, звідки має місце рівність $f'(x) = 0$.

Нехай:

$$f(x) = \sum_0^n a_v x^v$$

$$f'(x) = \sum_1^n v a_v x^{v-1}$$

Так як $f'(x) = 0$, в нуль може перетворюватись кожний коефіцієнт:

$$v a_v = 0 \quad (v=1, 2, \dots, n).$$

У випадку характеристики нулю звідси слідує, що $a_v = 0$ для всіх $v \neq 0$. Звідси, непостійний многочлен не може мати кратних коренів. У випадку ж характеристики p рівності $v a_v = 0$, і для $a_v \neq 0$, але тоді обов'язково повинно виконуватися порівняння

$$v \equiv 0(p).$$

Таким чином, щоб многочлен $f(x)$ володів кратними коренями всі його доданки повинні перетворюватись в нуль, за виключенням тих $a_v x^v$, для яких $v \equiv 0(p)$, тобто $f(x)$ повинен мати вигляд

$$f(x) = a_0 + a_p x^p + a_{2p} x^{2p} + \dots$$

Зворотно: якщо $f(x)$ має такий вигляд, то $f'(x) = 0$. В такому випадку ми можемо записати:

$$f(x) = \varphi(x^p).$$

Тим самим доведено твердження: у випадку характеристик нуль незвідний в $\Delta[x]$ многочлен $f(x)$ має тільки прості корені; у випадку характеристики p многочлен $f(x)$ (якщо він відмінний від константи) має кратні корені тоді і тільки тоді, коли його можна представити як многочлен φ від x^p .

В останньому випадку, може здатися, що $\varphi(x)$ в свою чергу являється многочленом від x^p . Тоді $f(x)$ являється многочленом від x^{p^2} . Нехай $f(x) = \psi(x^{p^e})$, але не являється многочленом від $x^{p^{e+1}}$. Звісно, многочлен $\psi(y)$ незвідний. Далі, $\psi'(y) \neq 0$, тому, що інакше $\psi(y)$ мав би вигляд $\chi(y^p)$, і звідси, $f(x)$ представлявся б у вигляді $\chi(x^{p^{e+1}})$, що протирічить припущенню. Звідси $\psi(y)$ має лише прості корені.

Розкладемо многочлен $\psi(y)$ в деякому розширенні основного поля на лінійні множники:

$$\psi(y) = \prod_1^m (y - \beta_i),$$

тоді:

$$f(x) = \prod_1^m (x^{p^e} - \beta_i).$$

Нехай α_i – деякий корінь многочлена $x^{p^e} - \beta_i$. Тоді

$$\alpha_i^{p^e} = \beta_i$$

$$x^{p^e} - \beta_i = x^{p^e} - \alpha_i^{p^e} = (x - \alpha_i)^{p^e}.$$

Звідси, α_i являється p^e - кратним коренем многочлена $x^{p^e} - \beta_i$ і

$$f(x) = \prod_1^m (x - \alpha_i)^{p^e}.$$

Всі корені, многочлена $f(x)$ мають, таким чином, одну і ту ж кратність p^e .

Степінь m многочлена ψ називається редукованим степенем многочлена $f(x)$ (або кореня α_1); число e називається показником многочлена $f(x)$ (або кореня α_1) над полем Δ . Між степенем, редукованим степенем та показником має місце співвідношення $n = m p^e$, де m рівне числу різних коренів многочлена $f(x)$.

Якщо θ – корінь незвідного в кільці $\Delta[x]$ многочлена, що має лише прості корені, то θ називається сепарабельним елементом над Δ або елементом першого роду над Δ . При цьому незвідний многочлен, всі корені котрого сепарабельні, називається сепарабельним. В протилежному випадку алгебраїчний елемент θ і незвідний многочлен $f(x)$ називаються несепарабельними або елементом (відповідно, многочленом) другого роду. Врешті, алгебраїчне розширення Σ , всі елементи якого сепарабельні над Δ , називається сепарабельним на Δ , а будь-яке інше алгебраїчне розширення називається несепарабельним.

У випадку характеристики нуль, згідно із сказаним вище, кожний незвідний многочлен (а тому і кожне алгебраїчне розширення), являється сепарабельним. Пізніше буде показано, що більшість найважливіших і цікавих розширень полів сепарабельні, а також існування цілих класів полів таких, які взагалі не мають несепарабельних розширень (так звані «досконалі поля»).

Розглянемо тепер алгебраїчне розширення $\Sigma = \Delta(\theta)$. Коли степінь n рівняння $f(x) = 0$, що визначає це розширення, дорівнює степеню $(\Sigma : \Delta)$, то редукована степінь m виявляється рівною числу ізоморфізмів поля Σ в такому змісті: розглянемо такі ізоморфізми $\Sigma \cong \Sigma'$, при яких елементи підполя Δ залишаються нерухомими, і відповідно, Σ переводиться в еквівалентне поле Σ' (ізоморфізм поля Σ над полем Δ) та при яких поле-образ Σ' лежить всередині деякого спільного для них поля Ω . В таких випадках має місце твердження: при підходящому виборі поля Ω розширення $\Sigma = \Delta(\theta)$ має рівно m ізоморфізмів над Δ та при будь-якому виборі поля Ω поле Σ не може мати більше m таких ізоморфізмів.

4.3. Досконалі і недосконалі поля.

Поле Δ називається досконалим, якщо будь-який незвідний в $\Delta[x]$ многочлен $f(x)$ сепарабельний. Всі інші поля називаються недосконалими.

Умови, при яких поле являється досконалим, описуються таким чином.

Теорема. Поле характеристики нуль завжди досконале.

Доведення наведено в попередньому пункті.

Теорема. Поле характеристики p являється досконалим тоді і тільки тоді, коли воно разом з кожним своїм елементом містить і корінь p -го ступеня з нього.

Доведення. Якщо разом з кожним елементом поля мати корінь p -го ступеня з нього, то кожний многочлен $f(x)$, який містить лише степені елемента x^p , являється p -им степенем, так як

$$f(x) = \sum_k a_k (x^p)^k = \sum_k \left\{ \sqrt[p]{a_k} x^k \right\}^p = \sum_k \left\{ \sqrt[p]{a_k} x^k \right\}^p.$$

Тобто кожний незвідний многочлен являється в такому випадку сепарабельним.

З іншого боку, якщо в полі є елемент α , корінь p -го ступеня, з якого в полі не міститься, то розглянемо многочлен $f(x) = x^p - \alpha$.

Нехай $\varphi(x)$ – незвідний дільник многочлена $f(x)$. Після приєднання елемента $\sqrt[p]{\alpha} = \beta$ многочлен $f(x)$ розкладається на рівні лінійні множники $(x - \beta)$, тобто $\varphi(x)$, будучи дільником $f(x)$, являє собою деякий степінь двочлена $(x - \beta)$. Якби $\varphi(x)$ був лінійним, тобто

$\varphi(x) = x - \beta$, то елемент β належав би полю Δ , що суперечить умові. Звідси, $\varphi(x) = (x - \beta)^k$ при $k > 1$ – деякий несепабельний многочлен над Δ , а тому Δ – недосконале поле. Між іншим, степінь многочлена $\varphi(x)$ обов’язково ділиться на p , а тому в цьому випадку вона просто рівна p , тобто $\varphi(x) = f(x)$.

З наведених вище теорем слідує висновок, що всі поля Галуа досконалі.

Поле Ω називається алгебраїчно замкнутим, якщо кожний многочлен з кільця $\Omega[x]$ розкладається на лінійні множники. В кожному такому полі будь-який незвідний многочлен лінійний. Отже, всі алгебраїчно замкнуті поля – досконалі.

З визначення досконалого поля одразу отримаємо дві теореми.

Теорема. Кожне алгебраїчне розширення досконалого поля сепарабельне над цим полем.

Теорема. Для будь-якого недосконалого поля існує несепабельне розширення.

Справді, ці несепабельні розширення одержуються додаванням кореня будь-якого незвідного несепабельного многочлена.

В досконалому полі характеристики p кожний многочлен $f(x)$, який залежить лише від x^p , являється p -им степенем. Дане твердження зберігає силу і для випадку многочлена від декількох змінних $f(x, y, z, \dots)$, що являється в дійсності многочленом від $x^p, y^p, z^p, \dots$. Для полів характеристики p ця властивість використовується часто.

4.4. Простота алгебраїчних розширень. Теорема про примітивний елемент.

Виясимо тепер, в яких випадках розширення Σ поля Δ буде простим, тобто отримується приєднанням примітивного елемента. З цього слідує наступна теорема про примітивний елемент, яка справедлива для досить широкого класу випадків.

Теорема. Нехай $\Delta(\alpha_1, \dots, \alpha_n)$ – кінцеве алгебраїчне розширення поля Δ і $\alpha_2, \dots, \alpha_n$ – сепарабельні елементи. Тоді $\Delta(\alpha_1, \dots, \alpha_n)$ є простим розширенням:

$$\Delta(\alpha_1, \dots, \alpha_n) = \Delta(\theta).$$

Доведення. Доведення доцільно привести спочатку для двох елементів α, β , з яких β – сепарабельний. Нехай $f(x) = 0$ – рівняння яке не розкладається для елемента α і $g(x) = 0$ – рівняння яке не розкладається для елемента β . Перейдемо до поля, в якому $f(x)$ і $g(x)$ розкладаються. Нехай $\alpha_1, \dots, \alpha_r$ – різні корені многочлена $f(x)$, а $\beta_1, \dots, \beta_s$ – корені многочлена $g(x)$. Нехай $\alpha_1 = \alpha, \beta_1 = \beta$.

Припускається, що поле $\Delta(\alpha, \beta)$ також могло бути кінцевим, а для кінцевих полів існування примітивного елемента (навіть примітивного кореня з одиниці, степенями якого є всі ненульові елементи поля) вже було доведено.

Для $k \neq 1$ має місце нерівність $\beta k \neq \beta_1$, тому рівняння

$$\alpha_i + x\beta_k = \alpha_1 + x\beta_1$$

для будь-якого i кожному $k \neq 1$ має щонайбільше один корінь x в Δ . Вибирається елемент c з відмінним від всіх коренів цих лінійних рівнянь; тоді для всіх i і $k \neq 1$, а також $\alpha_1 + c\beta k \neq \alpha_1 + c\beta_1$.

Нехай $\theta = \alpha_1 + c\beta_1 = \alpha + c\beta$, тоді θ є елементом поля $\Delta(\alpha, \beta)$. Слід показати, що θ має властивості шуканого примітивного елемента: $\Delta(\alpha, \beta) = \Delta(\theta)$. Елемент β задовольняє рівняння:

$$\begin{aligned} g(\beta) &= 0, \\ f(\theta - c\beta) &= f(\alpha) = 0. \end{aligned}$$

Коефіцієнт яких лежить в $\Delta(\theta)$. Многочлени $g(x), f(\theta - cx)$ мають загальний лише корінь β , тому що для інших коренів $\beta k (k \neq 1)$ першого рівняння маємо:

$$\theta - c\beta_k \neq \alpha_i \quad (i = 1, \dots, r)$$

і відповідно,

$$f(\theta - c\beta_k \neq 0).$$

Елемент β є простим коренем многочлену $g(x)$; відповідно $g(x)$ і $f(\theta - cx)$ мають спільним лише один лінійний множник $x - \beta$. Коефіцієнти цього найбільшого спільного дільника повинні лежати в $\Delta(\theta)$; відповідно β лежить в $\Delta(\theta)$.

Із рівняння $\alpha = \theta - c\beta$ те саме впливає для α , тобто, $\Delta(\alpha, \beta) = \Delta(\theta)$.

Тим самим теорема доведена для $h = 2$. Якщо рахувати її доведеною для $h - 1$, то $\Delta(\alpha_1, \dots, \alpha_h - 1) = \Delta(\eta)$

і відповідно,

$$\Delta(\alpha_1, \dots, \alpha_h) = \Delta(\eta, a_h) = \Delta(\theta).$$

В відповідності з вже доведеною частиною теореми, теорема справджується і для h .

Наслідок. Кожне кінцеве сепарабельне розширення є простим.

Цей наслідок спрощує вивчення кінцевих сепарабельних розширень, тому що будова і ізоморфізми цих розширень дуже легко описуються через представлення базисів

$$\sum_0^{n-1} a_k \theta^k.$$

Наприклад, доведення твердження здійснюється з допомогою послідовного продовження ізоморфізмів: кінцеве сепарабельне розширення $\sum$ поля Δ має стільки ж ізоморфізмів над Δ , яка степінь ($\sum : \Delta$). Дійсно, для простих сепарабельних розширень це твердження вже було доведено тобто, всяке кінцеве сепарабельне розширення є простим.

4.5. Норми і сліди.

Нехай $\sum$ - кінцеве розширення поля Δ або деяке кільце, яке є одночасно кінцевомірним векторним простором над Δ . Тоді елементи кільця $\sum$ можуть бути вираженими через n базисних елементів $u_1, \dots, u_n$ з коефіцієнтами із Δ $u = u_1 c_1 + \dots + u_n c_n$.

Для довільних t, u, v із $\sum$ які задовольняють співвідношенням:

$$t(u + v) = tu + tv$$

$$t(uc) = (tu)c \quad (c \in \Delta).$$

Таким чином, множення з ліва на t є лінійним перетворенням простору $\sum$ в себе. Матриця T цього лінійного перетворення в базисі $u_1, \dots, u_n$ визначається умовою $tu_k = \sum u_i t_{ik}$.

Визначник $D(T)$ цієї матриці, не залежить від вибору елементу базису, називається регулярною нормою або просто нормою елемента t в розширенні $\sum$ поля Δ :

$$N(t) = D(T) = \text{Det} \|t_{ik}\|. \quad (4.2)$$

Норму можна визначити як визначник векторів tu_k відносно базису $u_1 \dots u_n$:

$$N(t) = D(tu_1 \dots tu_n) \quad (4.3)$$

Слід $S(T)$ матриці T також не залежить від вибору базису; цей елемент основного поля називається регулярним слідом або просто слідом елемента t розширення $\sum$ над полем Δ :

$$S(t) = S(T) = \sum t_{kk}. \quad (4.4)$$

Якщо елементу t відповідає матриця T , то елементу t' – матриця T' , по добутку tt' відповідає матриця TT' , а сумі $t + t'$ – сума $T + T'$. Звідси:

$$N(tt') = N(t)N(t'), \quad (4.5)$$

$$S(t + t') = S(t) + S(t'). \quad (4.6)$$

Починаючи звідси Σ є деяким тілом, в центрі якого міститься поле Δ , тобто завжди $cu = uc$ для $c \in \Delta$, $u \in \Sigma$.

Кожний елемент t з Σ міститься в деякому комутативному тілі $\Delta(t)$ та існує мінімальний многочлен $\varphi(z) = z^m + a_1 z^{m-1} + \dots + a_m$ з властивістю $\varphi(t) = 0$. Будова простого розширення $\Delta(t)$ повністю визначається мінімальним многочленом, і звідси, норму і слід елементу t в розширенні $\Delta(t)$ можна вирахувати через коефіцієнти мінімального многочлена.

В якості базису $u_1, \dots, u_n$ розширення $\Delta(t)$ вибирається набір:

$$1, t, t^2, \dots, t^{m-1}. \quad (4.7)$$

Якщо базисні вектори помножити на t , то отримаємо набір:

$$t, t^2, t^3, \dots, t^m \quad (4.8)$$

Тепер, можна виразити вектори через базисні вектори, тоді:

$$\begin{aligned} t &= t, \\ t^2 &= t^2, \\ &\dots \\ t^{m-1} &= t^{m-1}, \\ t^m &= -a_m 1 - a_{m-1}t - a_{m-2}t^2 - \dots - a_1 t^{m-1}. \end{aligned}$$

Сума діагональних елементів матриці перетворення дорівнює $-a_1$; отже слід елемента t розширення $\Delta(t)$ рівний:

$$s(t) = -a_1. \quad (4.9)$$

Норма елемента t в розширенні $\Delta(t)$ є визначником векторів (4.8) $n(t) = D(t, t^2, \dots, t^m)$.

Якщо змінити цей визначник відповідно з правилами дій над визначниками, то:

$$n(t) = (-1)^{m-1} D(t^m, t, t^2, \dots, t^{m-1}). \quad (4.10)$$

Після цього, виразивши t^m через $1, t, \dots, t^{m-1}$:

$$t^m = -a_m 1 - a_{m-1}t - a_{m-2}t^2 - \dots - a_1 t^{m-1}. \quad (4.11)$$

Визначник з двома однаковими стовбцями рівний нулю, через це із всіх доданків в правій частині приймається до уваги лише перше. Тоді отримуємо рівність:

$$\begin{aligned} n(t) &= (-1)^{m-1} D(-a_m, 1, t^2, \dots, t^{m-1}) = \\ &= (-1)^m a_m D(1, t, t^2, \dots, t^{m-1}), \end{aligned}$$

або так як визначник з базисних векторів рівний одиниці,

$$n(t) = (-1)^m a_m. \quad (4.12)$$

Слід і норма елемента t в полі $\Delta(t)$ є, таким чином, з точністю до знака другим і останнім коефіцієнтами в мінімальному многочлені $\varphi(z)$.

В деякому підходящому образі вибраному розширенні поля $\Delta(t)$ мінімальний многочлен $\varphi(z)$ розкладається на лінійні множники:

$$\varphi(z) = (z - t_1) \dots (z - t_m) \quad (t_1 = t). \quad (4.13)$$

Тоді

$$n(t) = (-1)^m a_m = t_1 t_2 \dots t_m, \quad (4.14)$$

$$s(t) = -a_1 = t_1 + t_2 + \dots + t_m. \quad (4.15)$$

Отже, норма і слід елемента t в розширенні $\Delta(t)$ над Δ виявляються рівними добутками і сумі елементів $t_1, \dots, t_m$, сполучених з t в полі розкладу многочлена $\varphi(z)$, при чому кожен сполучений елемент t_i береться стільки разів, скільки раз відповідний множник з t_i входить в розкладення. Якщо елемент t сепарабельний над Δ , то кожен зв'язаний елемент береться один раз.

Цим методом, але тільки з деякими більш громіздкими розрахунками, отримується норма $N(t)$ і слід $S(t)$ елемента t в розширенні Σ . Якщо знову m – степінь розширення $\Delta(t)$ над Δ і g – степінь розширення Σ над $\Delta(t)$, то $n = mg$ – степінь розширення Σ над Δ . Базис розширення $\Delta(t)$ поля Δ складають степені. Нехай $v_1, \dots, v_g$ – деякий базис розширення Σ поля $\Delta(t)$. Тоді добуток $1v_1, tv_1, \dots, t^{m-1}v_1; 1v_2, \dots, 1v_g, \dots, t^{m-1}v_g$, складають деякий базис поля Σ над полем Δ . Якщо помножити базисні елементи з ліва на t і виразити добуток знову через цей базис, то сума діагональних елементів виявиться рівною

$$S(t) = (-a_1) + \dots + (-a_1) = g(-a_1)$$

або

$$S(t) = gs(t). \quad (4.16)$$

Визначник базисних елементів, перемножених на t , рівне:

$$\begin{aligned} N(t) &= D(tv_1, t^2v_1, \dots, t^mv_1; \dots; tv_g, \dots, t^mv_g) = \\ &= (-1)^{g(m-1)} D(t^mv_1, tv_1, t^2v_1, \dots; \dots; t^mv_g, tv_g, \dots, t^{t-1}v_g). \end{aligned}$$

Виразивши t^m через $1, t, \dots, t^{m-1}$ і скориставшись теоремами про визначники, тоді:

$$N(t) = (-1)^{gm} a_m^g = \{(-1)^m a_m\}^g,$$

або

$$N(t) = n(t)^g. \quad (4.17)$$

Отже, норма в розширенні Σ є g -им степенем норми в розширенні, а слід є g -кратним сліду в $\Delta(t)$.

РОЗДІЛ 5

КОДОВІ СИСТЕМИ ГАЛУА

5.1. Базис Уолша – теоретична основа кодів поля Галуа.

Отже, теоретичною основою кодів поля Галуа є теорія чисел, теорія структур, груп та полів Галуа. Одночасно теоретичні засади кодів поля Галуа базуються на системі ортогональних дискретних функцій базису Уолша, у середовищі яких існує система квазістаціонарних рекурентних функцій Галуа.

Система функцій Уолша є повною системою ортонормованих прямокутних функцій. Існують різні впорядкування такого класу базисних функцій по Уолшу, по Петлі та інш.

При впорядкуванні по частоті, яке вперше дослідив Р. Уолш множина функцій, впорядкованих таким чином, позначається:

$$Sw = \{wal_w(i, t); i \in \overline{N-1}\},$$

де: $N=2^n$; $n=1, 2, \dots$; w -ознака впорядкування по Уолшу.

Частота S_i функцій $wal_w(i, t)$ визначається умовами:

$$S_i = \begin{cases} 0, i = 1; \\ i/2, i - \text{парне}; \\ (i+1)/2, i - \text{непарне}. \end{cases}$$

Функції $cal(S_i, t)$ і $sal(S_i, t)$, які відповідають $Wal_w(i, t)$ описуються наступним чином:

$cal(S_i, t) = wal(i, t)$, i -парне;

$sal(S_i, t) = wal(i, t)$, i -непарне.

Базис Уолша при відповідному впорядкуванні формує базиси Адамара і Галуа. При цьому породжуються коди Грея, Хемінга, М-последовності, коди Баркера і коди поля Галуа.

Базис Галуа заданий системою дискретно-постійних функцій $sal(z, p, t)$ та інтервалом обмежень аргументу $[0, 2^{n-1}]$ та $[0, 2^n]$, які визначені для будь-яких натуральних чисел w і p при $0 \leq w \leq 2^n$ згідно наступного виразу:

$$G_w(Z_i) = \begin{cases} 1, Z(t_i) = Z(t_{i-1}) \oplus Z(t_{i-1}) = 1; \\ -1, Z(t_i) = Z(t_{i-1}) \oplus Z(t_{i-1}) = 0, \end{cases}$$

де w і z визначаються з двійкових розкладів:

$$w = \sum_{i=0}^{n-1} w_i \cdot 2_{n-1-i}; \quad Z = \sum_{i=0}^{n-1} Z_i \cdot 2_{n-1-i},$$

де w -індекс, а $\sum_{i=0}^{n-1} w_i$ - вага індексу, або ранг функції Галуа.

Визначимо головні властивості функцій Галуа:

1) повнота і ортогональність

Множина 2^n-1 функцій Галуа утворює повну квазіортогональну систему функцій у просторі дискретно-постійних функцій на інтервалі $[0, 2^n-1]$, причому повна ортогональність існує в просторі $(0, +2^n-1)$, тобто:

$$\phi_1(Z) = \begin{cases} 2^n - 1, \text{при } i = j; \\ -1/N, \text{при } i \neq j. \end{cases}$$

Дана система функцій Галуа ортогональна у просторі $(0, 2^n-1)$ і ортогональна у просторі $(0, -2^n-1)$, тобто при $N=2^n-1 \rightarrow \infty$, $1/N \rightarrow 0$. Такі функції Галуа породжують псевдовипадкові послідовності максимальної довжини (так звані М-сигнали) у просторі $(-1, +1)$ та відомі коди Баркера у просторі $(-1, 0, +1)$.

При цьому ортогональні властивості таких функцій Галуа які характеризуються особливими автокореляційними функціями широко використовуються у системах передавання даних для виявлення, приймання та виправлення помилок у сучасних комп'ютерних мережах та систем.

Наприклад, для 7-ми бітної рекурентної функції Галуа $\phi_1(Z_i) = (-1-1-1+1+1-1+1)$ на інтервалі $N=2^3-1$ на основі її циклічних зсувів у нормованому $(-1,+1)$ та логічному $(0, +1)$ просторах утворюються відповідні базисні матриці (рис.5.1). Графіки автокореляційних характеристик таких функцій Галуа для розглянутого прикладу розраховані на основі знакового аналітичного виразу $B_{xx}(j) = \frac{1}{N} \sum_{i=0}^{N-1} \text{sign}Z_i \otimes \text{sign}Z_{i+j}$ і приведені на рис.5.2.

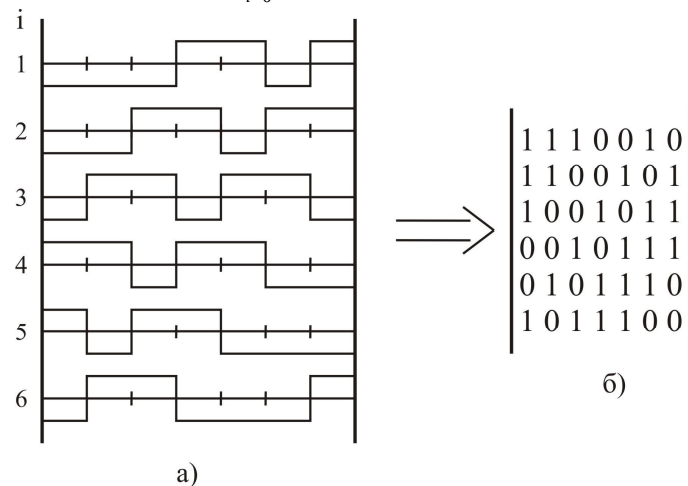


Рис.5.1. Базисні функції Галуа у нормованому (а) також логічному (б) просторах.

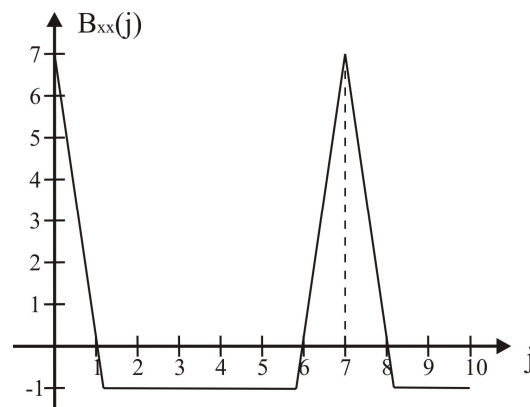


Рис. 5.2. Автокореляційна характеристика базисної функції Галуа на інтервалі 2^n-1 ($n=3$).

З рис.5.2. видно, що повнота ортогональності таких базисних функцій Галуа однозначно забезпечується у просторі $(0, +N)$, що саме і використовується у телекомунікаційних системах.

Обмеженням таких функцій Галуа є те, що вони породжують тільки унітарні коди Хаара-Галуа згідно кодової матриці (рис. 5.3), де $G_1 G_2 \dots G_i \dots G_{n-1}$ – біти коду Галуа, $G_i \in \overline{0, t}$.

$$\begin{vmatrix} G_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & G_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & G_3 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 & G_{N-1} \end{vmatrix}$$

Рис. 5.3. Кодова матриця базису Хаара-Галуа.

Множина 2^n функцій Галуа утворює повну квазіортогональну систему у просторі нормованих дискретно-постійних функцій на інтервалі $[0, 2^n]$, при чому повна ортогональність існує у просторі $(0, +2^n)$, тобто:

$$\phi_2(Z) = \begin{cases} 2^n, \text{ при } i = j; \\ -1/N_i, \text{ при } i \neq j, \end{cases}$$

де значення N_i залежить від інтервалу 2^n та незвідного поліному, на основі якого генерується функція Галуа.

Слід зауважити, що функції $\phi_1(Z)$ та $\phi_2(Z)$ можуть генеруватись одним і тим самим ключем незвідного полінома поля Галуа. При цьому $\phi_2(Z)$ буде відрізнятись від $\phi_1(Z)$ виконанням так званої операції біт-стафінгу (вставки символу «0») шляхом розширення інтервалу 2^n-1 функції $\phi_1(Z)$ до інтервалу 2^n у діапазоні максимальної кількості її одиничних значень.

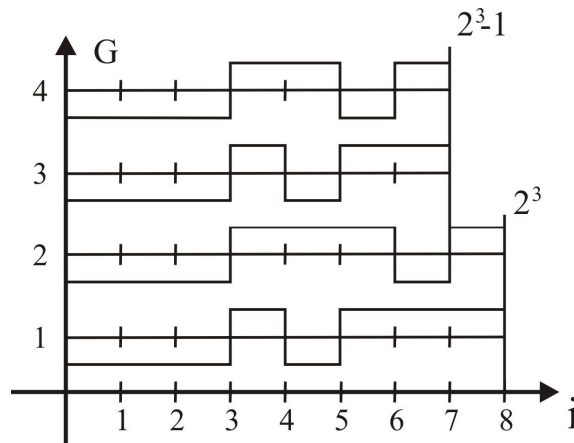


Рис. 5.4. Приклад розширення інтервалу функцій Галуа.

Наприклад, семибітні функції Галуа $\phi_1(Z)$ і $\phi_1(Z)$, які генеруються відповідно різними ключами $G_{i+1}=G_i \oplus G_{i-3}$ та $G_{i+1}=G_i \oplus G_{i-2}$ на інтервалі 2^3-1 розширюються до однієї функції $\phi_2(Z)$ на інтервалі 2^3 (рис.5.4.)

При цьому породжується система квазіортогональних функцій Галуа у нормованому та логічному просторах, матриці яких показані на рис. 5.5.

Автокореляційна характеристика функцій $\phi_2(Z)$ показана на рис.5.6, звідки видно, що система таких базисних функцій також ортогональна у просторі $(0, +2^n)$.

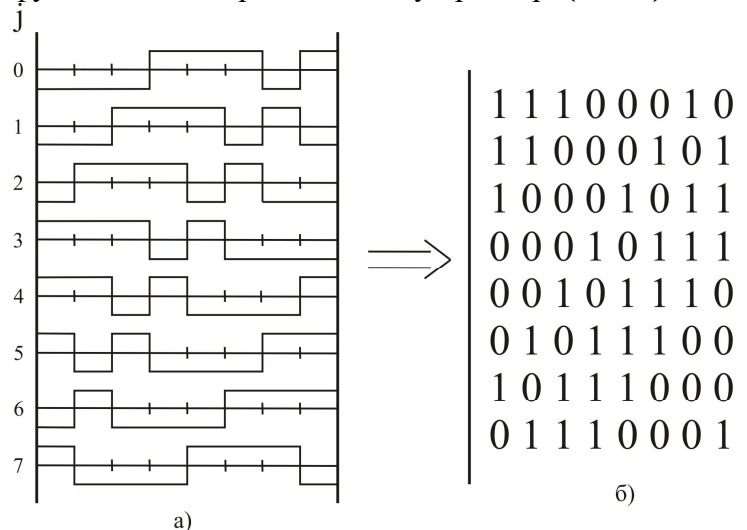


Рис. 5.5. Базисні функції Галуа у нормованому (а) та логічному (б) просторах.

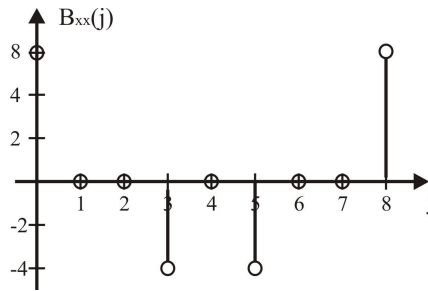


Рис.5.6. Автокореляційна характеристика базисної функції Галуа на інтервалі 2^n ($n=3$).

Можна показати, що від'ємні пелюстки автокореляційної характеристики базисної функції Галуа асимптотично зменшуються при розширенні інтервалу 2^n .

Наприклад, код функції Галуа $GF(4/2)$ (1111010110010000) має автокореляційну характеристику приведену на рис. 5.7.

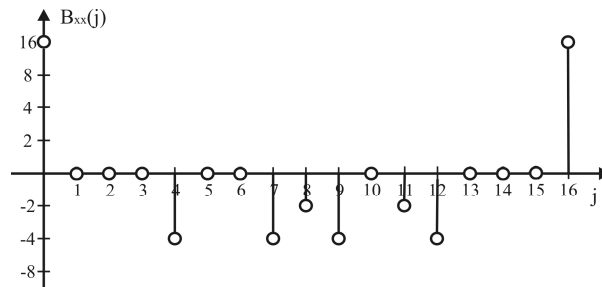


Рис. 5.7. Автокореляційна характеристика базисної функції Галуа на інтервалі 2^3 .

У загальному випадку функція $\phi(Z)$, яка описує початкову систему P -ічних функцій алгебри логіки від n -аргументів, є дискретно-постійною на інтервалі $[0, p^n-1]$. Якщо функція $\phi(Z)$ така, що:

$$\sum_{Z=0}^{p^n-1} \phi(Z) \cdot G_w(Z) = 0; \quad \omega \in 0, p^n-1, \quad \text{то } \phi(Z)=0.$$

Таким чином, функції Галуа є базисом асимптотично ортогональних рядів дискретно-постійних функцій, які описують системи булевих функцій.

Кінцевість ряду, який описує функцію Галуа визначається виходячи з виразів:

$$\phi(Z) = \sum_{\omega=0}^{2^n-1} S(\omega) \cdot G_{\omega}(Z), \quad (5.1.)$$

$$S(\omega) = 2^{-n} \sum_{Z=0}^{2^n-1} \phi(Z) \cdot G_{\omega}(Z) \quad (5.2.)$$

де вираз (5.1.) розклад функцій в ряд Фур'є, а (5.2.) - коефіцієнтами Фур'є.

З (5.1.) слідує, що розклад в ряд по функціях Галуа будь-якої дискретно-постійної функції, що описує деяку систему булевих функцій від n -аргументів, містить не більше 2^n -х членів, а вага індексів ω функцій Галуа у розкладі не перевищує n .

Симетрія індексу і аргументу функції Галуа визначається згідно аксіоми: для будь-яких $\omega, Z, \in 0, 2^n$ справедливо $G_{\omega}(Z) = G_Z(\omega)$, що слідує з (5.1).

Властивість зсуву аргументу і рекурентність функцій Галуа визначається наступним чином: для будь-яких $\omega, Z, \tau \in 0, 2^n$ тоді $G_{\omega}(Z + \tau) = G_{\omega}(Z) \cdot G_{\omega}(\tau) \bmod 2$, отже, множина функцій Галуа замкнена у полі відносно операції множення.

Властивість рекурентності є особливо важливою у сфері застосування функцій базису Галуа, а також кодів та систем числення в полях Галуа при рішенні широкого класу задач

теорії чисел, кодування, перетворення та цифрового опрацювання сигналів, стиснення даних, їх захисту від помилок та несанкціонованого доступу шляхом шифрування.

Використання властивості рекурентності аргументу базису Галуа в наш час склало фундаментальну основу розробки теорії та принципово нових технічних рішень багатьох складних задач і практичних застосувань у галузі цифрового формування, передавання і опрацювання інформаційних потоків, в тому числі:

- 1) побудову
 - кодових систем Галуа;
 - генераторів кодів поля Галуа;
 - кодових шкал Галуа;
 - АЦП Галуа;
 - Багатопортової асоціативної пам'яті Галуа;
- 2) розробки нових методів:
 - передавання інформації в умовах інтенсивних завад;
 - формування та передавання інформації на основі вертикальної інформаційної технології;
 - стиснення даних у базисі Галуа;
 - блокової синхронізації потоків даних на основі протоколів Галуа;
 - виявлення та виправлення помилок в потоках даних сигнальними коректуючими кодами Галуа;
 - рандомізації гармонічних сигналів в кодах Галуа;
 - ідентифікації передаварійних та аварійних станів квазістаціонарних об'єктів управління;
 - структуризації інформаційних даних в комп'ютеризованих мережах та системах.

5.2. Класи кодів поля Галуа (КПГ).

5.2.1 Одновимірні дворівневі КПГ.

Принципи кодування методом залишків базуються на теоретико-числовому підході і представляють собою клас векторних багаторівневих кодів поля Галуа. Застосування цього класу КПГ дозволяє зменшити надлишковість вимірювальної інформації за рахунок виключення з інформаційного потоку кодів рангів Діофантового рівняння.

$$Y_i = b_i \pmod{P},$$

яке відповідає лінійному рівнянню з розв'язком у цілих числах

$$Y_i = a_i \cdot P + b_i, \quad (5.3)$$

де $a_i = \tilde{E} \left[\frac{Y_i}{P} \right]$ - ранг цифрового відліку Y_i ; $0 \leq b_i \leq P-1$ - найменший невід'ємний

залишок Y_i по модулю P , тобто

$$b_i = \text{rez} Y_i \pmod{P}.$$

З виразу (5.3) видно, що величина Y_i може бути однозначно відновлена, якщо крім b_i можна обчислити ранг a_i на основі попередньо відомого Y_{i-1} . Тобто існує рішення функціонеру:

$$Y_i = F(Y_{i-1}, b_i).$$

Покажемо, що достатньо простим при цьому є рекурентний алгоритм обчислення рангу a_i по текучому значенню b_i і відомому рангу a_{i-1} попереднього відліку Y_{i-1} .

Для доведення існування функціонала Y_i вирішимо рівняння (5.3) відносно a_i :

$$a_i = \frac{Y_i - b_i}{P},$$

Враховуючи цілочисельний характер рангу a_i , введемо праву частину останнього виразу під знак цілочисельної функції:

$$a_i = E \left[\frac{Y_i - b_i}{P} \right]$$

Виразимо значення відліку Y_i через результат попереднього вимірювання Y_{i-1} :

$$Y_i = Y_{i-1} + \alpha P \quad (5.5)$$

де α – деяка величина, які визначає відмінність відліків Y_i та Y_{i-1} .

Підставивши (5.5) у a_i і виконавши прості перетворення отримаємо:

$$a_i = E \left[\frac{Y_{i-1} - b_i}{P} + \alpha \right], \quad (5.6)$$

де α має зміст помилки обчислення рангу відліку Y_i значення Y_{i-1} .

Ранг відліку Y_i з рівняння (5.6) може бути точно визначений виходячи з властивостей цілочисельної функції $E[\cdot]$, якщо величина помилки його обчислення менша одиниці, тобто $0 \leq \alpha < 1$.

Доведемо ствердність формули (5.6) при умові, що зміна рангу a_{i-1} та a_i не перевищує одиниці:

$$(a_i - a_{i-1}) \leq 1 \quad (5.7)$$

На рис.5.8 показані варіанти зміни цифрових відліків Y_i та Y_{i-1} при виконанні умови (5.7).

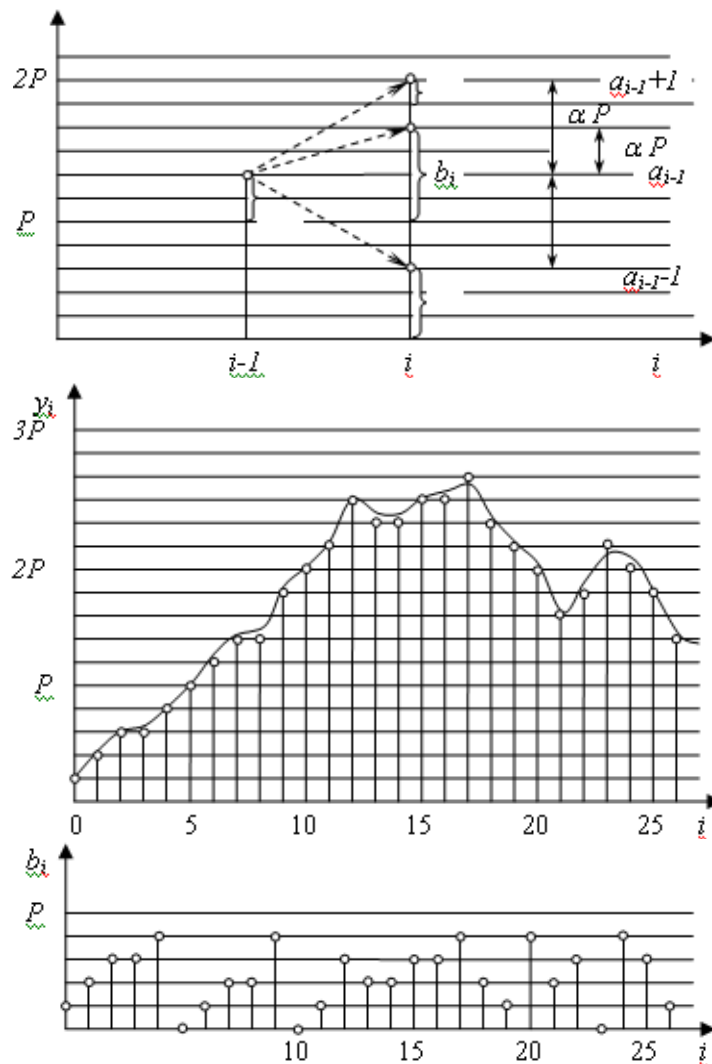


Рис.5.8. Рекурентне кодування інформаційних потоків методом залишків.

Нехай відлік Y_{i-1} має ранг a_{i-1} і залишок b_{i-1} .

Згідно рис.5.8 можливі три випадки отримання відліків, які відрізняються значенням рангів a_i від рангу a_i відліку Y_{i-1} :

1) Ранг Y_i збільшується на одиницю ($a_i = a_{i-1} + 1$).

Очевидно, що в цьому випадку приріст відліку Y_i не може бути менший $P - b_{i-1}$, а також не перевищує величини αP , тобто:

$$P - b_{i-1} \leq P + b_{i-1} - b_i \leq \alpha P,$$

або підсилюючи нерівність

$$P - \alpha P \leq b_{i-1} - b_i \leq b_{i-1}. \quad (5.8)$$

Підставивши Y_{i-1} у рівняння (5.6) і потім у (8.4), отримаємо рішення для шуканого Y_i

$$Y_i = E \left[a_{i-1} + \frac{b_{i-1} - b_i}{P} + \alpha \right] P + b_i. \quad (5.9)$$

З співвідношень (5.8) та (5.9) витікає оцінка для текучого відліку у вигляді нерівності

$$E \left[a_{i-1} + \frac{P - \alpha P}{P} + \alpha \right] P + b_i \leq Y_i \leq E \left[a_{i-1} + \frac{b_{i-1}}{P} + \alpha \right] P + b_i,$$

звідси враховуючи, що $b_{i-1} < P$, отримаємо

$$E[a_{i-1} + 1]P + b_i \leq Y_i \leq E[a_{i-1} + 1 + \alpha]P + b_i \quad (5.10)$$

Верхня границя у (5.10) співпадає з нижньою, оскільки

$$\alpha < 1 \text{ і } E[a_{i-1} + \alpha + 1] = [a_{i-1} + 1].$$

Тому

$$Y_i = E[a_{i-1} + 1]P + b_i$$

і справедливості формули (5.6) доведена для випадку 1).

2) ранг по відношенню до Y_{i-1} зберігається ($a_i = a_{i-1}$).

Використовуючи рівняння (5.9) та нерівність

$$-\alpha P \leq b_{i-1} - b_i \leq \alpha P,$$

які слідують з рис.8.8, оцінимо, як і у випадку п.1) значення Y_i

$$E \left[a_{i-1} - \frac{\alpha P}{P} + \alpha \right] P + b_i \leq Y_i \leq E \left[a_{i-1} - \frac{\alpha P}{P} + \alpha \right] P + b_i,$$

звідки

$$E[a_{i-1}]P + b_i \leq Y_i \leq E[a_{i-1} + 2\alpha]P + b_i. \quad (5.11)$$

Границі в (8.11) співпадають, якщо

$$2\alpha = \frac{2(b_{i-1} - b_i)}{P} < 1 \text{ або } \frac{b_{i-1} - b_i}{P} < 0.5 \quad (5.12)$$

і формула (5.6) доведена для випадку 2).

3) Ранг Y_i зменшується на одиницю ($a_i = a_{i-1} - 1$).

Приріст відліку Y_i згідно рис.5.8 у цьому випадку не менше залишку b_{i-1} і не перевищує αP .

Оцінюючи, як і у попередніх випадках значення Y_i

$$E \left[a_{i-1} - \frac{b_{i-1} - P}{P} + \alpha \right] P + b_i \leq Y_i \leq E \left[a_{i-1} - \frac{2P - P}{P} + \alpha \right] P + b_i$$

і підсилюючи оцінку нерівності знизу $b_{i-1} \geq -\alpha P$, отримаємо

$$E[a_{i-1} - 1]P + b_i \leq Y_i \leq E[a_{i-1} - 1 + 2]P + b_i. \quad (5.13)$$

Підставивши помилку обчислення рангу α в умову (5.12), добиваємося співпадання границь у (5.13). Тим самим доведення формули (5.6) завершено.

З викладеного доведення, видно, що умова існування алгоритму обчислення рангу відліку Y_i на основі Y_{i-1} та b_i по формулі (5.6) виконується, коли для двох послідовних відліків справедлива нерівність

$$-0.5 < \frac{b_{i-1} - b_i}{P} < 0.5 \quad (5.14)$$

Остання умова також обмежує допустиму величину першої різниці між відліками Y_{i-1} та Y_i

$$|Y_i - Y_{i-1}| < 0.5P, \quad (5.15)$$

що добре узгоджується другим випадком, коли залишки b_i та b_{i-1} належать одному рангу $a_i = a_{i-1}$.

Таким чином, рівняння (5.9) задовольняє вимоги однозначного декодування для всіх розглянутих випадків, якщо $\alpha = 0.5$. При цьому у відповідності з (5.6) помилка обчислення рангу a_i завжди буде достатньою.

Враховуючи встановлений знак помилки α у виразах (5.6) та (5.9), отримаємо шукану рекурентну формулу для відновлення потоку вимірювальних даних, які кодуються методом залишків багаторівневим векторним кодом поля Галуа

$$Y_i = \hat{E} \left[\frac{Y_{i-1} + b_i}{P} + 0.5 \right] P + b_i \quad (5.16)$$

де $\hat{E}[\cdot]$ - цілочисельна функція з округленням до більшого цілого.

Розглянемо метод кодування даних методом залишків на прикладі рис.5.8.

Кодування ведеться по модулю $P=5$, тому найбільший перший приріст відліків $|Y_i - Y_{i-1}| \leq 2$ згідно умови (5.15).

Нехай $i=8$, тоді: $Y_{i-1}=7$; $b_i=2$; $b_{i+1}=4$; $b_{i+2}=0$; $b_{i+3}=1$. Згідно (5.16) маємо:

$$Y_i = \hat{E} \left[\frac{7-2}{5} + 0.5 \right] 5 + 2 = \hat{E}[1.5] 5 + 2 = 7;$$

$$Y_{i+1} = \hat{E} \left[\frac{7-4}{5} + 0.5 \right] 5 + 4 = \hat{E}[1.1] 5 + 4 = 9;$$

$$Y_{i+1} = \hat{E} \left[\frac{9-0}{5} + 0.5 \right] 5 + 0 = \hat{E}[2.3] 5 + 0 = 10;$$

$$Y_{i+1} = \hat{E} \left[\frac{10-1}{5} + 0.5 \right] 5 + 1 = \hat{E}[2.3] 5 + 1 = 11.$$

Таким чином, доведені умови та продемонстрована однозначність кодування та декодування інформаційних потоків багаторівневими векторними кодами поля Галуа.

Найширше застосування такі коди отримали у алгоритмах стиснення інформації та спектрального аналізу сигналів на основі нелінійних цифрових фільтрів.

5.2.2. Двовимірні дворівневі КПП.

Двовимірні дворівневі КПП можуть бути задані у вигляді матриць з числом елементів $V = 2^k \cdot 2^m$, які приймають пари бінарних значень $b_{ij} \in \{00, 01, 10, 11\}$; $i \in \overline{1, 2^k}$; $j \in \overline{1, 2^m}$

$$b_{ij} = \begin{vmatrix} b_{k1} & b_{k2} & \dots & b_{kj} & \dots & b_{km} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ b_{i1} & b_{i2} & \dots & b_{ij} & \dots & b_{im} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ b_{21} & b_{22} & \dots & b_{2j} & \dots & b_{2m} \\ b_{11} & b_{12} & \dots & b_{1j} & \dots & b_{1m} \end{vmatrix} \quad (5.17);$$

Приклад двохмірного коду Галуа GG_2^3 ілюструється бульовою матрицею

$$GG^{(3)}_{(2)} = \begin{pmatrix} 01 & 01 & 01 & 00 & 01 & 00 & 00 & 00 \\ 01 & 01 & 01 & 00 & 01 & 00 & 00 & 00 \\ 01 & 01 & 01 & 00 & 01 & 00 & 00 & 00 \\ 11 & 11 & 11 & 10 & 11 & 10 & 10 & 10 \\ 01 & 01 & 01 & 00 & 01 & 00 & 00 & 00 \\ 11 & 11 & 11 & 10 & 11 & 10 & 10 & 10 \\ 11 & 11 & 11 & 10 & 11 & 10 & 10 & 10 \\ 11 & 11 & 11 & 10 & 11 & 10 & 10 & 10 \end{pmatrix}$$

Таке подання двомірного дворівневого КПП (2D КПП) визначає теорію і методологію кодування інформації у двомірному дискретному просторі. При цьому нове рішення отримують у кратному випадку дві задачі:

1) прив'язка координат і позиціонування об'єкта в мережі двомірного простору з двох бітовими мітками Галуа. При цьому у залежності від геометрії площі позиціонування по вертикалі і горизонталі можуть бути вибрані різні 1D КПП (рис.5.9).

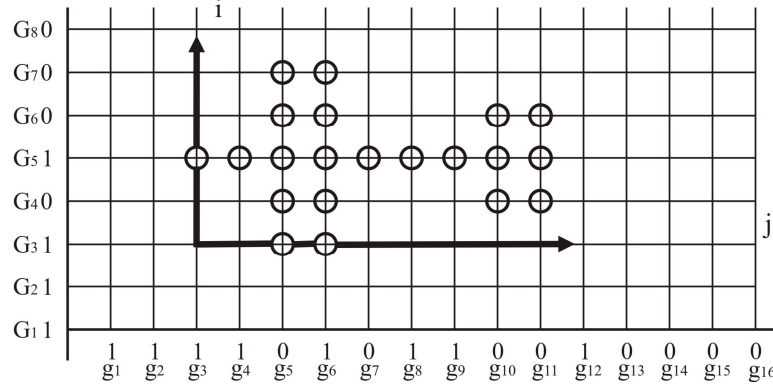


Рис.5.9. Позиціонування об'єкта в координатах 2D коду поля Галуа.

Згідно рис.5.9 однозначне позиціонування об'єкта в координатах 2D КПП описується наступним кодом базисних функцій нульового порядку

$$F(ij) = \begin{cases} G_7 : g_4 g_5 \overline{g_6} \overline{g_7} g_8 g_9 g_{10} g_{11} g_{12} \\ G_6 : g_4 g_5 \overline{g_6} \overline{g_7} g_8 g_9 g_{10} \overline{g_{11}} \overline{g_{12}} \\ G_5 : \overline{g_4} \overline{g_5} \overline{g_6} \overline{g_7} \overline{g_8} \overline{g_9} \overline{g_{10}} \overline{g_{11}} g_{12} \\ G_4 : g_4 g_5 \overline{g_6} \overline{g_7} g_8 g_9 g_{10} \overline{g_{11}} \overline{g_{12}} \\ G_3 : g_4 g_5 \overline{g_6} \overline{g_7} g_8 g_9 g_{10} g_{11} g_{12} \end{cases} \quad (5.18)$$

Вираз (5.18) подамо у вигляді мулевих значень $G_i g_j$:

$$F(ij) = \begin{cases} 0:100111001 \\ 0:100111010 \\ 1:011000110 \\ 0:100111010 \\ 1:100111001 \end{cases}$$

Аналіз такого способу застосування 2D КПП показує, що при стандартному кодуванні об'єкта у базисі Радемахера потрібно код з ентропією:

$$I_x = (\log_2 k + \log_2 m) \cdot V,$$

де V -число точок ідентифікації координат об'єкта .

Для розглянутого прикладу:

$$I_x(R) = (3 + 4) \cdot 21 = 147 \text{ біт};$$

$$I_x(G) = 5 \cdot 10 = 50 \text{ біт},$$

Що практично у три рази менше у базисі Галуа по відношенню до базису Радемахера.

Оцінка зменшення об'єму інформації при кодуванні зображень у базисі Галуа аналітично описується наступним виразом:

$$k_{\text{еф}} = \frac{V(\log_2 n + \log_2 m) \cdot \log_2 a}{n_j \log_2 a \cdot m_j} \quad (5.19)$$

де n_j, m_j – відповідно число інформаційних елементів 2D КППГ, в паралелепіпеді яких вміщений кодований об'єкт; a – число рівнів квантування елементів півтонового зображення. З виразу (5.19) очевидно, що ефективність кодування зображень у базисі Галуа не залежить від значення a .

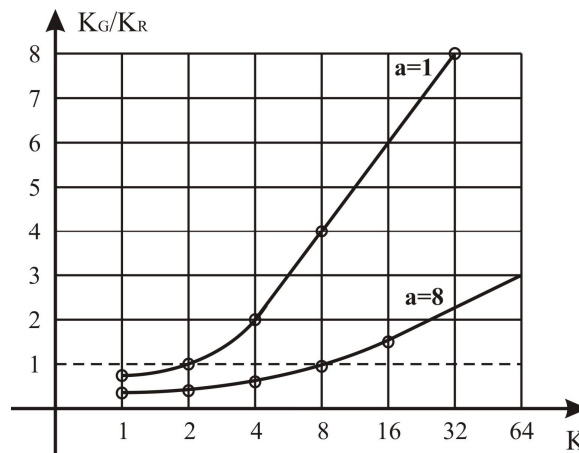


Рис.5.10. Відносна ефективність кодування зображень на основі 2D КППГ.

На рис.5.10 показані графіки відносної ефективності кодування дворівневих зображень.

5.2.3. Просторова форма двовимірних КППГ.

Просторова форма КППГ на основі розгалуженого кодового графа.

Така форма КППГ описується математичною моделлю:

$$N_j = (b_0 \ b_1 \dots b_j);$$

$$b_j = \text{res}(b_{j-1} + b_{j+n}) \bmod P. \quad (5.20)$$

Це подання КППГ визначає методологію побудови транспортних систем і мережу маршрутизації мобільних об'єктів на площині. При цьому реалізація «м'якого» позиціонування зводиться до розміщення вздовж розгалужених транспортних шляхів сигнальних міток КППГ, маніпульованих згідно деяких двох ознак: символів, кольору, геометричному розмірі, електричному, оптичному, магнітному та інш. параметру чи ефекту.

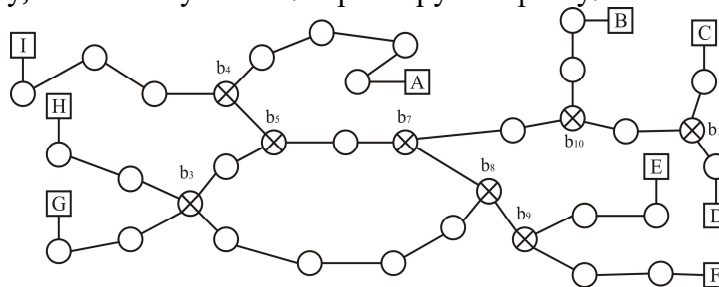


Рис.5.11. Структура транспортної мережі руху мобільного об'єкта в двох бітовому КППГ.

На рис.5.11 показаний приклад розгалуженої структури транспортної мережі, в якій елементи позиціонування утворює КППГ кільця $GF\left(\begin{smallmatrix} 4 \\ 2 \end{smallmatrix}\right) = (1111010110010000)$.

У процесі переміщення об'єкта послідовно зчитується сигнальні мітки двох бітового КППГ. Причому n – розрядний кодон визначає місцезнаходження об'єкта на траєкторії руху. Дії об'єкта руху, повертання вправо або вліво визначається виходячи з текучого кодона КППГ і програми руху, яку об'єкт отримує на початковому або кінцевому пункті.

Для однозначного позиціонування місцезнаходження і програмного переміщення рухомого об'єкту між різними пунктами А, В, С, D, E, F, G, H, L сила дається програма, яка в операторній формі має вигляд:

$$S_1 : (b, \dots b_i) \equiv V_{1j}(b_i, \dots, b_j) \equiv V_{ij} \dots (b_m, \dots, b_k) \equiv S_2,$$

де: S_1, S_2 – відповідні символи (коди) початкового та кінцевого пунктів переміщень, V_j – ознака повороту:

$$V_j = \begin{cases} 0, \uparrow (\text{вліво}) \\ 1, \downarrow (\text{направо}) \end{cases},$$

Наприклад програми можливих шляхів переміщень рухомого об'єкта (РО) з пункту Н в пункт Е має наступний вид:

$$\begin{array}{ll} 1) H : b_1 b_2 b_3 \equiv 0; & 2) H : b_1 b_2 b_3 \equiv 1; \\ b_3 b_4 b_5 \equiv 1; & b_3 b_4 b_5 b_6 b_7 b_8 \equiv 1; \\ b_5 b_6 b_7 \equiv 1; & b_6 b_9 \equiv 1; \\ b_7 b_8 \equiv 0; & b_9 b_{10} b_{11} \equiv 0; \\ b_8 b_9 \equiv 0; & \\ b_9 b_{10} b_4 \equiv 0 & \end{array}$$

Описана форма двох бітового КППГ не є реверсивною, тому транспортна мережа на її основі характеризується одно напрямленістю, тобто переміщення РО можливі тільки тільки з пунктів А, G, H, L в один з інших В, С, D, E, F.

Для опису програмного руху РО у зворотньому напрямі необхідно подати модель (М) в інвертованому виді:

$$N_j = (b_j, b_{j-1} \dots b_0)$$

$$b_{j-1} = \text{res}(b_j + b_{j+n}) \bmod P,$$

А у програму руху ввести ознаку зворотнього руху.

Тоді:

$$V_j = \begin{cases} S_0, \rightarrow; \\ 0, \uparrow; \\ S_1, \leftarrow; \\ 1, \downarrow. \end{cases}$$

При цьому для правильного зчитування КППГ у зворотньому напрямі РО повинен переміщуватися на $n-1$ позицію даліше мітки початку зворотнього руху або завантажитись зворотнім ключем коду Галуа.

Приклад програми руху РО з пункту А в пункт Н має вигляд:

$$\begin{aligned} A : b_m b_1 b_2 b_3 b_4 &\equiv 0; \\ b_4 b_5 &\equiv 1, \\ b_5 b_6 b_7 b_8 &\equiv S; \\ b_8 b_7 &\equiv 1; \\ b_7 b_6 b_5 &\equiv 0; \\ b_5 b_4 b_3 &\equiv 1; \\ b_3 b_2 b_1 &\equiv H; \end{aligned}$$

Очевидно, що розширивши число ознак V_i наприклад:

$$V_i = \begin{cases} S_0, \rightarrow; \\ 0, \uparrow; \\ \mathbf{1}, \downarrow \\ S_1, \lrcorner \\ S_2, \sqsupset \\ S_3, \leftarrow \end{cases}$$

Де символи $\uparrow, \downarrow$ - відповідно визначають рух вгору або вниз, можна побудувати багаторівневу 3D мережу позиціонування переміщень РО в трьохмірному просторі.

5.2.4. Полярно-спіральна форма КПГ.

Полярно-спіральна форма КПГ у вигляді двомірної розгортки описується матрицею:

$$N_{ij} = \begin{vmatrix} b_{11} & b_{12} & \dots & b_{1j} & \dots & b_{1m} \\ b_{21} & b_{22} & \dots & b_{2j} & \dots & b_{2m} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ b_{i1} & b_{i2} & \dots & b_{ij} & \dots & b_{im} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & \dots & b_{nj} & \dots & b_{nm} \end{vmatrix}, \quad (5.21)$$

де:

$$\begin{aligned} b_{ij} &= \text{res}(b_{ij} + b_{i,j+m}) \bmod \rho \\ b_{ij} &= \text{res}(b_{i-1,j} + b_{i+n,j}) \bmod P \end{aligned}$$

Для КПГ $GF(2^4)$ матриця (5.21) має вигляд:

$N_{ij} =$

Особливістю такого подання КПП є одночасне існування рекурентності по спіралі та твірній циліндра. Дана властивість спіральних форм КПП ефективно використана в сигнальних коректуючи кодах Галуа для виявлення та виправлення пачок помилок в каналах зв'язку сучасних безпроводних сенсорних та комп'ютерних мережах.

Нехай маємо код Галуа G_2^4 який формується рекурентним кодом:

$$G_{i+1} = G_i \oplus G_{i-4}, \quad (5.22)$$

ТОБТО:

$$\begin{aligned} G_2^4 &= 1111\ 01011\ 0010\ 0011\ 1101\ 0110 \\ &0100\ 0111\ 1010\ 1100\ 1000\ 1111\ 0101\ \dots \end{aligned} \quad (5.23)$$

Можна показати, що код генерований на основі виразу (5.23) можна упакувати в спіраль (рис.5.12), причому по кожній з чотирьох твірних формується рекурентна послідовність, яка має відповідні рекурентні властивості коду в базисі Галуа $G_{i+1} = G_i \oplus G_{i-4}$.

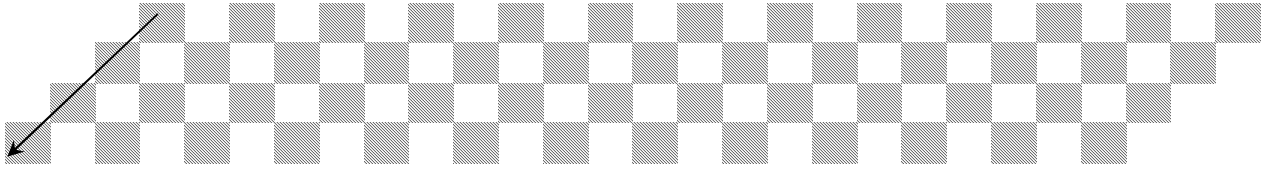


Рис. 5.12. Сигнальний коректуючий код упакований у вигляді спіралі.

Але коли дану спіраль закодовану рекурентним кодом розкрутити, то рекурентність збережеться через 12 символів і можливо буде виправляти помилки по твірних спіралі згідно виразів:

$$G_{i+1} = G_i \oplus G_{i-13}. \quad (5.24)$$

Виправлення помилок в сигнальних кодах з врахуванням спіральних властивостей відбувається наступним чином: з врахуванням спіральних властивостей сигнальний рекурентний код можна використати при виявленні пакетів помилок. Оскільки спочатку виконується перевірка і виправлення помилок згідно виразу (5.22), а потім, згідно виразу (5.24) - по твірних спіралі.

На рис.5.13 показано виявлення помилок при використанні спіральних властивостей сигнальних коректуючих кодів поля Галуа. Як видно з рис.5.13, при виникненні помилок в п'яти підряд позиціях, виявлення помилок завдяки рекурсивному виразу (5.23) стає неможливим, оскільки можлива неправильна корекція. Тому потрібно виявляти дані помилки завдяки спіральним властивостям коду Галуа, тобто виразу (5.24).

Номер позиції бітів	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	...
G_4^2	1	1	1	1	0	1	0	1	1	0	0	1	0	0	0	1	1	1	1	0	1	0	...
G_{i-4}										*	*	*	*	*									
G_{i-13}												*											

Рис. 5.13. Виявлення помилок при використанні спіральних властивостей сигнальних коректуючих кодів поля Галуа.

5.2.5. Двомірний однобітовий КПГ.

Даний клас кодів Галуа формується на основі квадратної матриці з першою стороною і першим стовпцем у вигляді лінійних рекурентних послідовностей з ключем

$$G_{i+1} = a_i G_i \oplus a_{i-1} G_{i-1} \oplus \dots \oplus a_{i-n} G_{i-n},$$

де $a_i \in \overline{0,1}$ для всіх бітів Галуа стартового коду $(G_i, G_{i-1}, \dots, G_{i-n})$.

У загальному випадку матриця коду Галуа даного класу має вигляд:

$$\begin{vmatrix} G_i & G_{i-1} & \dots & G_j & \dots & G_{i-n} \\ G_{i-1} & G_{i-2} & \dots & G_{j-1} & \dots & G_i \\ G_{i-2} & G_{i-3} & \dots & G_{j-2} & \dots & G_{i-1} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ G_{i-n} & G_i & \dots & G_{j+1} & \dots & G_{i-n+1} \end{vmatrix}$$

Наприклад: для поля Галуа $GF(2^3)$ з ключами: $G_{i+1} = G_i \oplus G_{i-3}$ та $G_{i+1} = G_i \oplus G_{i-2}$ - відповідно отримуємо матриці двомірних кодів Галуа даного класу:

$$\begin{vmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{vmatrix}; \quad \begin{vmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \end{vmatrix}$$

Аналіз приведених двомірних кодів Галуа показує, що вони характеризуються симетрією, як видно з рис.5.14 і потужними можливостями виявлення та виправлення помилок.

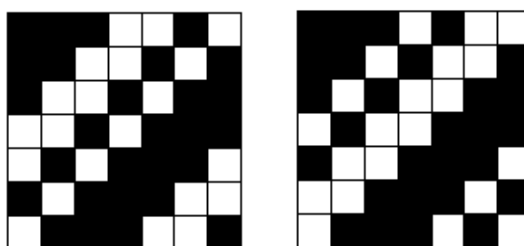


Рис 5.14. Симетрія двомірних кодів поля Галуа.

Їх застосування у системах передавання даних в умовах впливу інтенсивних завад або значному затуханні сигналів у каналах зв'язку.

РОЗДІЛ 6

ТЕОРЕТИКО-ЧИСЛОВІ БАЗИСИ (ТЧБ)

6.1. Характеристика кодових матриць ТЧБ

Теоретико-числові базиси широко застосовується для формування, передавання цифрового опрацювання та зберігання даних. Застосування різних ТЧБ при організації БД створює перспективи вдосконалення організації баз даних. Зменшення об'ємів ідентифікаційних кодів доступу до даних, стиснення інформації та інше.

Вибір базисної функції виконується в залежності від системних характеристик різних каналів зв'язку та умов експлуатації комп'ютерних та телекомунікаційних систем. В сучасних комп'ютерних та телекомунікаційних системах широко використовуються ТЧБ на основі кусково-постійних дискретних функцій, які забезпечують значно простішу реалізацію цифрових генераторів, а також спрощують алгоритми цифрового опрацювання сигналів.

6.2. Системні характеристики СПД

Системні характеристики СПД, в першу чергу, визначаються способами формування, перетворення та кодування даних. Згідно аналізу застосування основних кодових систем, встановлено, що первинними при реалізації системних функцій інформаційних технологій кодування даних є унітарні, розрядно-позиційні та коди Лібова-Крейга. Внаслідок надлишковості, зумовленої значною розрядністю подання, первинні коди застосовуються в перетворювачах форми інформації, при аналого-цифровому перетворенні, функціональних вузлах комп'ютерних систем.

Підвищити ефективність кодування даних шляхом зменшення розрядності дозволяє використання менш надлишкових двійкових та кодів Грея. Двійкові коди Радемахера використовуються, як основні в комп'ютерних системах. Коди Грея застосовуються, як основа кодових шкал та в перетворювачах форми інформації завдяки унікальній властивості зміни лише одного розряду при переході між позиціями, яка дозволяє однозначно ідентифікувати кодові комбінації.

Здійснити перехід від паралельного подання коду повідомлення до вертикальної інформаційної технології та покращити показники систем дозволяють кодові системи Галуа. Кодові системи Галуа, як і двійкові коди, використовують при реалізації процедур цифрової обробки інформації.

До основних дискретних теоретико-числових базисів належать унітарні функції та коди, функції Хаара та розрядно-позиційні коди, дискретно-фазові функції та коди Лібова-Крейга, функції Радемахера та двійкові коди, функції Грея та коди Грея, функції Уолша, функції Галуа та кодові системи Галуа. Вибір кодової системи, базису або системи функцій залежить від задачі, властивостей інформаційного потоку, умов застосування даних та інш.

Зокрема, базисами для виконання дискретних ортогональних перетворень і дискретного подання одновимірної інформаційного потоку зі скінченною енергією, визначеного в просторі $L_2[a,b]$ на часовому інтервалі $T=[a,b]$, є повні ортонормовані системи функцій. Вейвлет-аналіз здійснюється на основі ортогональних або базисів Ріса. Повними ортонормованими системами функцій у просторі $L_2[a,b]$ є тригонометрична система та дискретні експоненціальні функції – як базис перетворення Фур'є, системи Уолша, Хаара, функції піллоподібного базису, Віленкіна-Крестенсона та інші, проте актуальною залишається задача визначення галузей, способів і методів їх ефективного застосування, формування та дослідження інших базисів. Визначення особливостей та ефективності застосування різних систем функцій для виконання дискретних перетворень і аналізу інформаційних потоків

зумовлює необхідність дослідження властивостей цих систем, наведених у наступних викладах.

1. Унітарні функції та коди.

В якості вихідних у засобах перетворення форми інформації широкого застосування набули унітарні коди, розрядність бінарного подання слова яких відповідає повній шкалі квантування діапазону перетворення N . Здійснити перехід до ефективніших кодів із меншою розрядністю дозволяє аналітичне подання унітарних кодів і встановлення функціональних залежностей з іншими кодами чи системами кодування.

Для подання унітарних кодів використовуються унітарні функції:

$$Uni(m, \theta, i) = \text{sign}(\sin(2^m \pi(\theta + i \cdot 2^{-n}))), \quad (6.1)$$

де $m = 0, 1, \dots, n+1$ – порядок набору системи функцій; $n = \log_2 N$; N – модуль цілочислових дискретних значень системи; $\theta = t/T$; $(0 \leq \theta < 1)$ – нормований параметр часу; $T = 2\pi$; t – поточкове значення часу; $0 \leq t < 2\pi$; $i = 0, 1, \dots, 2^{n-m+1} - 1$ – порядковий номер функції в наборі порядку m .

Набір нульового порядку $Uni(0, \theta, i)$ містить $2N$ функцій (рис.6.1).

Властивості унітарних функцій:

1) Система з перших N унітарних функцій порядку m є лінійно незалежною, оскільки виконується достатня умова лінійної незалежності: ранг матриці N функцій дорівнює кількості функцій N . Наступні N функцій є лінійними комбінаціями N перших.

2) Унітарні функції не ортогональні, оскільки $\int_0^1 Uni(m, \theta, i) Uni(k, \theta, j) d\theta \neq 0$.

Неортогональність системи унітарних функцій зумовлює некомпактне пакування кодових елементів системи, що приводить до значної надлишковості інформаційних потоків. Внаслідок неортогональності та відсутності досліджень властивостей система не використовується як основа ТЧП.

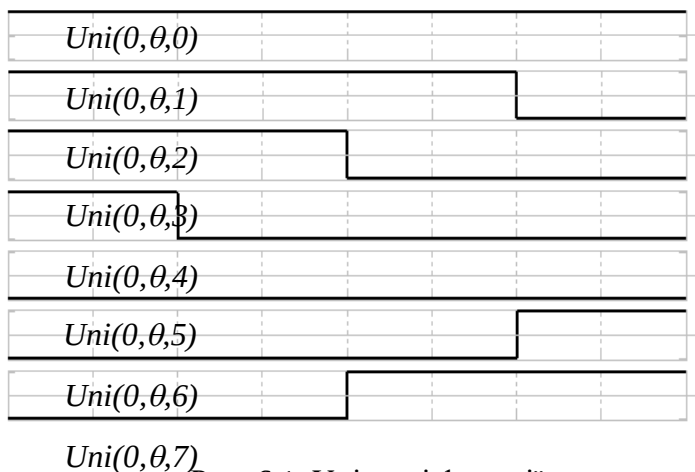


Рис. 6.1. Унітарні функції нульового порядку.

Породжуючу кодову матрицю унітарного коду розміру $N \times N$ одержують при дискретизації з інтервалом $1/N$ за параметром часу перших $N=2^n$ із системи $2N$ унітарних функцій та здійсненні бінарної заміни значень функцій 1 на 0, -1 на 1 в точках $\theta_s = s/2^n$, $s=0, 1, \dots, 2^n-1$, яка реалізується за допомогою операції

$$u_i = (1 - Uni(0, \theta_s, 2^n - 1 - i)) / 2, \quad (6.2)$$

де $u_0, u_1, \dots, u_i, \dots, u_{2^n-1}$ – значення розрядів унітарного коду $\theta_s, i=0,1,\dots,2^n-1$.

Для прикладу, при $n=3$ восьми функціям відповідають такі елементи кодової матриці

$$\begin{aligned} Uni(0,\theta,0) &\rightarrow 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \\ Uni(0,\theta,1) &\rightarrow 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \\ Uni(0,\theta,2) &\rightarrow 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \\ Uni(0,\theta,3) &\rightarrow 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \\ Uni(0,\theta,4) &\rightarrow 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \\ Uni(0,\theta,5) &\rightarrow 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \\ Uni(0,\theta,6) &\rightarrow 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \\ Uni(0,\theta,7) &\rightarrow 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \end{aligned}$$

Наведені властивості системи унітарних функцій дозволяють у наступних викладках визначити процедури перетворення до інших базисів. У ролі первинних при перетворенні форми інформації та при переході від N -розрядних унітарних до кодів із меншою розрядністю також використовуються розрядно-позиційні коди.

2. Функції Хаара та розрядно-позиційні коди.

Основою розрядно-позиційних кодів є система функцій Хаара. Функції Хаара $Har(n, \theta, j)$ (рис.6.2) визначаються за формулою:

$$Har(n, \theta, j) = \begin{cases} 2^{\frac{n-1}{2}} \text{sign}(\sin 2^n \pi \theta), & \frac{j}{2^{n-1}} \leq \theta < \frac{j+1}{2^{n-1}}, \\ 0 & \text{при інших } \theta \in [0,1), \end{cases} \quad (6.3)$$

де $n = 0,1,\dots, \log_2 N$; $j = 0,1,\dots, 2^{n-1} - 1$; ($j = 0$ при $n = 0$), $0 \leq \theta < 1$.

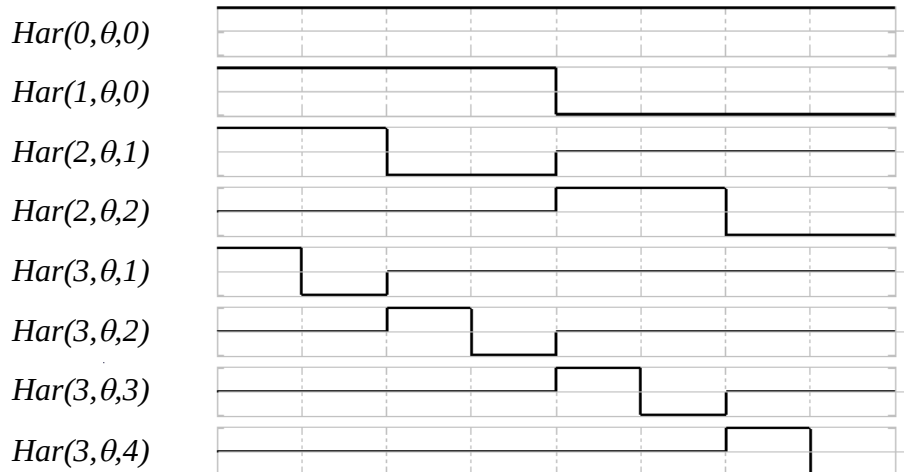


Рис.6.2. Система функцій Хаара.

При реалізації ТЧП використовуються наступні властивості системи Хаара.

1. Функції Хаара $\{Har(n, \theta, j)\}$ утворюють повну ортонормовану систему в просторі інтегровних із квадратом функцій $L_2[0,1)$, що дає можливість використовувати систему в якості базису для виконання ортогонального перетворення, яке є вейвлет-перетворенням.

2. На значній частині інтервалу визначення функції дорівнюють нулю, що дає можливість скоротити кількість арифметичних операцій при обчисленні перетворення. У результаті зменшується час обробки інформації.

Ненормований базис Хаара $\{\text{sign}(\sin 2^n \pi \theta)\}$ (без нормуючого множника $2^{\frac{n-1}{2}}$), в якому функції набувають значень $\pm 1, 0$, є основою розрядно-позиційних кодів. Розрядно-позиційні коди застосовуються в засобах перетворення форми інформації, в якості проміжних при аналогово-цифровому перетворенні, в давачах переміщень, для ініціювання комірок пам'яті тощо.

Для встановлення аналітичних співвідношень зв'язку систем функцій, які лежать в основі перетворення даних із розрядно-позиційного коду та в розрядно-позиційний код, використовуються розрядно-позиційні функції:

$$RP(i, \theta) = \begin{cases} -1, \frac{i}{2^n} \leq \theta \leq \frac{i+1}{2^n}, \\ 1 \text{ при інших } \theta \in [0, 1], \end{cases} \quad (6.4)$$

$$n = 0, 1, 2, \dots, i = 0, 1, \dots, 2^n - 1.$$

Дискретне подання 2^n розрядно-позиційних функцій та бінарна заміна значень функцій 1 на 0, -1 на 1, яка подається за допомогою виразу

$$p_i = (1 - RP(i, \theta_s)) / 2 = \frac{1}{2^{n/2}} \text{Har}(n + 1, \theta_s, i - 1), \quad (6.5)$$

де p_i – значення розрядів розрядно-позиційного коду, породжує кодову матрицю:

$$\begin{array}{cccccc} 0 & 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 \\ \hline 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & 0 & 0 & \dots & 0 & 0 \end{array}$$

За умови простої реалізації перетворення інформації унітарному та розрядно-позиційному кодам властивий недолік необхідності використання повної N -розрядної шини кодового подання даних для кодування N дискретних повідомлень. Це зумовлює необхідність переходу до ефективніших методів кодування зі зменшеною розрядністю кодів до $n = \log_2 N$ в системах Радемахера та Грея.

В якості проміжних при переході до n -розрядних кодів використовуються коди Лібова-Крейга, які дозволяють у два рази зменшити розрядність коду та характеризуються властивістю абсолютного позиціонування.

3. Дискретно-фазові функції та коди Лібова-Крейга.

Залежність унітарних кодів з іншими встановлюється за допомогою системи дискретно-фазових функцій, що є основою кодів Лібова-Крейга.

Дискретно-фазові функції порядку m подаються згідно наступного аналітичного виразу:

$$Dyf(m, \theta, i) = \text{sign}(\sin(2^m \pi (\theta + i \cdot 2^{-n}))), \quad (6.6)$$

де $i = 0, 1, \dots, 2^{n-m+1} - 1$ – порядковий номер функції в наборі порядку m .

Графіки дискретно-фазових функцій першого порядку наведені на рис.6.3.

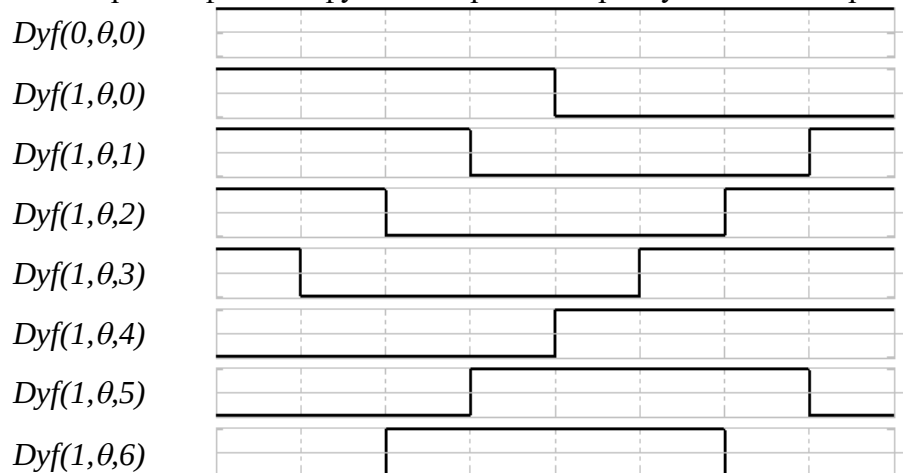


Рис.6.3. Дискретно-фазові функції першого порядку.

Властивості дискретно-фазових функцій:

1. Система з N дискретно-фазових функцій є лінійно залежною, оскільки частина функцій системи є лінійною комбінацією інших функцій системи

$$Dyf(m, \theta, j + 2^{n-m}) = -Dyf(m, \theta, j),$$

де $j = 0, 1, \dots, 2^{n-m} - 1$.

Внаслідок лінійної залежності перші N функцій не утворюють повної системи.

2. Система є неортогональною, тому що

$$\int_0^1 Dyf(m, \theta, i) Dyf(m, \theta, j) d\theta \neq 0.$$

Породжуючу кодову матрицю розміру $\frac{N}{2} \times N$ одержують за допомогою дискретизації перших $N=2^{n-1}$ дискретно-фазових функцій порядку m та здійснення бінарної заміни значень функцій 1 на 0, -1 на 1, яка реалізується за формулою

$$d_j = (1 - Dyf(1, \theta_s, 2^{n-1} - 1 - j)) / 2, \quad (6.7)$$

де $d_0, d_1, \dots, d_j, \dots, d_{2^{n-1}-1}$ – значення розрядів коду Лібова-Крейга $\theta_s = \frac{s}{2^n}$,

$s=0, 1, \dots, 2^n-1$.

Наприклад, при $N=8$ елементи матриці відповідатимуть таким функціям

$$\begin{aligned} Dyf(1, \theta, 0) &\rightarrow 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \\ Dyf(1, \theta, 1) &\rightarrow 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \\ Dyf(1, \theta, 2) &\rightarrow 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \\ Dyf(1, \theta, 3) &\rightarrow 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \\ s &\rightarrow 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7. \end{aligned}$$

Внаслідок неповноти, система не використовується як основа для ортогональних ТЧП. Дискретно-фазові функції розглядаються як перехідні та як основа творення базисів і систем функцій Радемахера, Грея, кодування даних в яких здійснюється з розрядністю $n = \log_2 N$ порівняно з N для унітарних кодів. Із цією метою в складі системи дискретно-фазових функцій виокремлюють дві підсистеми функцій виду $sign(\sin(2^n \pi \theta))$ та $sign(\cos(2^n \pi \theta))$.

4. Система функцій Радемахера та двійкові коди.

Екстракція $\sin$ -складових набору дискретно-фазових функцій утворює систему функцій Радемахера (рис. 6.4)

$$Rad(n, \theta) = Dyf(n, \theta, 0) = sign(\sin(2^n \pi \theta)). \quad (6.8)$$

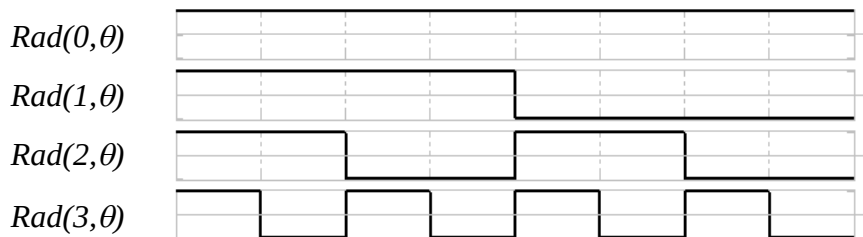


Рис.6.4. Функції Радемахера.

Система Радемахера є основою двійкової системи числення.

Відповідність між значеннями функцій у точках $\theta_s = s/2^n$, $s=0, 1, \dots, 2^n-1$ та їх поданням у двійковому коді $\theta_s = r_n r_{n-1} \dots r_0$ встановлюється співвідношенням

$$r_k = (1 - \text{Rad}(n - k, \theta_s)) / 2, \quad (6.9)$$

де r_k – значення розрядів двійкового коду, $k = 0, 1, \dots, n$.

Наприклад, при $n=3$ чотирьом функціям відповідають такі елементи кодової матриці розміру 4×8 :

$$\begin{array}{ll} \text{Rad}(0, \theta) & \rightarrow 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \\ \text{Rad}(1, \theta) & \rightarrow 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \\ \text{Rad}(2, \theta) & \rightarrow 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \\ \text{Rad}(3, \theta) & \rightarrow 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \\ s & \rightarrow 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7. \end{array}$$

Наступні властивості системи Радемахера:

- функції Радемахера ортонормовані на відрізку $[0, 1)$, оскільки

$$\int_0^1 \text{Rad}(n, \theta) \text{Rad}(k, \theta) d\theta = 0 \text{ та } \int_0^1 \text{Rad}(n, \theta) \text{Rad}(n, \theta) d\theta = 1.$$

- система функцій Радемахера утворює в просторі інтегровних із квадратом функцій $L_2[0, 1)$ неповну систему ортонормованих функцій, оскільки для довільного n не виконується означення повноти системи

$$\int_0^1 \text{Rad}(n, \theta) \text{Rad}(1, \theta) \text{Rad}(2, \theta) d\theta = 0,$$

тобто існує функція $\text{Rad}(1, \theta) \text{Rad}(2, \theta)$, яка тотожно не дорівнює нулю на інтервалі $[0, 1)$ та ортогональна до всіх функцій системи.

Неповнота системи Радемахера обмежує її застосування для подання інформаційних потоків на основі ортогональних перетворень. Одночасно із широким застосуванням, творенням за допомогою системи Радемахера двійковим кодам властивий недолік, що полягає в неоднозначності формування відліків суміжних кодів при мікрозрядному позиціонуванні. Уникнути такої вади дозволяє перехід до кодів Грея

5. Система функцій Грея та коди Грея.

Екстракція $\cos$ -складових згідно кожного з порядків n набору дискретно-фазових функцій утворює систему функцій Грея. Система функцій Грея є підмножиною системи дискретно-фазових функцій

$$\begin{aligned} \text{Gry}(0, \theta) &= \text{Dyf}(0, \theta + 2^{-1}, 0); \\ \text{Gry}(m, \theta) &= \text{Dyf}(m, \theta, 2^{n-m-1}), \quad m = 1, 2, \dots, n. \end{aligned} \quad (6.10)$$

Графіки функцій Грея наведено на рис.6.5.

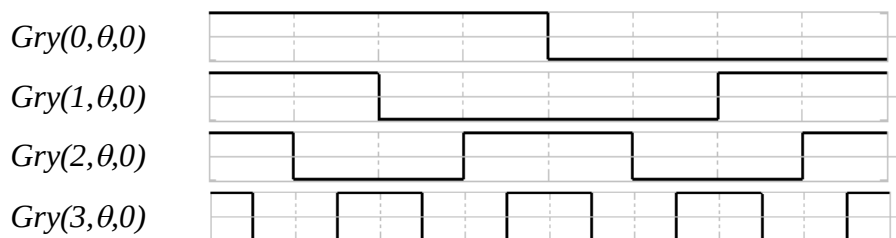


Рис.6.5. Функції Грея.

Система функцій Грея є ортонормованою та неповною, що доводиться наступними викладками.

Відомо, що система Грея є складовою підсистемою функцій Уолша. Оскільки система функцій Уолша є ортонормованою, то і функції Грея, як її підсистема, є ортонормованими, тобто

$$\int_0^1 Gry(n, \theta) Gry(k, \theta) d\theta = 0, \quad \int_0^1 Gry(k, \theta) Gry(k, \theta) d\theta = 1.$$

Функції Грея утворюють у $L_2[0,1]$ неповну систему, оскільки існують функції, які тотожно не дорівнюють нулю та ортогональні до всіх функцій системи, зокрема, для довільного n

$$\int_0^1 Gry(n, \theta) Rad(2, \theta) d\theta = 0.$$

Неповнота системи Грея обмежує її застосування для розкладання інформаційних потоків та реалізації ТЧП.

Розширення функціональних можливостей при реалізації системних функцій перетворення форми та обробки інформації забезпечує перехід до базису Уолша. Аналітичні залежності процедури переходу базуються на наступній властивості скінченних добуток функцій системи Грея.

Якщо $(k_1, \dots, k_m)$ і $(l_1, \dots, l_r)$ дві різні скінченні послідовності, то

$$\int_0^1 [Gry(k_1, \theta) \dots Gry(k_m, \theta)] [Gry(l_1, \theta) \dots Gry(l_r, \theta)] d\theta = 0 \quad (6.11)$$

Для доведення властивості необхідно перепозначити множники в підінтегральному виразі $Gry(j_1, \theta), Gry(j_2, \theta), \dots, Gry(j_p, \theta)$ ($j_1 < j_2 < \dots < j_p$). Добуток пар функцій $Gry(j_k, \theta) Gry(j_k, \theta) = 1$. Добуток інших множників $Gry(j_1, \theta) Gry(j_2, \theta) \dots Gry(j_{p-1}, \theta)$ є частково-сталою функцією, кожний з інтервалів сталості якої можна поділити на парне число рівних підінтервалів, на яких $Gry(j_p, \theta)$ набуває по чергові значення $+1$ і -1 або -1 і $+1$. Із врахуванням чого

$$\int_0^1 [Gry(j_1, \theta) \dots Gry(j_p, \theta)] d\theta = \text{const} \int_I Gry(j_p, \theta) d\theta = 0,$$

а тому буде рівним нулю значення інтегралу на інтервалі $[0;1]$. Тобто два різні добутки функцій системи є ортогональними, що і треба довести.

Система функцій Грея є основою кодів Грея. Відповідність між значеннями функцій у точках $\theta_s = s/2^n$, $s=0,1,\dots,2^n-1$ та їх поданням у коді Грея $\theta_s = h_n h_{n-1} \dots h_0$ встановлюється співвідношенням

$$h_k = (1 - Gry(n - k - 1, \theta_s)) / 2, \quad (6.12)$$

де $k = 0, 1, \dots, n-1$.

Наприклад, чотирьом функціям відповідають елементи кодової матриці розміру 4×8

$$\begin{array}{ll} Gry(0, \theta) & \rightarrow 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \\ Gry(1, \theta) & \rightarrow 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \\ Gry(2, \theta) & \rightarrow 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \\ Gry(3, \theta) & \rightarrow 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \\ s & \rightarrow 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7. \end{array}$$

Кодування Грея дозволяє зменшити похибки формування відліків суміжних кодів унаслідок зміни тільки одного розряду порівняно із застосуванням двійкових кодів. Однак, неповнота систем функцій Радемахера та Грея звужує галузі їх ефективного застосування. Розширення функціональних можливостей при реалізації системних функцій перетворення форми та обробки інформації забезпечує базис Уолша.

Властивості систем Радемахера, Грея та відповідних кодів визначають процедуру

переходу в базис Уолша $Wal(i, \theta)$, $i = 0, 1, \dots, 2^n - 1$, впорядкований за Уолшем, із системи Радемахера $Rad(n, \theta)$. Функції Уолша $Wal(i, \theta)$ (рис.6.6) визначаються, як добуток функцій Радемахера

$$Wal(i, \theta) = Rad(1, \theta)^{b_0} Rad(2, \theta)^{b_1} \dots Rad(n, \theta)^{b_{n-1}} = \prod_{k=0}^{n-1} (Rad(k+1, \theta))^{b_k}, \quad (6.13)$$

де $i = b_{n-1}b_{n-2} \dots b_1b_0$ – подання в коді Грея порядкового номера функції $Wal(i, \theta)$.

Система функцій Уолша є ортонормованою, повною та мультиплікативною.

Функції Уолша ортонормовані на інтервалі $0 \leq \theta \leq 1$

$$\int_0^1 Wal(k, \theta) Wal(i, \theta) d\theta = \begin{cases} 1 & \text{if } k = i, \\ 0 & \text{if } k \neq i. \end{cases}$$

Функції Уолша утворюють мультиплікативну систему

$$Wal(k, \theta)Wal(i, \theta) = Wal(k \oplus i, \theta).$$



Рис.6.6. Система функцій Уолша.

Системи Радемахера та Грея є підсистемами функцій Уолша

$$Wal(2^n-1, \theta) = Rad(n, \theta),$$

$$Wal(2^n, \theta) = Gry(n, \theta).$$

Відомі схеми генераторів функцій Уолша базуються на методі формування функцій Уолша із системи Радемахера, але використання в ньому двійкового коду зумовлює виникнення помилки неоднозначності. На основі доведеної властивості (6.11) ортогональності добутоків функцій Грея розроблено альтернативний метод формування функцій Уолша із системи функцій Грея:

$$Wal(i, \theta) = Gry(0, \theta)^{a_0} Gry(1, \theta)^{a_1} \dots Gry(n-1, \theta)^{a_{n-1}} = \prod_{k=0}^{n-1} (Gry(k, \theta))^{a_k}, \quad (6.14)$$

де $i = a_{n-1}a_{n-2} \dots a_1a_0$ – подання в двійковому коді порядкового номера функції Уолша $Wal(i, \theta)$.

У відомих генераторах і перетворювачах функції Уолша формуються згідно (6.19) на основі двійкового коду наступним способом. Функції Радемахера відповідають значенням розрядів двійкового коду

$$Rad(m, \theta_s) = \overline{r_{n-m}} - r_{n-m} = \begin{cases} 1, & \text{якщо } r_{n-m} = 0, \\ -1, & \text{якщо } r_{n-m} = 1, \end{cases} \quad (6.15)$$

де r_m – значення m -го розряду двійкового коду аргумента θ_s , $m=0,1,\dots,n$.

При підстановці (6.15) у вираз (6.13) функції Уолша визначаються на основі двійкового коду аргументу θ_s

$$\begin{aligned} Wal(i, \theta_s) &= (\overline{r_{n-1}} - r_{n-1})^{b_0} (\overline{r_{n-2}} - r_{n-2})^{b_{n1}} \dots (\overline{r_0} - r_0)^{b_{n-1}} = \\ &= (\overline{b_0 r_{n-1}} \oplus b_1 r_{n-2} \oplus \dots \oplus b_{n-1} r_0) - (b_0 r_{n-1} \oplus b_1 r_{n-2} \oplus \dots \oplus b_{n-1} r_0) = \\ &= \overline{\varphi_i(\theta_s)} - \varphi_i(\theta_s) \end{aligned} \quad (6.16)$$

де $\varphi_i(\theta_s) = b_0 r_{n-1} \oplus b_1 r_{n-2} \oplus \dots \oplus b_{n-1} r_0$.

Із використанням наведеної методики можна визначити спосіб формування функцій Уолша на основі аргумента, поданого в коді Грея.

Функції Грея $Gry(k, \theta_s)$ відповідають значенням n розрядів коду Грея θ_s згідно залежності

$$Gry(k, \theta_s) = \overline{h_{n-k-1}} - h_{n-k-1} = \begin{cases} 1, & \text{якщо } h_{n-k-1} = 0, \\ -1, & \text{якщо } h_{n-k-1} = 1, \end{cases} \quad (6.17)$$

де h_k – значення k -го розряду коду Грея аргумента θ_s ; $k=0,1,\dots,n-1$.

Функції Уолша визначаються при підстановці функцій Грея з (6.17) у (6.14):

$$Wal(i, \theta_s) = (\overline{h_{n-1}} - h_{n-1})^{a_0} (\overline{h_{n-2}} - h_{n-2})^{a_1} \dots (\overline{h_0} - h_0)^{a_{n-1}},$$

де $\theta_s = h_{n-1} \dots h_1 h_0$ – код Грея, $i = a_{n-1} a_{n-2} \dots a_0$ – подання у двійковому коді числа i .

Перетворення добутку в правій частині рівності з використанням основних тотожних співвідношень булевої алгебри дозволяє визначити функції Уолша

$$Wal(i, \theta_s) = (\overline{a_0 h_{n-1} \oplus a_1 h_{n-2} \oplus \dots \oplus a_{n-1} h_0}) - (a_0 h_{n-1} \oplus a_1 h_{n-2} \oplus \dots \oplus a_{n-1} h_0), \quad (6.18)$$

$$Wal(i, \theta_s) = \overline{\varphi_i(\theta_s)} - \varphi_i(\theta_s) \quad (6.19)$$

де $\varphi_i(\theta_s) = a_0 h_{n-1} \oplus a_1 h_{n-2} \oplus \dots \oplus a_{n-1} h_0$.

Перевагою перетворювача на основі (6.18) є використання кодів Грея, які дозволяють зменшити помилки в засобах перетворення та обробки.

Таким чином, повна система Уолша, яка утворюється із систем Радемахера та Грея, є базисом для виконання ортогональних перетворень і формування функцій Галуа.

7. Система функцій Галуа та кодові системи Галуа.

Перехід до різних упорядкувань функцій у системі Галуа здійснюється з базису Уолша з упорядкуванням функцій за рекурсивним законом. За n -розрядними фрагментами рекурсивної послідовності, яка утворюється відповідно до породжуючого вектора поля Галуа $GF(2^n)$, згідно відображення через систему функцій Радемахера формуються номери функцій Уолша та Галуа в системі.

Наприклад, у полі $GF(2^3)$ існують породжуючі вектори 1011 та 1101. Для даного поля $GF(2^3)$ рекурсивні послідовності $v_0, v_1, v_2, v_3, v_4, \dots$ формуються з початкового вектора $(v_0 v_1 v_2) = (111)$ за правилами:

$$1) 1101 \rightarrow v_{i+3} = v_i \oplus v_{i+1}: v_0, v_1, v_2, v_0 \oplus v_1, v_1 \oplus v_2, v_0 \oplus v_1 \oplus v_2, v_0 \oplus v_2, v_0 \oplus v_1, v_2, \dots; \quad (6.20)$$

$$2) 1011 \rightarrow v_{i+3} = v_i \oplus v_{i+2}: v_0, v_1, v_2, v_0 \oplus v_2, v_0 \oplus v_1 \oplus v_2, v_0 \oplus v_1, v_1 \oplus v_2, v_0, v_1, v_2, \dots \quad (6.21)$$

Для початкового вектора $(v_0 v_1 v_2) = (111)$ утворена на основі породжуючого вектора 1101 рекурсивна послідовність $\{0 0 0 1 0 1 1 1\}$, визначає наступне рекурсивне впорядкування номерів функцій Уолша в системі $\{0 1 2 5 3 7 6 4\}$.

$$\begin{array}{cccccccc} 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & & & & & & & \rightarrow 0 \\ 0 & 0 & 1 & & & & & & & \rightarrow 1 \\ 0 & 1 & 0 & & & & & & & \rightarrow 2 \\ 1 & 0 & 1 & & & & & & & \rightarrow 5 \\ 0 & 1 & 1 & & & & & & & \rightarrow 3 \\ 1 & 1 & 1 & & & & & & & \rightarrow 7 \\ 1 & 1 & 0 & & & & & & & \rightarrow 6 \\ 1 & 0 & 0 & & & & & & & \rightarrow 4 \end{array}$$

Утворена на основі породжуючого вектора 1011 рекурсивна послідовність $\{0 0 0 1 1 1 0 1\}$ визначає інше рекурсивне впорядкування номерів функцій Уолша: $\{0 1 3 7 6 5 2 4\}$.

Із рекурсивно впорядкованої системи Уолша відповідно впорядковані перші n функцій Галуа формуються згідно співвідношення

$$Gal(n, \theta, i) = Wal(Ent(2^n \theta), \frac{2^{i+1} - 1}{2^n}) \quad (6.22)$$

де $i = 0, 1, \dots, 2^n - 1$, Ent – функція виділення цілої частини.

Проведені дослідження встановили можливість формування функцій Галуа із систем Радемахера та Грея. Згідно співвідношень 65.19) та (6.20) перші n функцій Галуа в системі подаються у вигляді добутку функцій Радемахера та Грея

$$Gal(n, \theta, i) = \prod_{k=0}^{n-1} (Rad(k+1, \frac{2^{i+1} - 1}{2^n}))^{h_k} = \prod_{k=0}^{n-1} (Gry(k+1, \frac{2^{i+1} - 1}{2^n}))^{r_k}, \quad (6.23)$$

де $h_{n-1} h_{n-2} \dots h_0$ – запис у коді Грея числа q , двійковий код якого є n -розрядним фрагментом $v_i v_{i+1} v_{i+2} \dots v_{i+n-1}$ рекурсивної послідовності $v_0, v_1, v_2, \dots$; $r_{n-1} r_{n-2} \dots r_0$ – двійковий код, який є n -розрядним фрагментом $v_i v_{i+1} v_{i+2} \dots v_{i+n-1}$ рекурсивної послідовності $v_0, v_1, v_2, \dots$.

Повний набір 2^n функцій рекурсивної системи Галуа $Gal(n, \theta, i)$ отримують із перших n функцій системи процедурою рекурсивного зсуву на $\Delta\theta = \frac{1}{2^n}$ згідно другої діагоналі кожної наступної функції відносно попередньої

$$Gal(n, \theta, i+1) = Gal(n, \theta + \Delta\theta, i) \quad (6.24)$$

Впорядкування функцій Галуа в наборі відповідає синтезованому за породжуючим вектором упорядкуванню функцій Уолша.

Процедура переходу від дискретних значень функцій Уолша до дискретних значень функцій Галуа подається матричною операцією

$$\|Gal\| = \|W\| \cdot \|R\|,$$

де $\|Gal\|$ – матриця розміру $N \times n$ системи Галуа; $\|W\|$ – матриця розміру $N \times N$ рекурсивно впорядкованих функцій Уолша; $\|R\|$ – матриця розміру $N \times n$ відображеної вагової мережі Радемахера.

Для прикладу, матрична операція переходу від функцій Уолша до функцій Галуа та матриця розміру 8×8 дискретних значень функцій Галуа в полі $GF(2^3)$ з породжуючим вектором 1101 згідно процедури рекурсивного розширення подаються відповідно

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & -1 \\ -1 & -1 & -1 \\ -1 & -1 & 1 \\ -1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \\ 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ -1 & -1 & -1 & 1 & 1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 \end{pmatrix}.$$

Графіки функцій Галуа $Gal(n, \theta, i)$ з породжуючим вектором 1101 при $n=3$ наведено на рис.6.7.

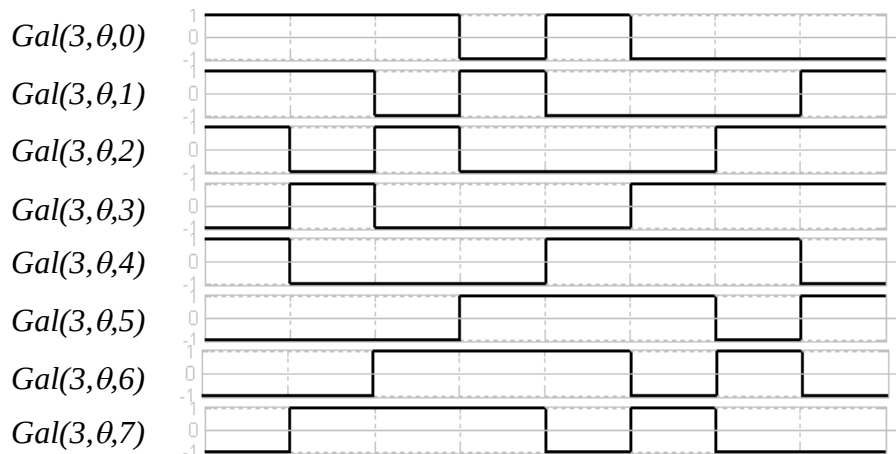


Рис.6.7. Функції Галуа $Gal(3, \theta, i)$ з породжуючим вектором 1101.

При виконанні перетворень використовуються наступні властивості системи функцій Галуа.

1. У системі 2^n функцій n -го порядку кожна підсистема із n функцій $\{Gal(n, \theta, i), Gal(n, \theta, i+1), Gal(n, \theta, i+2), \dots, Gal(n, \theta, i+n-1)\}$ ортогональна.
2. Симетрія індекса та аргумента. Елементи матриці системи Галуа симетричні відносно головної діагоналі

$$Gal(n, \theta_s, i) = Gal(n, \frac{i}{2^n}, s),$$

де $i, s \in \{0, 1, \dots, 2^n - 1\}$.

Таким чином, матриці системи Галуа є ганкелевими антициклічними, оскільки при $i+j=k+l$ $G_{ij}=G_{kl}$. Якщо рекурсивний зсув у (6.28) здійснюється згідно головної діагоналі та у формулі (6.28) $\Delta\theta = -\frac{1}{2^n}$, то матриці є тоєпліцевими циклічними (циркулянтними), оскільки $i+j=k+l$ $G_{ij}=G_{kl}$.

3. Міри довжин інтервалів, на яких $Gal(n, \theta_s, i) = 1$ і $Gal(n, \theta_s, i) = -1$ однакові, отже

$$\int_0^1 Gal(n, \theta, i) d\theta = \sum_{s=0}^{2^n-1} Gal(n, \theta_s, i) = 0 \quad (6.25)$$

тобто множина функцій Галуа задовольняє необхідну умову для вейвлет-функцій.

При дискретизації за параметром часу перших n функцій Галуа та здійсненні бінарної заміни значень функцій 1 на 0, -1 на 1, згідно виразу

$$g_k(\theta_s) = (1 - Gal(n - k - 1, \theta_s)) / 2, \quad (6.26)$$

одержують матрицю кодових елементів Галуа розміру $n \times N$, $k = 0, 1, \dots, n - 1$.

Наприклад, при $N=8$ рядки матриці кодових елементів Галуа з породжуючим вектором 1101 відповідатимуть таким функціям

$$\begin{aligned} Gal(3, \theta, 0) &\rightarrow 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1 \\ Gal(3, \theta, 1) &\rightarrow 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \\ Gal(3, \theta, 2) &\rightarrow 0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \\ s &\rightarrow 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \end{aligned}$$

Елементи матриці кодових елементів Галуа з породжуючим вектором 1011 відповідатимуть наступним функціям

$$\begin{aligned} Gal(3, \theta, 0) &\rightarrow 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \\ Gal(3, \theta, 1) &\rightarrow 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\ Gal(3, \theta, 2) &\rightarrow 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \\ s &\rightarrow 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \end{aligned}$$

При дискретизації системи N функцій Галуа $\{Gal(n, \theta, i)\}$, $i = 0, 1, \dots, 2^n - 1$ та перетворенні значень функцій згідно (5.30) отримують повну матрицю кодових елементів Галуа розміру $N \times N$, впорядкованих із поелементним рекурсивним зсувом згідно другої діагоналі матриці Галуа. Номер s повідомлення однозначно визначається n -координатним вектором $n = \log_2 N$.

Представлення систем ортогональних функцій різних ТЧБ подані у табл.6.1.

З метою оцінки ефективності кодування даних на основі різних ТЧБ доцільно провести аналіз кодових матриць, які породжують різні системи числення.

При цьому важливою характеристикою кожного базису є об'єм його кодової матриці M_j та число активних елементів m_j (рис.6.8), що визначає характеристики надлишковості представлення інформації на основі аналітичної оцінки

$$V_i = n_i \cdot N_i,$$

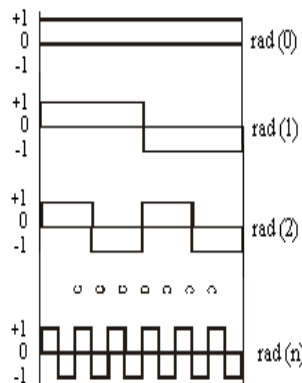
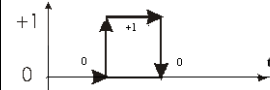
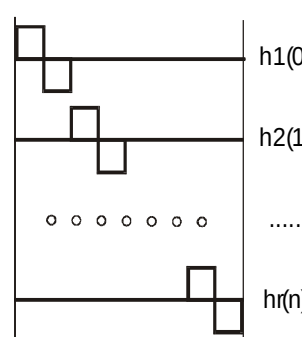
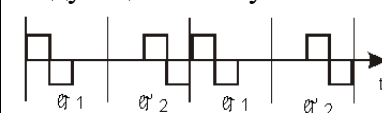
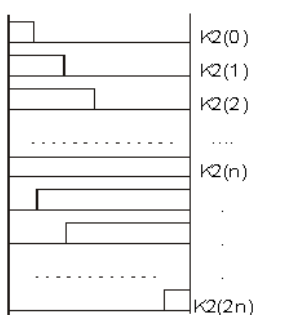
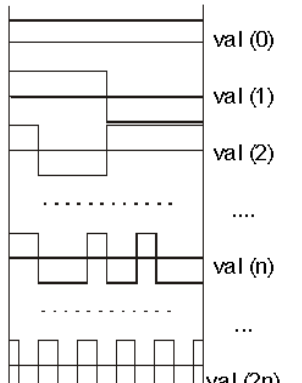
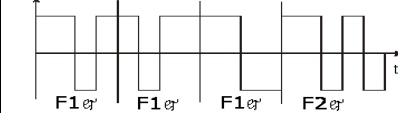
де n_j – розрядність числа; N_i – число незалежних кодових значень.

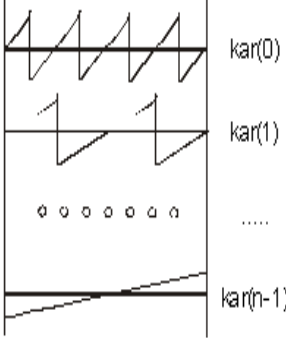
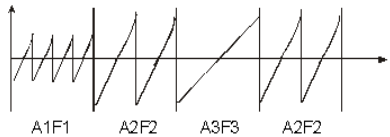
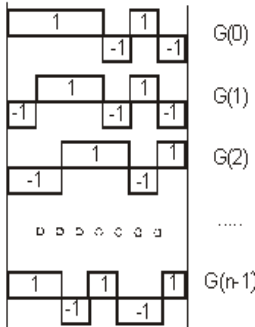
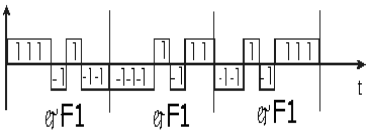
Кожен з названих базисів характеризується визначеним об'ємом кодової матриці для представлення даних. При цьому найбільш надлишковим базисом є унітарний, в якого кодова матриця $V = N^2$, а число активних кодових елементів $n = N^2 / 2$, де N – діапазон кодування даних. Аналогічну надлишковість забезпечує базис Хаара, в два рази меншу надлишковість забезпечує базис Крейга, тобто $V = N^2 / 4$, а $n = N^2 / 8$. Максимально широке застосування для кодування даних в сучасних КС отримали базиси Радемахера та Крестенсона, в яких $V = N \log_2 N$. Дані базиси відповідно породжують двійкову систему числення та систему числення залишкових класів.

Базис Уолша максимально широко використовується в сучасних телекомунікаційних КС. Даний базис породжує систему ортогональних шумоподібних сигналів, які використовуються в сотових системах мобільного зв'язку.

Таблиця 6.1

Представлення теоретико-числових базисів

Базис	Представлення базису	Базисна функція та об'єм матриці V	Модуляція сигналу та спектр
Радемахера		$\text{Rad}(n, \theta) = \text{sign}[2^n \pi \cdot \theta]$ $V = N \cdot \log_2 N$	Базисні функції Радемахера є основою для модуляції Прямокутних сигналів  Базис Радемахера породжує двійкову систему числення і двійкові коди
Хаара		$\text{Har}(n, \theta, i) = \text{sign}[\sin(i 2^n \pi, \theta)]$ $V = N^2$	Використовується при фазовій модуляції сигналу.  В даному базисі використовуються розрядно-позиційні коди.
Крейга		$\text{Crg}(n, \theta) = \text{sign}[\sin((2^{n-1} \cdot \pi \cdot \theta)]$ $V = \frac{N^2}{2}$	Породжує тривалісні і фазові методи модуляції сигналу. 
Уолша		$\text{Had}(h, x) = \prod_{i=1}^K [r_i(x)] h_i$	Породжує частотно-фазові методи модуляції сигналу.  Базис Уолша має найбільш широкий спектр сигналу

Крес- тенсона		$N_i = \text{res} \sum_{i=1}^n (B_i \cdot b_i) \bmod P$ $V = \sum_{i=1}^m \log_2(P_i)$	Даний базис породжує амплітудно-частотні методи модуляції.  Базис представлений трикутними функціями. Спектр сигналів такого базису є експоненціальний.
Галуа		$N_j = f(C_{j-n-1}, \dots, C_{j-1}, C_j),$ $C_j = \sum_{j=0}^{n-1} C_{j-1} \cdot A \cdot (\bmod 2)$ $V = N$	Базис Галуа породжує коди поля Галуа і систему числення Галуа.  Модуляція в базисі як фазова, так і частотна.

$$M_{\text{Uni}} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & 0 & \dots & 0 & 0 \\ 1 & 1 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 1 & 0 \\ 1 & 1 & 1 & \dots & 1 & 1 \end{pmatrix} \quad a)$$

$$M_{\text{Har}} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix} \quad б)$$

$$M_{\text{Gr}} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 1 & 1 & 1 \\ 1 & 1 & 1 & \dots & 1 & 1 & 0 \\ 1 & 1 & 1 & \dots & 0 & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & \dots & 0 & 0 & 0 \end{pmatrix} \quad в)$$

$$M_{\text{Rad}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & 0 & \dots & 1 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 1 & 0 \\ 1 & 1 & 1 & \dots & 1 & 1 \end{pmatrix} \quad г)$$

$$M_{\text{LibCr}} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 1 & 0 \\ 1 & 1 & 1 & \dots & 1 & 1 \\ 0 & 1 & 1 & \dots & 1 & 1 \\ 0 & 0 & 1 & \dots & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix} \quad д)$$

$$M_{\text{Cres}} = \begin{pmatrix} P_1 & P_2 & \dots & P_n \\ 0 & 0 & \dots & 0 \\ 1 & 1 & \dots & 1 \\ 2 & 2 & \dots & 2 \\ 0 & 3 & \dots & 3 \\ 1 & 4 & \dots & 4 \\ 2 & 0 & \dots & 5 \\ 0 & 1 & \dots & 6 \\ \dots & \dots & \dots & \dots \\ a_1 & a_2 & \dots & a_n \end{pmatrix} \quad е)$$

$$M_{\text{Gal}} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad є)$$

Рис.6.8. Кодові матриці дискретних базисів:
а) унітарного; б) Хаара; в) Грея; г) Радемахера; д) Крейга; е) Крестенсона; є) Галуа.

Найменшу надлишковість кодування даних забезпечує базис Галуа, кодова матриця якого $V = N$, а $n = N / 2$.

Згідно викладеного, характеристики ТЧБ кодування даних, як системного об'єкта, подані в табл.6.2.

Таблиця 6.2.

Характеристики потоків даних

Формувачі вхідних та вихідних інформаційних сигналів даних	Характеристики інформаційних потоків даних
Унітарний базис	$V = N^2; n = N^2 / 2$
Базис Хаара	$V = N^2, n = N$
Базис Крейга	$V = N^2 / 4, n = N^2 / 8$
Базис Радемахера	$V = N \cdot \log_2 N, n = \frac{N}{2} \log_2 N$
Базис Крестенсона	$V = N \cdot \log_2 N$
Базис Уолша	$V = N^2, n = N^2 / 2$
Базис Галуа	$V = N, n = N / 2$

СПД представлені швидкістю приймання, швидкістю передавання даних, імовірністю помилок, часом затримки в каналі

$$E_{СПД} = F(V_R, V_W, P_i, T),$$

де V_R – швидкість приймання даних, V_W – швидкість передавання даних, P_i – імовірність помилок, T – час затримки в каналі [6].

В таблиці 6.3 подані символи та аналітичні вирази моделюючих функцій в залежності від вибору базису [5].

Таблиця 6.3

Символи та аналітичні вирази моделюючих функцій

№	Базисні функції	ТЧБ	Базисні функції
0	$\psi = \cos$	Фур'є	$S_x = A\psi(\omega t + \varphi)$
1	$\psi = \text{uni}$	Унітарний	$\text{Uni}(n, \Theta, i) = \text{sign} [\sin((2^n - 1) i \pi)]$
2	$\psi = \text{har}$	Хаара	$\text{har}(0,0,t)=1, t \in [0;1);$ $\text{har}(r,m,t) = \begin{cases} 2^{r/2}, \frac{m-1}{2^r} \leq t < \frac{m-1/2}{2^r}; \\ 2^{r/2}, \frac{m-1/2}{2^r} \leq t < \frac{m}{2^r} \end{cases}$ де $0 \leq r < \log_2 N$ і $1 \leq m < 2^r$.
3	$\psi = \text{crg}$	Крейга	$\text{Crg}(n, \Theta) = \text{sign} [\sin((2^n - 1) \pi \Theta)],$
4	$\psi = \text{rad}$	Радемахера	$\text{Rad}(m,t) = \text{rad}(1, 2^{m-1} t),$ $\text{Rad}(1,t) = \begin{cases} 1, t \in [0;1/2), \\ 1, t \in [0;1/2;1) \end{cases}$

5	$\psi = \text{gry}$	Грея	$\text{Gry}(0, \Theta) = \text{Dyf}(0, \Theta + 2^{-1}, 0);$ $\text{Gry}(m, \Theta) = \text{Dyf}(m, \Theta + 2^{n-m-1}), m = 1, 2, \dots, n$
6	$\psi = \text{crs}$	Крестернсона	$N_i = \text{res} \sum_{i=1}^n (B_i \cdot b_i) \bmod P \quad V = \sum_{i=1}^n \log_2(P_i),$
7	$\psi = \text{wal}$	Уолша	$\text{Wal}(j, \Theta) = \prod_{v=0}^{n-1} [\text{rad}(n, \Theta)]^{j_v},$
8	$\psi = \text{gal}$	Галуа	$N_j = f(C_{j-n-1}, \dots, C_{j-1}, C_j),$ $C_j = \sum_{j=0}^{n-1} C_j - 1 \cdot A \cdot (\bmod 2)$

Теоретико-числові бази— є фундаментальними теоретичними основами систем числення та методів кодування даних. Базис—породжується системами ортогональних функцій. Тип сигналу визначається базисом. Класифікація охоплює наступні вісім ТЧБ: Унітарний, Хаара, Крейга, Уолша, Радемахера, Грея, Крестенсона, Галуа.

6.3. Кодові матриці дискретних ТЧБ

6.3.1. ТЧБ Унітарний

Базис унітарних функцій можна представити у вигляді [4,5,77]:

$$\text{Uni}(n, \Theta, i) = \text{sign}[\sin((2^n - 1) i \pi)], \quad (6.27)$$

де $i = 0, 1, 2, \dots, n$ - порядковий номер функції; при цьому $\Theta = 1/T$ - параметр часу, час формований до інтервалу значення T ; біжуче значення часу- t .

Унітарний базис описується наступною кодовою матрицею (6.28) :

$$M_{\text{uni}} = \begin{pmatrix} 0 & 0 & . & . & . & 0 & 0 & 1 \\ 0 & 0 & . & . & . & 0 & 1 & 1 \\ 0 & 0 & . & . & . & 1 & 1 & 1 \\ . & . & . & . & . & . & . & . \\ 1 & 1 & . & . & . & 1 & 1 & 1 \end{pmatrix} N \quad (6.28)$$

$\xrightarrow{\quad m \quad}$

де m —число стовпців; (m) ; N - число рядків; об'єм матриці : $V = N m$; або $V = N^2$ при $m = N$; число активних елементів: $n = N^2 / 2$.

Перевагою: унітарного ТЧБ є широке використання в обчисленнях з високим ступенем паралелізму операцій. Зокрема в матричних око-процесорах, спецпроцесорах кореляційної обробки даних, процесорах швидкого перетворення Фур'є та аналого-цифрових перетворювачах.

Недоліком: унітарного ТЧБ є велика надлишковість та обумовлена значним числом активних елементів і великим об'ємом кодової матриці [10,43].

6.3.2. ТЧБ Хаара

Множину функції Хаара $\{\text{har}(n, m, t)\}$, що створюють періодичну, ортонормовану й повну систему функцій, запропонував Хаари в 1910 році. Рекурентне співвідношення, що дає змогу отримати $\text{har}(n, m, t)$, має вигляд

$$\text{har}(0, 0, t) = 1, t \in [0; 1);$$

$$\text{har}(r, m, t) = \begin{cases} 2^{r/2}, \frac{m-1}{2^r} \leq t < \frac{m-1/2}{2^r}; \\ 2^{r/2}, \frac{m-1/2}{2^r} \leq t < \frac{m}{2^r}, \end{cases} \quad (6.29)$$

де $0 \leq r < \log_2 N$ і $1 \leq m < 2^r$.

Дискретизація системи функцій Хаара ($N=8$) зумовлює матрицю, кожен рядок якої є дискретною функцією Хаара $\text{Har}(n, m, t)$.

Функції Хаара $\text{Har}(n, \Theta, i)$ є набором вибірових значень функцій Фур'є (Радемахера) на i -му періоді, i є теоретичною основою розряднопозиційних кодів.

$$\text{Har}(n, \Theta, i) = \text{sign}[\sin((2^n i \pi), \Theta)], \quad (6.30)$$

ТЧБ Хаара реалізується на основі імпульсних ортогональних функцій Хаара і його широко використовують при фазовому кодуванні інформаційних сигналів, таблиць, меню, та інш. [4,5,77].

Кодова матриця має наступні характеристики (6.31):

$$M_{\text{har}} = \begin{array}{c} \begin{array}{cccccc|c} 0 & 0 & . & . & . & 0 & 0 & 1 \\ 0 & 0 & . & . & . & 0 & 1 & 0 \\ 0 & 0 & . & . & . & 1 & 0 & 0 \\ . & . & . & . & . & . & . & . \\ 1 & 0 & . & . & . & 0 & 0 & 0 \end{array} \begin{array}{c} N \\ \downarrow \end{array} \end{array} \quad (6.31)$$

$\xrightarrow{\quad m \quad}$

Отримані таким чином матриці використовуються для перетворення Хаара і позначається $M^*(n)$, де m -число стовпців; N -число рядків при $m=N$;об'єм матриці: $V = N^2$.

Базис Хаара також характеризується великою надлишковістю, але має мінімальне число активних елементів $n = \log_2 N$.

Перетворення Хаара, пілкоподібне, дискретне косинусне перетворення, а також перетворення Карунена–Лоева застосовується в системах стиснення даних, при обробці зображень та в інших системах цифрової обробки сигналів.

Переваги: широко використовується в обчислювальній техніці і реалізується адресними дешифраторами та організації інтерактивної інтерфейсної взаємодії оператора з комп'ютером на основі інформаційних вікон та „меню” [4,5,77]. Використовується у процесорах з 2-ю арифметикою та засобах індикації цифрових даних [4,5,77].

Недоліки: кодова матриця недостатньо компактна і характеризується високою надлишковістю.

6.3.3. ТЧБ Крейга

Базисні функції Крейга описуються формулою [77,115]

$$\text{Crg}(n, \Theta) = \text{sign}[\sin((2^n - 1) \pi \Theta)]. \quad (6.32)$$

Базис Крейга породжує коди Лібова-Крейга, які мають більш компактну кодову матрицю по відношенню до унітарного базису та базису Хаара.

Базис Крейга використовує пряму та інвертовану кодові матриці Унітарного базису. При цьому упрощується кодування декадних кодів.

Базис Крейга є більш ефективним і компактним у порівнянні з Унітарним базисом та базисом Хаара [1]. Вони є більш розширеними, ніж унітарні. Представлені числом імпульсів та фазою. Використовуються для фазового, фазово-імпульсного кодування. Характеризуються розширеними функціональними можливостями, підвищеною завадозахищеністю. Кодова матриця ТЧБ Крейга має такий вигляд (6.33):

$$\text{Merg} = \begin{array}{cccccc|cc} 1 & 1 & . & . & . & 1 & 0 & & \\ 1 & 1 & . & . & . & 0 & 0 & & \\ . & . & . & . & . & . & . & N & \\ 1 & 1 & . & . & . & 1 & 0 & & \\ . & . & . & . & . & . & . & & \\ 1 & 0 & . & . & . & 0 & 0 & & \end{array} \quad (6.34)$$

де $m=N/2$ -число стовпців; N -число рядків; об'єм матриці: $V = N^2/2$; число активних елементів: $p=N^2/4$ [77,115].

Переваги: використовується у процесорах з 2-ю арифметикою та засобах індикації цифрових даних[4,5,77].

Недоліки:

6.3.4. ТЧБ Радемахера

Функції Радемахера становлять неповну систему ортогональних функцій і є основою для модуляції прямокутних сигналів. Функція Радемахера з індексом m , що позначається $\text{rad}(m,t)$, має вигляд послідовності прямокутних імпульсів, вміщує 2^{m-1} періодів на напіввідкритому інтервалі $[0;1)$ і набуває значень $+1$ чи -1 . Винятком є функція $\text{rad}(0,t)$, що має вигляд одиничного імпульсу. Функції Радемахера періодичні з періодом 1, тобто $\text{rad}(m,t) = \text{rad}(m,t+1)$. Крім того, вони періодичні і на коротших інтервалах: $\text{rad}(m,t+n \cdot 2^{1-m}) = \text{rad}(m,t)$, $m=1,2,\dots$; $n=\pm 1, \pm 2, \dots$. Їх можна дістати за допомогою рекурентного співвідношення:

$$\text{rad}(m,t) = \text{rad}(1, 2^{m-1} t),$$

$$1, t \in [0;1/2),$$

$$\text{де} \quad \text{rad}(1,t) = \begin{cases} 1, t \in [0;1/2), \\ -1, t \in [0;1/2;1) \end{cases} \quad (6.35)$$

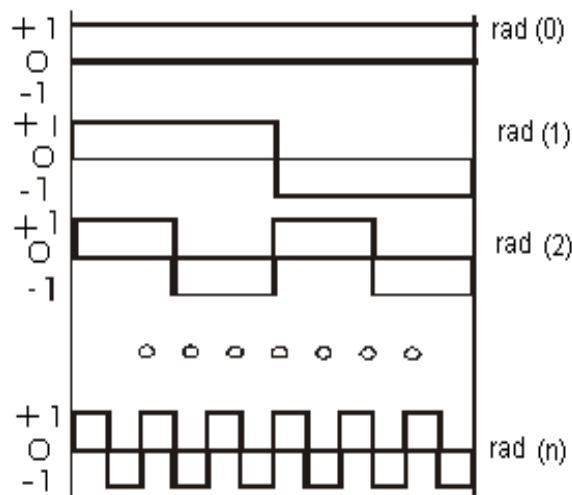


Рис. 6.9 Представлення базису ортогональних функцій Радемахера

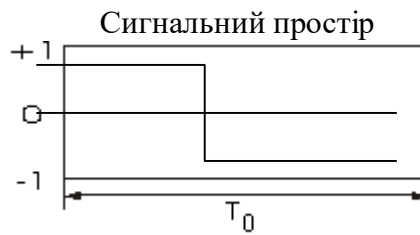


Рис. 6.10 - Представлення сигнального простору в базисі Радемахера.

Базис Радемахера породжує двійкову систему числення і двійкові коди.

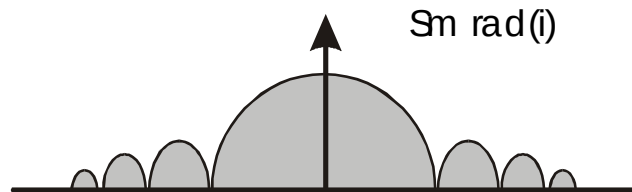


Рис. 6.11 – Спектр базисної функції Радемахера.

Базис Радемахера має ще більш компактну кодову матрицю і породжує двійкову систему числення. В базисі Радемахера працюють всі сучасні комп'ютерні системи [4,5,77].

Кодова матриця ТЧБ Радемахера має вигляд (6.36):

$$M_{rad} = \begin{pmatrix} 0 & 0 & . & . & . & 0 & 0 & 1 \\ 0 & 0 & . & . & . & 0 & 1 & 0 \\ 0 & 0 & . & . & . & 0 & 1 & 1 \\ 0 & 0 & . & . & . & 1 & 0 & 0 \\ 0 & 0 & . & . & . & 1 & 0 & 1 \\ . & . & . & . & . & . & . & . \\ 0 & 1 & . & . & . & 0 & 0 & 0 \\ 0 & 1 & . & . & . & 0 & 0 & 1 \\ 0 & 1 & . & . & . & 0 & 1 & 0 \\ . & . & . & . & . & . & . & . \\ 1 & 1 & . & . & . & 1 & 1 & 1 \end{pmatrix} \quad \begin{matrix} \downarrow \\ N \\ \downarrow \end{matrix} \quad (6.36)$$

$\xrightarrow{m}$

де $m = \log_2 N$ - число стовпців; N - число рядків; об'єм матриці: $V = N \log_2 N$; число активних елементів: $n = N \log_2 N / 2$.

Перевага: – максимальна потужність в сигнальному вікні.

Недоліки: – широка смуга частот сигнальної характеристики, незахищеність від помилок.

6.3.5. ТЧБ Грея

Використовується в кодових масках оптичних перетворювачів кут-код. В кодї Грея відсутній ефект „дребезгу” при переході з однієї кодової позиції а іншу [4,5,77]. Кодова матриця ТЧБ Грея має такий вигляд (6.37)

$$M_{gra} = \begin{pmatrix} 0 & 0 & . & . & . & 0 & 0 & 0 \\ 0 & 0 & . & . & . & 0 & 0 & 1 \\ 0 & 0 & . & . & . & 0 & 1 & 1 \\ 0 & 0 & . & . & . & 0 & 1 & 0 \\ 0 & 0 & . & . & . & 1 & 1 & 0 \\ 0 & 0 & . & . & . & 1 & 1 & 1 \\ 0 & 0 & . & . & . & 1 & 0 & 1 \\ . & . & . & . & . & . & . & . \\ 0 & .1 & . & . & . & 1 & 1 & 1 \\ . & . & . & . & . & . & . & . \\ 1 & 1 & . & . & . & 1 & 1 & 1 \end{pmatrix} \quad N \quad (6.37)$$

ТЧБ Грея аналогічні характеристики базису Радемахера, а кодова матриця відрізняється тільки впорядкуванням кодів.

Переваги: зміна тільки одного біта у впорядкованій кодовій послідовності.

Недоліки: не породжує систему числення.

6.3.6. ТЧБ Крестенсона

Даний базис представлений пилоподібними функціями (6.12).

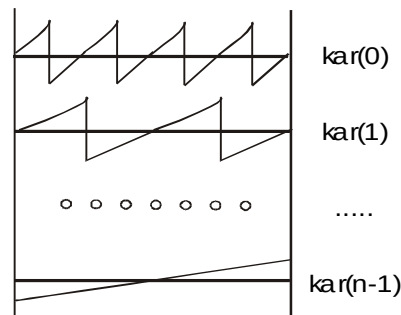


Рис. 6.12 - Представлення базису Крестенсона.

Даний базис породжує систему числення залишкових класів і кодів залишкових класів, породжує амплітудно-частотні методи модуляції.

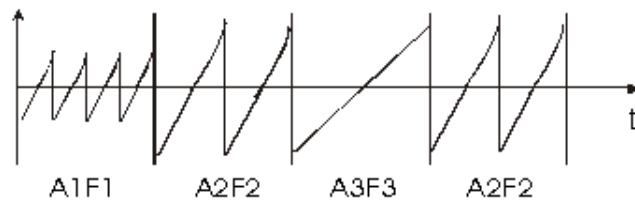


Рис. 6.13 - Модуляція сигналу в базисі Крестенсона.

Спектр сигналів такого базису є експоненціальний:

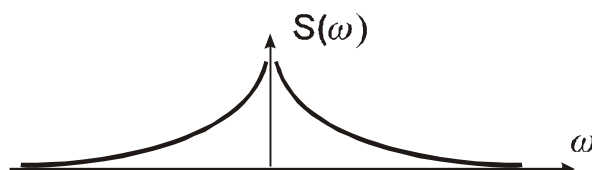


Рис. 6.14 - Спектр сигналу в базисі Крестенсона.

В основу ТЧБ Крестенсона покладена система пілоподібних трикутних ортогональних функцій. Базис Крестенсона породжує непозиційну систему числення залишкових класів (СЗК) [4,5,77].

Кодова матриця ТЧБ Крестенсона має такий вигляд (6.38) :

$$M_{crs} = \begin{matrix} & \begin{matrix} 7 & & & & 3 & & 2 \end{matrix} \\ \begin{matrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ \cdot \\ 1 \end{matrix} & \begin{bmatrix} 0 & 0 & \cdot & \cdot & \cdot & 0 & 0 & 0 \\ 0 & 0 & 1 & \cdot & \cdot & \cdot & 0 & 1 & 1 \\ 0 & 1 & 0 & \cdot & \cdot & \cdot & 1 & 0 & 0 \\ 0 & 1 & 1 & \cdot & \cdot & \cdot & 1 & 1 & 1 \\ 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 & 0 & 0 \\ 1 & 0 & 1 & \cdot & \cdot & \cdot & 0 & 1 & 1 \\ 1 & 1 & 0 & \cdot & \cdot & \cdot & 1 & 0 & 0 \\ 0 & 0 & 0 & \cdot & \cdot & \cdot & 1 & 1 & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & 1 & 0 & \cdot & \cdot & \cdot & 1 & 0 & 1 \end{bmatrix} \end{matrix} \quad \begin{matrix} \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \end{matrix} \quad \begin{matrix} N \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} \quad (6.38)$$

де $m = \log_2 N$ —число стовпців; N —число рядків; об'єм матриці: $V = N \log_2 N$; число активних елементів: $n = N \log_2 N/2$;

Переваги: породжує систему числення залишкових класів та коректуючі коди.

Недоліки: складність виконання операцій порівняння та ділення.

6.3.7. ТЧБ Уолша

У 1926 році Уолш отримав повну систему ортогональних прямокутних функцій, яка доповнює систему функцій Радемахера і відома тепер як система функцій Уолша. Розглянемо означення функцій Уолша, оскільки немає однозначного його тлумачення. Деякі вчені пропонують це зробити перемноженням функцій Радемахера

$$\text{rad}(m, t) = \text{sing}(\sin^2 \pi t). \quad (6.39)$$

Інші вважають це визначення помилковим, оскільки за ним неможливо побудувати систему функцій, упорядковану за Уолшем, і пропонують визначити функції Уолша за допомогою різницевого рівняння

$$\text{wal}(2i+pt) = (-1)^{\text{int}[i/2]} \{ \text{wal}[i, 2(t+\frac{1}{4})] + (-1)^{i+p} \text{wal}[i, 2(t-\frac{1}{4})] \} \quad (6.40)$$

де $p=0$ або $i=0,1,2,\dots$,

$$\text{де } \text{wal}(0, t) = \begin{cases} 1, t \in [-\frac{1}{2}, \frac{1}{2}], \\ 0, t \in [-\frac{1}{2}, \frac{1}{2}], \end{cases}$$

а позначення $\text{int}[i/2]$ використовується для знаходження найбільшого цілого числа, меншого або дорівнює $i/2$. У різницевому рівнянні (6.41) функція $\text{wal}(2i, t)$ має той самий вигляд, що й $\text{wal}(i, t)$,

й відрізняється лише тим, що задана на інтервалі $t \in [-\frac{1}{4}, \frac{1}{4}]$, тобто на інтервалі, довжина

якого у два рази менше за довжину інтервалу аргументу функції $\text{wal}(i, t)$, де $t \in [-\frac{1}{2}, \frac{1}{2}]$ [7].

Повними ортогональними системами базисних кусково-постійних функцій є система Уолша .Функції Уолша утворюються шляхом перемноження функцій Радемахера [4,5,77].

$$\text{Wal}(j, \Theta) = \prod_{v=0}^{n-1} [\text{rad}(n, \Theta)]^{j_v}, \quad (6.42)$$

$$\text{де } j = \sum_{v=0}^{n-1} j_v 2^v = \langle j_{n-1} \dots j_1 j_0 \rangle, j_v \in \{0,1\}.$$

По функціям Уолша можна здійснити розклад довільних сигналів в ряд Уолша-Фур'є, і вони набуватимуть лише два значення (+1,-1), тому зручні для обчислення. Функції Уолша є періодичними з двійково-раціональним періодом, тому їх задають на інтервалі $N = 2^n$, $n=1,2,\dots$,

Найбільше застосування знайшли системи функцій Уолша, відомі як система Пелі, Адамара, і Хармута [4,5,77].

Базис Уолша має матрицю аналогічну базису Хаара . В базисі Уолша містяться всі інші базиси як вибірки функцій Уолша .В базисі Уолша є окремі унікальні функції, які породжують базис Галуа [4,5,77] .

Кодова матриця ТЧБ Уолша має такий вигляд (6.43) :

$$M_{\text{wal}} = \begin{array}{c|cccccc|c} & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ \hline & \xrightarrow{m} & & & & & & \downarrow \end{array} \quad N \quad (6.43)$$

де $m > N$ - число стовпців; N - число рядків; об'єм матриці : $V = m N$;
число активних елементів: $p = m N / 2$.

Число ортогональних функцій Уолша залежить від їх типу та способу впорядкування ,наприклад по Адамару, Уолшу та інш.

Перетворення Уолша-Адамара найбільш відоме серед несинусоїдних ортогональних перетворень. Воно широко застосовується при цифровій обробці сигналів, оскільки може бути обчисленим тільки з використанням додавання і та віднімання [4,5,77].

Переваги: широка сфера застосування в задачах цифрового опрацювання та стиснення зображень.

Недоліки: не породжує систему числення.

6.3.8. ТЧБ Галуа .

Базис Галуа-це є вибірка окремих функцій базису Уолша які характеризуються рекурентними властивостями. Базис Галуа породжує коди поля Галуа і систему числення Галуа. Модуляція йде, як фазова так і частотна. Спектр сигналу більший ніж в базисі Уолша.

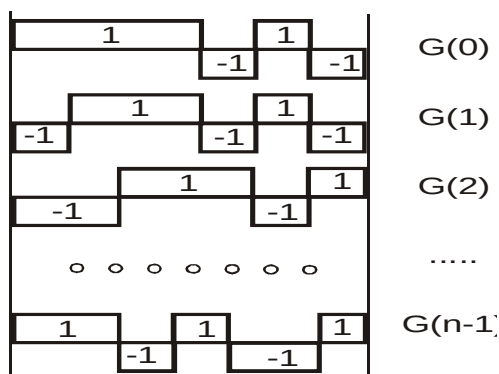


Рис. 6.15- Представлення базису Галуа

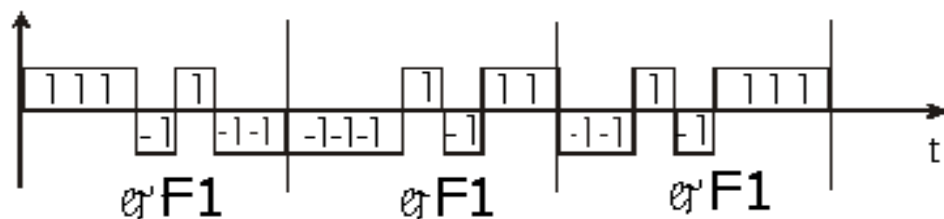


Рис. 6.16 - Модуляція сигналу в базисі Галуа.

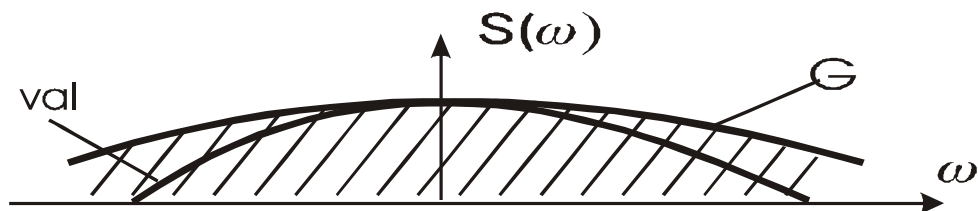


Рис. 6.17 - Спектр сигналу в базисі Галуа.

Базис Галуа є найбільш компактним з відомих ТЧБ. Кодова матриця базису Галуа перетворюється у вектор. Кодування даних в базисі Галуа є рекурентним, і описується теорією полів Галуа. Таким чином ТЧБ Галуа має кодову матрицю, яка перетворюється у вектор. [4,5,77,123].

Кодова матриця-вектор ТЧБ Галуа - M_{Gal} має такий вигляд (6.44);

$$M_{Gal} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ \vdots \\ \vdots \\ \vdots \\ 1 \\ 1 \\ 1 \\ \vdots \\ \vdots \\ 1 \\ 1 \end{pmatrix} \quad \begin{matrix} N \\ \downarrow \end{matrix}$$

$m=1,$

де $m=1$ – число стовпців ; N – однобітових рядків ; об'єм матриці $V = N$; число активних елементів: $n = N/2$.

Переваги: породжує коди поля Галуа та системи числення Галуа.

Недоліки: складність виконання операцій порівняння та модульного множення.

В залежності від вибору формуваців вхідних та вихідних інформаційних сигналів даних – ТЧБ, в табл.6.4 подані символи та аналітичні вирази характеристики інформаційних потоків даних.

Таблиця. 6.4

Системні характеристики теоретико–числових базисів

№п/п	Теоретико–Числові Базиси	m–число стовпців	N– число рядків	V–об’єм матриці	n–число активних елементів
1	Унітарний	m	N	N^2	$N^2/2$
2	Хаара	$m = N$	N	N^2	N
3	Крейга	$N/2$	N	$N^2/4$	$N^2/8$
4	Радемахера	$\log_2 N$	N	$N \log_2 N$	$N/2 \log_2 N$
5	Грея	$\log_2 N$	N	$N \log_2 N$	$N/2 \log_2 N$
6	Крестенсона	$\log_2 N$	N	$N \log_2 N$	$N \log_2 N/2$
7	Уолша	$m > N$	N	mN	$mN/2$
8	Галуа	$m = 1$	N	N	$N/2$

РОЗДІЛ 7

СИСТЕМИ ЧИСЛЕННЯ НА ОСНОВІ СУМ ПОЛІВ ГАЛУА

7.1. Теоретичні основи цілочисельної СЗК ТЧБ Крестенсона.

Найбільш загальною математичною основою ТЧБ є базис прямих ортогональних сум полів Галуа. Даний базис визначається властивостями Китайської теореми про залишки і описується системою дискретних ортогональних функцій Віленкіна-Крестенсона у системі взаємно простих модулів $P_1, P_2, \dots, P_k$. Цей базис породжує систему числення залишкових класів (СЗК), яка представляє коди поля Галуа у k -мірному просторі.

Теоретично основа СЗК глибоко досліджується І.Я. Акушським, В. М. Амербаєвим, Г.Є. Пуховим, В.М. Синьковим, В.А. Торгашевим, Є.І. Брюховичем, М.І. Черв'яковим, А.А. Колядою, а також автором.

Теоретичною базою цілочисельного перетворення СЗК є доведення існування в кільці набору сукупності елементів $u_1, u_2, \dots, u_j, \dots, u_k$, які задовільняють систему Діафантових рівнянь (порівнянь)

$$\left. \begin{aligned} y_j &\equiv 1 \pmod{A_j} \\ y_j &\equiv 0 \pmod{\prod_{i \neq j}^k A_i}, \forall j = \overline{1, k} \end{aligned} \right\},$$

де A_j – символ ідеалу причому

$$A_j + \prod_{i \neq j}^k A_i = A; \forall j = \overline{1, k}.$$

В СЗК елементи Y_j позначається через B_j і називаються ортогональними базисами. При цьому $\{B_j\}$ повинні здовільняти систему порівнянь

$$\left. \begin{aligned} \sum_{j=1}^k B_j &\equiv 1 \pmod{\prod_{j=1}^k P_j}; \\ B_j &\equiv 1 \pmod{P_j} \end{aligned} \right\} \quad (7.1)$$

де P_j – попарно взаємнопрості модулі, звідки слудує, що:

$$\left. \begin{aligned} B_j &\equiv 1 \pmod{\prod_{j=1}^k P_j}; \\ B_j &\equiv A \pmod{P_j}, i \neq j, \forall i = \overline{1, k} \end{aligned} \right\} \quad (7.2)$$

Рішення (7.2) означає, що кожне B_j ділиться без остачі на всі P_i , якщо $i \neq j$ і кожне B_j ділиться без остачі на добуток P_j , тобто

$$B_j = m_j \prod_{i \neq j}^k P_i,$$

де m_j – деяке ціле число у діапазоні $1 \leq m_j \leq P_j - 1$.

При цьому діапазон однозначного представлення чисел N_j у цілочисельних СЗК знаходиться у межах

$$0 \leq N_j \leq \prod_{j=1}^k P_j - 1.$$

Якщо позначити

$$P = \prod_{j=1}^k P_j, \text{ то } B_j = m_j \frac{P}{P_j} \quad (7.3)$$

А m_j може бути знайдено з рішення Діофантового рівняння:

$$m_j \frac{P}{p_j} \equiv 1(\text{mod } P_j). \quad (7.4)$$

Викладенні теоретичні положення визначають пару перетворень цілочисельної СЗК у вигляді зворотнього

$$N_j = \text{res} \sum_{j=1}^k b_j B_j (\text{mod } P) \quad (7.5)$$

та прямого перетворення

$$b_j = \text{res} N_k (\text{mod } P_j); \forall_j = \overline{1, k}, \quad (7.6)$$

де b_j – найменший невід’ємний залишок числа N_j по модулю P_j , тобто $0 \leq b_j \leq P_j - 1$.

Розглянемо приклад:

Нехай задана система взаємно простих модулів: $P_1=3, P_2=5, P_3=7$.

Визначимо діапазон однозначного кодування чисел P згідно виразу (7.3):
 $P = 3 \cdot 5 \cdot 7 = 105$,

та ортогональні базиси B_j

$$B_1 = \frac{105}{3} m_1 \equiv 1(\text{mod } 3); m_1 = 2; B_1 = 70;$$

$$B_2 = \frac{105}{5} m_2 \equiv 1(\text{mod } 5); m_2 = 1; B_2 = 21;$$

$$B_3 = \frac{105}{7} m_3 \equiv 1(\text{mod } 7); m_3 = 1; B_3 = 15;$$

Для перевірки правильності розрахунків задамо: $N_j=1$ тоді $b_1=b_2=b_3=1$ згідно виразу (7.5) отримаємо: $N_j = \text{res}(1 \cdot 70 + 1 \cdot 21 + 1 \cdot 15)(\text{mod } 205) = 1$.

Представимо розглянуту СЗК у вигляді системи ортогональних функцій Віленкіна-Крестенсона (рис. 7.1).

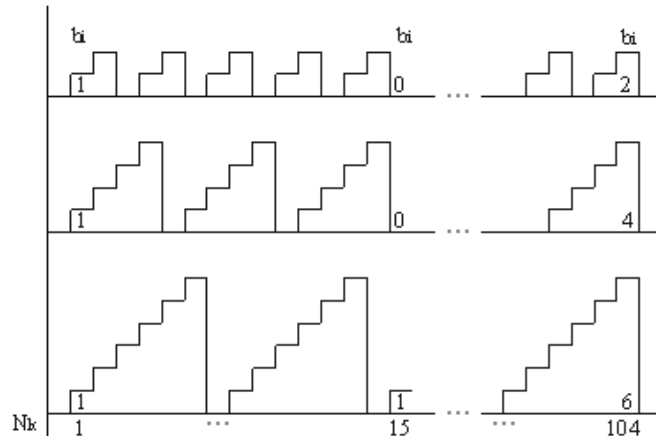


Рис.7.1. Сукупність ортогональних функцій Віленкіна-Крестенсона у системі модулів СЗК:
 $P_1=3; P_2=5; P_3=7$.

З рис.7.1 видно розраховане число $N_1=1$ подано кодом Галуа в СЗК $(1, 1, 1)$ ортогональне базисне число: $B_3 = 15 = (0,0,1)$, а $N_k = P - 1 = 104 = (P_1 - 1, P_2 - 1, P_3 - 1) = (2,4,6)$.

Традиційною галуззю застосування такого класу кодів Галуа та цілочисельної форми СЗК є побудова високопродуктивних процесорів, які працюють у модульній арифметиці СЗК.

Особливістю такої арифметики залишкових класів є відсутність наскрізних переносів, які існують у відомій 2-й системі числення базису Радемахера. Тобто: додавання виконується згідно наступного виразу:

$$\begin{aligned} x &= (a_1, a_2, \dots, a_j, \dots, a_k) \\ y &= (b_1, b_2, \dots, b_j, \dots, b_k) , \\ z &= (c_1, c_2, \dots, c_j, \dots, c_k) \end{aligned} \quad (7.7)$$

де $c_j = \text{res}(a_j + b_j) \bmod P_j, j \in \overline{1, k}$.

Подібно без наскрізних переносів виконується операція множення цілих чисел

$$\begin{aligned} x &= (a_1, a_2, \dots, a_j, \dots, a_k) \\ *y &= (b_1, b_2, \dots, b_j, \dots, b_k) \\ \hline z &= (d_1, d_2, \dots, d_j, \dots, d_k) \end{aligned} \quad (7.8)$$

де $d_j = \text{res}(a_j \cdot b_j) \bmod P_j, j \in \overline{1, k}$.

Наприклад: $x=13; y=7; (P_1, P_2, P_3)=(3, 5, 7)$.

Визначимо коди СЗК чисел x та y згідно (7.6):

$$\begin{aligned} X &\begin{cases} \rightarrow \text{res}13(\bmod 3)=1; a_1=1; \\ \rightarrow \text{res}13(\bmod 5)=3; a_2=3; x=(1, 3, 6). \\ \rightarrow \text{res}13(\bmod 7)=6; a_3=6; \end{cases} \\ Y &\begin{cases} \rightarrow \text{res}7(\bmod 3)=1; b_1=1; \\ \rightarrow \text{res}7(\bmod 5)=2; b_2=2; y=(1, 2, 0). \\ \rightarrow \text{res}7(\bmod 7)=0; b_3=0; \end{cases} \end{aligned}$$

Виконаємо операції додавання та множення в кодах СЗК згідно (7.7) та (7.8):

$$\begin{array}{r} \begin{array}{ccc} P_1 & P_2 & P_3 \\ x=(1, & 3, & 6) \\ +y=(1, & 2, & 0) \\ \hline z=(2, & 0, & 6) \end{array} \quad \begin{array}{ccc} P_1 & P_2 & P_3 \\ x=(1, & 3, & 6) \\ \cdot y=(1, & 2, & 0) \\ \hline z=(1, & 1, & 0) \end{array} \end{array}$$

Перевіряємо результати згідно виразу (7.5):

$$z = (2 \cdot 70 + 0 \cdot 21 + 6 \cdot 15) \bmod 105 = 20;$$

$$D = (1 \cdot 70 + 1 \cdot 21 + 0 \cdot 15) \bmod 105 = 91.$$

В окремих випадках, коли з якихось умов відомо, що $x \geq y$ в СЗК однозначно виконується операція віднімання. Тобто:

$$\begin{array}{r} \begin{array}{ccc} P_1 & P_2 & P_3 \\ x=(1, & 3, & 6) \\ -y=(1, & 2, & 0) \\ \hline V=(0, & 1, & 6) \end{array} \end{array}$$

$$\text{і } x - y = (0 \cdot 70 + 1 \cdot 21 + 6 \cdot 15) \bmod 105 = 6$$

Подібним чином, якщо відомо, що D цілочисельно ділиться на y , то у СЗК можливо виконати операцію ділення шляхом еквівалентного множення згідно виразу:

$$\begin{array}{r} \begin{array}{ccc} D=(d_1, & d_2, & d_3) \\ \cdot Y=(y_1, & y_2, & y_3) \\ \hline X=(a_1, & a_2, & a_3) \end{array} \end{array} \quad (7.9)$$

Наприклад $D = 4 \cdot 11 = 44$;

$$D = 44 = (2, 4, 2) \text{ тоді } y=11=(2, 1, 4)$$

$$D = (2, 4, 2)$$

$$Y = (2, 1, 4)$$

$$x = (1, 4, 4)$$

тобто:

$$P_1(1 \cdot 2) \bmod 3 = 2;$$

$$P_2(4 \cdot 1) \bmod 5 = 4;$$

$$P_3(4 \cdot 4) \bmod 7 = 4.$$

Використання особливих наборів модулів цілочисельного СЗК у діапазоні кодування чисел Ферма та Мерсена забезпечили ефективну реалізацію швидкого перетворення Фур'є-Галуа і відповідне ефективне застосування ТЧБ в галузі цифрової фільтрації сигналів.

Автором розширена сфера застосування СЗК для формування зменшення надлишковості інформації, підвищення завадозахищеності передавання даних та побудови високопродуктивних процесорів стосовно задач низових комп'ютерних систем (НКС). Результатом теоретичних досліджень у цій галузі є розробка сукупності нових методів кодування повідомлень, які передаються по каналах зв'язку та вводяться в комп'ютери.

7.2. Кодування інформаційних потоків в СЗК з довільним порядком реєстрації даних.

Суть даного метода полягає у тому, що кожен цифровий відлік Y_{ij} , i -го виміру j -го каналу множать на базисне число B_j і створюють нове значення $a_{ij} = Y_{ij} \cdot B_j$. Числа a_{ij} сумують по модулю P і фіксують у вигляді числа

$$N_{ik} = \text{res} \sum_{j=1}^k a_{ij} \pmod{P},$$

де k - число каналів.

Таким чином N_k однозначно виражає набір цифрових відліків $Y_{i1}, Y_{i2}, \dots, Y_{ij}, \dots, Y_{ik}$. При цьому однозначність забезпечується коли $0 \leq Y_{ij} = A_j \leq P_j - 1$, де: A_j - діапазон квантування цифрових відліків у j -му каналі, P_j - модулі перетворення СЗК.

Однозначне відновлення відліків Y_{ij} після передавання або збереження коду N_{ik} виконується на основі прямого перетворення в СЗК.

$$Y_{ij} = \text{res} N_{ik} \pmod{P_j}.$$

Коефіцієнт стиснення інформації згідно оцінки міри Хартлі визначається згідно відношення ентропії вхідних даних, де код кожного цифрового відліку з довільним порядком реєстрації супроводжується ідентифікаційним кодом номера каналу $k_c = \frac{I_{ex}}{I_{СЗК}}$.

Тобто:

$$I_{\hat{a}\hat{o}} = \sum_{j=1}^k \hat{E}[\log_2 A_j] + k \hat{E}[\log_2 k];$$

$$I_{\hat{a}\hat{o}\hat{o}} = \hat{A}[\log_2 (P-1)].$$

Чисельне моделювання описаного методу пакування даних на основі кодів Галуа цілочисельної СЗК показало, що значення.

$$\sum_{j=1}^k \hat{E}[\log_2 A_j] \approx \hat{E}[\log_2 (P-1)] \pm 1.$$

Таким чином ефект стиснення даних визначається згідно виразу:

$$k_c = 1 + k \hat{E}[\log_2 k] / \hat{E}[\log_2 (P-1)].$$

При зростанні числа каналів оцінка коефіцієнта стиснення інформації асимптотично наближається до величини 1,8.

Наприклад:

$$0 \leq Y_{ij} \leq 10; j \in 4, \text{ тоді } P_j = (10, 11, 12, 13) \text{ і } P-1 = 17159$$

$$k_c = 1 + 4 \cdot \frac{2}{15} = 1 + 0,53 \approx 1,5.$$

На основі аналізу характеристик та функціональних можливостей використання ТЧБ для формування структуризованих даних (СД) на низових рівнях РКС розроблений метод компактного завадозахищеного формування СД на основі ТЧБ Крестенсона, який описується наступним алгоритмом:

$$\left. \begin{array}{l} x_1(t) \rightarrow x_{i1} \rightarrow p_1 \rightarrow b_1 \\ x_2(t) \rightarrow x_{i2} \rightarrow p_2 \rightarrow b_2 \\ \dots \\ x_j(t) \rightarrow x_{ij} \rightarrow p_j \rightarrow b_j \\ \dots \\ x_m(t) \rightarrow x_{im} \rightarrow p_{k-1} \rightarrow b_{k-1} \\ D_i \rightarrow p_k \rightarrow b_k \end{array} \right\} N_k = \text{res} \sum_{j=1}^k b_j B_j \pmod{P} \quad (7.10)$$

$$B_j = \frac{P}{p_j} m_j = 1 \pmod{P_j},$$

де $x_j(t)$ - аналогові дані телеметрії;

x_{ij} - цифрові дані телеметрії;

D_i - техніко-економічні дані;

$p_1, p_2, \dots, p_k$ - система взаємно простих модулів;

$b_1, b_2, \dots, b_j, \dots, b_k$ - набір найменших невід'ємних залишків;

B_j - система ортогональних базисів СЗК.

7.3. Каскадне кодування даних на основі методу залишків та СЗК

Процеси кодування та декодування інформації методом залишків виконуються наступним чином. Вхідний масив даних на деякому інтервалі часу описується матрицею

$$Y_{ij} = \begin{bmatrix} Y_{11} & \dots & Y_{i1} & \dots & Y_{k1} & \dots & Y_{n1} \\ Y_{12} & \dots & Y_{i2} & \dots & Y_{k2} & \dots & Y_{n2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ Y_{1k} & \dots & Y_{ik} & \dots & Y_{kk} & \dots & Y_{nk} \end{bmatrix}, \quad (7.11)$$

де Y_{ij} - цифровий відлік i -го вимірювання j -го каналу n , k - число відліків ($n > k$).

Застосувавши метод залишків, матрицю (7.11) перетворюємо у матрицю залишків $\|b_{ij}\|$ шляхом нелінійної модульної операції в коді поля Галуа.

$$\|b_{ij}\| = \text{res} Y_{ij} \pmod{P_j}, i \in \overline{1, n}; j \in \overline{1, k}, \quad (7.12)$$

де $P_j - 1 \geq 2|Y_{ij} - Y_{i-1j}| \max$ - є умовою однозначності перетворень.

Вираз (7.12) відповідає лінійній формі

$$Y_{ij} = a_{ij} \cdot P_j + b_{ij},$$

де $a_{ij} = \overset{\vee}{E} \left[\frac{Y_{ij}}{P_j} \right]$ - ранг відліку Y_{ij} по модулю P_j .

Тому для однозначного відновлення цифрових відліків матриці (7.11) на основі рекурентності формули

$$Y_{i+1,j} = \hat{E} \left[\frac{Y_{ij} - b_{i+1,j}}{P_j} \right] P_j + b_{ij} \quad (7.13)$$

необхідно матрицю залишків (7.12) доповнити вектором-строкою опорних рангів $a_{ij} (i \in \overline{1, k})$ і елементом X ,

$$\begin{pmatrix} b_{11} & \dots & b_{i1} & \dots & b_{k1} & b_{k+1,1} & \dots & b_{n1} \\ b_{12} & \dots & b_{i2} & \dots & b_{k2} & b_{k+1,2} & \dots & b_{n2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ b_{1j} & \dots & b_{ij} & \dots & b_{kj} & b_{k+1,j} & \dots & b_{nj} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ b_{1k} & \dots & b_{ik} & \dots & b_{kk} & b_{k+1,k} & \dots & b_{nk} \\ a_{1k+1} & \dots & a_{ik+1} & \dots & b_{k,k+1} & X_{k+1} & \dots & X \end{pmatrix}, \quad (7.14)$$

де елементи $0 \leq X \leq P_{k+1} - 1$ утворюють поле довільних інформаційних повідомлень по mod P_{k+1} .

Таким чином, однозначність перетворень матриць (7.14) та (7.11) забезпечується в розширеній СЗК по модулю P_{k+1} . Тоді, застосувавши зворотнє перетворення СЗК у розширеній системі модулів $P = P_1 \cdot P_2 \cdot \dots \cdot P_k \cdot P_{k+1}$, отримаємо

$$N_{i,k+1} = \text{res} \sum_{j=1}^{k+1} C_{ij} \cdot B'_j (\text{mod } P),$$

де C_{ij} елементи векторів стовбців

$$C_{il} = \begin{pmatrix} b_{i1} \\ \dots \\ b_{ik} \\ a_{i,k+1} \end{pmatrix}, \quad 1 \leq i \leq k; \quad C_{il} = \begin{pmatrix} b_{i1} \\ \dots \\ b_{ik} \\ X \end{pmatrix}, \quad k+1 \leq i \leq n;$$

B_{ij} – ортогональні бази розширеної СЗК.

Структура даного перетворення СЗК має наступний вигляд:

$$\left. \begin{array}{l} x_1(t) \rightarrow x_{i1} \rightarrow b_{i1}(\text{mod } q_1) \rightarrow p_1 \rightarrow b_1 \\ x_2(t) \rightarrow x_{i2} \rightarrow b_{i2}(\text{mod } q_2) \rightarrow p_2 \rightarrow b_2 \\ \dots \\ x_j(t) \rightarrow x_{ij} \rightarrow b_{ij}(\text{mod } q_j) \rightarrow p_j \rightarrow b_j \\ \dots \\ x_{k-1}(t) \rightarrow x_{ik-1} \rightarrow b_{ik-1}(\text{mod } q_{k-1}) \rightarrow p_{k-1} \rightarrow b_{k-1} \\ D_k \rightarrow p_k \rightarrow b_k \end{array} \right\} N_k = \text{res} \sum_{j=1}^k b_j B_j (\text{mod } P),$$

де D_k - службові алфавітно-цифрові дані

Для однозначності перетворення (7.15) діапазони квантування відліків A_j по каналах розраховуються за формулою:

$$A_j = P_j \cdot P_{k+1} - 1.$$

Для захисту кодових слів $N_{i,k+1}$ від помилок необхідно введення деякої надлишковості шляхом застосування нерозділених арифметичних AN -кодів, що дозволяє ефективно виявляти та виправляти помилки при відносно короткій їх двійковій довжині 16 - 64 біт.

Тому, розширивши набір модулів

$$\rho' = P_0 \cdot P_1 \cdot P_2 \cdot \dots \cdot P_{k+1},$$

та, визначивши новий набір базисних чисел $\{B_i''\}$ і доповнивши матрицю (9.14) нульовим стовбцем і строкою, отримаємо:

$$\begin{pmatrix} b_{0,0} & b_{1,0} & \dots & b_{k,0} & b_{k+1,0} & \dots & b_{n,0} \\ b_{1,0} & b_{1,1} & \dots & b_{k,1} & b_{k+1,1} & \dots & b_{n,1} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{0,j} & b_{0,j} & \dots & b_{0,j} & b_{k+1,j} & \dots & b_{n,j} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{0,k} & b_{1,k} & \dots & b_{k,k} & b_{k+1,k} & \dots & b_{n,k} \\ X_0 & a_1 & \dots & a_k & X_{k+1} & \dots & X_n \end{pmatrix}. \quad (7.15)$$

Описаний метод реалізований в інформаційній системі контролю процесів буріння (комплекси СКУБ та АТОС-Б), які серійно випускалися заводом «Промприлад» (м. Івано-Франківськ).

В названих системах вектор стовбець $b_{0,0}, \dots, b_{0,k}$ - використовується для ідентифікації блоку даних матриці залишків $b_{i,j}$ та стартової інформації. Символу X_0 присвоюється код стану об'єкта управління $X_0 = S_i$, параметри якого описуються блоком даних (7.15). Вектор-строка є нульовим, тобто всі значення $b_{i,0} = 0$, що забезпечує захист даних від помилок. Символи $X_{k+1}, \dots, X_n$ використані для кодування даних, які вводяться оператором з пульта і кодуються півбайтами (4 біти) по модулю $P_{k+1} = 16$.

Наприклад у процесорі СЗК терміналу АТОС-Б реалізований набір модулів для двохбайтових слів $N_{i,S}$ ($P_0 = 5, P_1 = 7, P_2 = 9, P_3 = 13, P_4 = 16$), добуток яких $\rho'' = 2^{16} - 5$, а діапазони квантування цифрових даних відповідно рівні:

$$A_1 = 7 \cdot 16 - 1 = 111, \delta_1 < 1\%$$

$$A_2 = 9 \cdot 16 - 1 = 143, \delta_2 < 1\%$$

$$A_3 = 13 \cdot 16 - 1 = 207, \delta_3 < 0,5\%,$$

де δ_i - похибка вимірювання технологічного параметра.

Застосування модуля $P_{k+1} = 16$ при кодуванні $N_{i,k+1}$ у двійковій системі числення дозволяє виділяти службову інформацію по $\text{mod } P_0$ без декодування чисел $N_{i,k+1}$ шляхом вилучення 4-х молодших бітів. Тобто:

$$b_{i,k} = (a_3 \cdot 2^3 + a_2 \cdot 2^2 + a_1 \cdot 2^1 + a_0 \cdot 2^0)_N; a_i \in \overline{0,1}.$$

На рис.7.2 та 7.3 відповідно показана структура вихідного коду та зовнішній вигляд реалізованого в промисловості 32-канального процесора СЗК.

P_0	0	0	2	2	2	2	0	0	0	...	0	2	2	2	2	0	0	0	...
P_1	b_1	b_1	b_1	b_1	b_1	b_1	b_1	b_1	b_1	...	b_1	b_1	b_1	b_1	b_1	b_1	b_1	b_1	...
P_2	b_2	b_2	b_2	b_2	b_2	b_2	b_2	b_2	b_2	...	b_2	b_2	b_2	b_2	b_2	b_2	b_2	b_2	...
P_3	b_3	b_3	b_3	b_3	b_3	b_3	b_3	b_3	b_3	...	b_3	b_3	b_3	b_3	b_3	b_3	b_3	b_3	...
P_4	$\times$	$\times$	R_3	R_2	R_1	C_1	$\times$	$\times$	$\times$	...	$\times$	R_3	R_2	R_1	C_1	$\times$	$\times$	$\times$	...
P_0	1	1	3	3	3	3	1	1	1	...	1	3	3	3	3	1	1	1	...
P_1	b_4	b_4	b_4	b_4	b_4	b_4	b_4	b_4	b_4	...	b_4	b_4	b_4	b_4	b_4	b_4	b_4	b_4	...
P_2	b_5	b_5	b_5	b_5	b_5	b_5	b_5	b_5	b_5	...	b_5	b_5	b_5	b_5	b_5	b_5	b_5	b_5	...
P_3	b_6	b_6	b_6	b_6	b_6	b_6	b_6	b_6	b_6	...	b_6	b_6	b_6	b_6	b_6	b_6	b_6	b_6	...
P_4	$\times$	$\times$	R_6	R_5	R_4	C_2	$\times$	$\times$	$\times$	...	$\times$	R_6	R_5	R_4	C_2	$\times$	$\times$	$\times$	...

Рис.7.2 Структура вихідного коду процесора СЗК терміналу АТОС-Б.

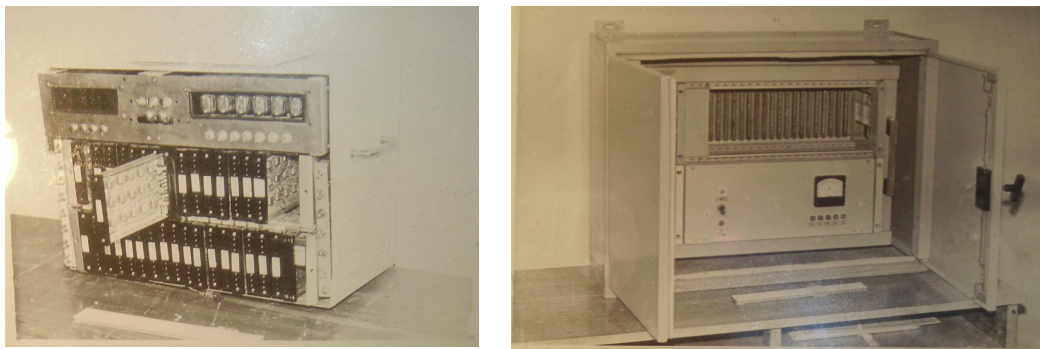


Рис.7.3 Зовнішній вигляд дослідного взірця 32-бітного процесора СЗК.

Позитивними якостями викладених методів пакування даних на основі сум полів Галуа та СЗК є наступні:

- строга зворотність та однозначність перетворення інформаційних повідомлень;
- неадаптивність процедур перетворення;
- рівномірність швидкості інформаційних потоків на виході кодера;
- формування байторієнтованих інформаційних даних;
- структуризованість блоків даних та їх захист від помилок;
- зменшення надлишковості інформації.

До недоліків слід віднести деяку громіздкість зворотного перетворення СЗК у цілочисельній формі згідно виразу (7.5), що включає операції множення, ділення та потребує розрахунку нормуючих вагових коефіцієнтів m_i .

7.4. Нормалізована форма СЗК.

Наявність операції згортки N_k по модулю P_j у виразі (7.6) при зростанні відношення ρ/P_j у великорозрядних процесорах знижує можливості створення високопродуктивних програмно-апаратних засобів опрацювання даних на основі теорії кодів поля Галуа. Аналогічно необхідність визначення залишку по модулю ρ у зворотному перетворенні цілочисельної форми СЗК від суми добутків $b_j \cdot V_j$ ускладнює реалізацію суматорів та знижує їх швидкодію. Цілочисельність подання даних також не завжди вигідна, оскільки числа в процесорах, як правило, кодуються у нормалізованій формі з плаваючою або фіксованою комою. Тому одним з шляхів оптимізації та підвищення швидкодії процесорів СЗК є приведення її перетворень до нормалізованої форми.

Умову нормування числа $N_{i,k}$ отримаємо розділивши ліву і праву частини виразу (7.5) на ρ

$$\frac{N_k}{\rho} = \text{res} \sum_{j=1}^k b_j \frac{\rho}{P_j} \cdot m_j (\text{mod } \rho) / \rho.$$

У результаті отримаємо

$$[N_k]_0 = \text{res} \sum_{j=1}^k [b_j]_0 \cdot m_j (\text{mod } 1), \quad (7.16)$$

де $[N_k]_0 = N_k / \rho$; $[b_j]_0 = b_j / P_j$; операція $\text{mod } 1$ - виконується шляхом відкидання цілої частини результату сумування добутків $[b_j] \cdot m_j$.

Враховуючи, що нормування по модулю P_j значення залишку b_j отримується при вирішенні рівняння:

$$[b_j]_0 = \text{res}[N_k]_0 \frac{\rho}{P_j} \pmod{1}. \quad (7.17)$$

Відновлення цілочисельного значення b_j виконується згідно виразу

$$b_j = \hat{E}([b_j]_0 \cdot P_j), \quad (7.18)$$

$\hat{E}[\cdot]$ - цілочисельна функція з округленням до більшого цілого.

Наявність цілочисельної функції $\hat{E}[\cdot]$ у виразі (7.17) обумовлена тим, що відношення b_j/P_j , як правило, є раціональним або безконечним дробовим числом, яке задається скінченним числом розрядів з деякою від'ємною похибкою, яка викликає похибку обчислення b_j .

Для оцінки допустимої величини вказаної похибки по кожному модулю нормалізованої СЗК (НСЗК) запишемо відношення b_j/P_j у вигляді:

$$\frac{b_j}{P_j} = A_j + \delta_j, \quad (7.19)$$

де A_j - округлене цілочисельне значення відношення b_j/P_j ;

δ_j - похибка округлення.

Підставивши (7.19) у (7.16), отримаємо

$$[N_k]_0 = \text{res}\left(\sum_{j=1}^k b_j \cdot A_j + \sum_{j=1}^k b_j \cdot \delta_j\right) \pmod{1},$$

де сума, яка містить елементи δ_j , визначає загальну похибку обчислення $[N_k]_0$.

Згідно (9.17) N_k нормується по ρ з точністю до ρ^{-1} , тому однозначність перетворення інформації в НСЗК досягається, якщо величина похибки обчислення $[N_k]_0$ задовольняє умову

$$\left| \sum_{j=1}^{k-v} b_j \cdot \delta_j \right| < 0.5\rho; \quad v \in \overline{1, k-1}. \quad (7.20)$$

Умова (7.20) дозволяє сформулювати задачу оцінки похибки округлення b_j/P_j у вигляді системи лінійних нерівностей. При цьому необхідно враховувати граничний випадок коли $b_j = P_j - 1$. З врахуванням останнього система вказаних нерівностей набуває вигляду:

$$\left. \begin{aligned} &|\rho(P_1 - 1)\delta_1| < 0,5 \\ &|\rho(P_2 - 1)\delta_2| < 0,5 \\ &\dots \quad \dots \quad \dots \\ &|\rho(P_k - 1)\delta_k| < 0,5 \end{aligned} \right\}; \quad 1)$$

$$\left. \begin{aligned} &|\rho[(P_1 - 1)\delta_1 + (P_2 - 1)\delta_2]| < 0,5 \\ &|\rho[(P_1 - 1)\delta_1 + (P_3 - 1)\delta_3]| < 0,5 \\ &\dots \quad \dots \quad \dots \\ &|\rho[(P_1 - 1)\delta_1 + (P_k - 1)\delta_k]| < 0,5 \end{aligned} \right\}; \quad 2)$$

$$|\rho[(P_1 - 1)\delta_1 + (P_2 - 1)\delta_2 + \dots + (P_k - 1)\delta_k]| < 0,5. \quad \dots \quad k)$$

Рішення отриманої системи нерівностей дозволяє визначити допустимі похибки подання парних добутоків, при яких забезпечується однозначність перетворення НСЗК.

Враховуючи, що кількість рівнянь в розглянутій системі зростає (по закону 2^{k-1}), при збільшенні числа модулів кодування її рішення у загальному вигляді для різних наборів модулів порівняно з великим об'ємом обчислень.

Розбиваючи систему (7.21) на окремі групи: 1), 2),... k), можна оцінити величину допустимої похибки δ_j по кожному модулю, їх парах і т.д., використовуючи методи лінійної алгебри. Але, подання $[N_k]_0$ з похибкою, яка отримує як додатне, так і від'ємне значення, нераціонально з точки зору розшифрування інформації програмованими обчислювальними засобами. Наявність у системі команд сучасних процесорів операцій виділення цілої та дробової частин результатів обчислень, наприклад, цілочисельної функції з округленням до меншого цілого

$$A_j = \check{E}[x_j]; x_j = A_j + \delta_j; 0 \leq \delta_j < 1,$$

а також велика швидкодія їх виконання визначає доцільність подання $[N_k]_0$ та $[b_j]_0$ тільки з додатною похибкою заокруглення. Це у свою чергу створює умови заміни процедур згортки по модулю ρ у перетворенні СЗК на процедури множення та виділення цілої частини результатів перетворення НСЗК.

У цьому випадку рішення системи (7.20) суттєво спрощується і зводиться до розв'язання однієї нерівності:

$$\rho[(P_1 - 1)\delta_1 + (P_2 - 1)\delta_2 + \dots + (P_k - 1)\delta_k] < 1. \quad (7.21)$$

При апаратурній або програмній реалізації перетворення НСЗК часто потрібна оцінка кількості необхідних двійкових розрядів відношення $\frac{b_j}{P_j}$.

Таку оцінку знизу можна отримати, розв'язуючи першу групу системи нерівностей (7.21) відносно додатної помилки округлення $[b_j]_0$ по кожному модулю P_j .

$$\delta_{j \max} < [\rho(P_j - 1)]^{-1}. \quad (7.22)$$

Перетворивши (7.22) шляхом логарифмування за основою 2 отримаємо:

$$\delta_{j \max} < 2^{\hat{E}[\log_2 \rho(P_j - 1)]}, \quad (7.23)$$

де $\hat{E}[\cdot]$ - необхідна кількість двійкових розрядів дробового числа $\frac{b_j}{P_j}$ округленого в сторону більшого.

На практиці, при створенні процесорів, які реалізують цифрові перетворення розрядність подання даних по кожному модулю P_j вибирається однаковою. Тоді $\delta_1 = \delta_2 = \dots = \delta_k$, і рішення нерівності (7.21) отримує вигляд:

$$\delta_{\text{дон}} < [\rho \sum_{j=1}^k (P_j - 1)],$$

або після логарифмування

$$\delta_{\text{дон}} < 2^{\hat{E}[\rho \sum_{j=1}^k (P_j - 1)]}, \quad (7.24)$$

$\delta_{\text{дон}}$ - допустима системна похибка округлення по кожному з модулів P_j .

У відповідності з вищевикладеним пряме та зворотнє перетворення НСЗК можна записати у вигляді

$$b_j = \overset{\vee}{E} \left[\text{res}[N_k]_0 \cdot \overset{\circ}{B}_j \pmod{1} \cdot P_j \right];$$

$$[N_k]_0 = \text{res} \sum [b_j]_0 \cdot m_j \pmod{1},$$

$$\text{де } \overset{\circ}{B}_j = \begin{cases} \rho/P_j, & \text{якщо } m_j = 1 \\ -\rho/P_j, & \text{якщо } m_j = P_j - 1. \end{cases}$$

В останніх виразах операція згортки по модулю одиниця еквівалентна виділенню дробової частини результату обчислень, $\overset{\vee}{E}[\cdot]$ - відповідно виділенню цілої частини, а від'ємне значення базисного числа $\overset{\circ}{B}_j$ відповідає реалізації зворотнього перетворення НСЗК на основі найменших залишків, які можуть приймати від'ємні значення. Тобто $-\frac{P_j}{2} \leq \left[\overset{\circ}{b}_j \right] \leq \frac{P_j}{2}$.

Так як величина $\delta_{\text{дон}}$ задовольняє будь якому відношенню $\frac{b_j}{P_j}$, $a \leq b_j < P_j - 1$, $1 \leq m_j < P_j - 1$, то алгоритм прямого та зворотнього перетворень НСЗК отримують вид:

$$[b_k]_P = \text{res}[N_k]_P \cdot \overset{\circ}{B}_j \pmod{1};$$

$$[N_k]_P = \text{res} \sum_{j=1}^k [b_j]_P \cdot m_j \pmod{1};$$

або у вигляді:

$$[N_k]_P = \dots \sum_{j=1}^k [b_j]_P \cdot m_j;$$

$$[b_k]_P = \dots \frac{b_j}{P_j};$$

$$[b_k]_P = \dots [N_k]_P \cdot \overset{\circ}{B}_j. \quad (7.25)$$

Структура перетворення СЗК в нормалізованій формі має наступний вигляд:

$$\left. \begin{array}{l} x_1(t) \rightarrow x_{i1} \rightarrow p_1 \rightarrow [b_1]_0 \\ x_2(t) \rightarrow x_{i2} \rightarrow p_2 \rightarrow [b_2]_0 \\ \dots \\ x_j(t) \rightarrow x_{ij} \rightarrow p_j \rightarrow [b_j]_0 \\ \dots \\ x_m(t) \rightarrow x_{im} \rightarrow p_{k-1} \rightarrow [b_{k-1}]_0 \\ D_i \rightarrow p_k \rightarrow [b_k]_0 \end{array} \right\} [N_k]_0 = \text{res} \sum_{i=1}^k [b_i]_0 \pmod{1}; \quad (7.26)$$

$$b_i = \text{int} \text{res}[N_k]_0 \pmod{1} \cdot P,$$

де int - символ виділення цілої частини результату обчислень.

Позитивною якістю отриманих виразів НСЗК є подання $[N_k]_0$ та $[b_j]_0$, при виконанні умови (7.24), у формі нормалізованих двійкових чисел з фіксованою комою, що забезпечує високу степінь сумісності кодування даних та виконання перетворень НСЗК на універсальних та спеціалізованих процесорах.

Розглянемо приклади.

Нехай задано систему модулів НСЗК $P_1 = 3, P_2 = 5, P_3 = 7$, для якої $\rho = 105$; $m_1 = 2$; $m_2 = 1$; $m_3 = 1$.

Визначимо допустиму похибку округлення перетворень НСЗК

$$\delta_{\text{дон}} = \rho^{-1} = \frac{1}{105} = 0,00952_{(10)} = 0,00000010_{(2)}$$

Обчислимо таблиці нормалізованих значень $[b_j]_p$.

а) у десятковій системі числення:

	$P_1 = 3$	$P_2 = 5$	$P_3 = 7$
b_j			
0	0,00	0,00	0,00
1	0,33	0,20	0,143
2	0,66	0,40	0,286
3	-	0,60	0,428
4	-	0,80	0,571
5	-	-	0,714
6	-	-	0,857

б) у двійковій системі числення:

	$P_1 = 3$	$P_2 = 5$	$P_3 = 7$
b_j			
0	0,00000000	0,00000000	0,00000000
1	0,01010101	0,00110011	0,00011111
2	0,10101011	0,01100110	0,00111110
3	-	0,10011001	0,01101101
4	-	0,11001100	0,01111101
5	-	-	0,01010111
6	-	-	0,11011011

Задано $N_k = 1$, тоді $b_1 = b_2 = b_3 = 1$ і $[N_k]_0 = 0,00000010_{(2)}$.

Виконаємо зворотнє перетворення НСЗК згідно виразу (9.25):

$$[N_k] = (0,33 \cdot 2 + 0,2 \cdot 1 + 0,143 \cdot 1) \bmod 1 = 0,00966$$

звідки

$$N_k = \check{E}[\rho \cdot [N_k]_0] = \check{E}[105 \cdot 0,00966] = 1.$$

Аналогічно виконуємо перетворення НСЗК у двійковій системі числення, тобто:

$$[b_1]_0 = 0,01010101_{(2)};$$

$$[b_2]_0 = 0,00110011_{(2)};$$

$$[b_3]_0 = 0,00100100_{(2)},$$

звідки

$$\begin{aligned}
[N_k]_0 &= (2 \cdot [b_1]_0 + 1 \cdot [b_2]_0 + 1 \cdot [b_3]_0) \bmod 1 = \\
&0,10101010 \\
+ &0,00110011 = 0,00000011_{(2)} \\
+ &0,00100100 \\
\hline
&1,00000011
\end{aligned}$$

Отже,

$$N_k = E[\rho \cdot [N_k]_0] = E[105_{(10)} \cdot [N_k]_0]$$

Оскільки $N_k = 105_{(10)} = 1101001_{(2)}$, отримаємо N_k перемноживши його у двійковій системі числення на $[N_k]_0$

$$\begin{aligned}
&001101001 \\
\times &0,00000010, \\
\hline
&001,101010
\end{aligned}$$

що відповідає дійсному значенню $N_k = E[1,101010] = 1$.

Допустимо: $b_1 = 2$; $b_2 = 4$; $b_{31} = 6$, тобто

$$N_k = (2, 4, 6) = \rho - 1 = 104_{(10)}.$$

Виконаємо пряме і зворотнє перетворення НСЗК наступним чином:

$$b_1 = 2/3 = 0,66_{(10)} = 0,10101010_{(2)};$$

$$b_2 = 4/5 = 0,8_{(10)} = 0,11001100_{(2)};$$

$$b_3 = 6/7 = 0,85_{(10)} = 0,11011011_{(2)},$$

$$\begin{aligned}
[N_k]_0 &= (2 \cdot 0,10101010 + 1 \cdot 0,11001100 + 1 \cdot 0,11011011) \bmod 1 = \\
&0,01010101 \\
+ &0,11001100 \\
= + &0,11011011 = 0,11110100_{(2)}. \\
\hline
&1,11110100
\end{aligned}$$

Обчислимо N_k

$$N_k = E[105_{(10)} \cdot [N_k]_0] = 1100100_{(2)}.$$

Виконаємо перевірку обчислень шляхом множення двійкових кодів:

$$\begin{aligned}
&0,11110100 \\
\times &1101001 \\
\hline
&11110100 \\
+ &11110100 \\
&11110100 \\
&11110100 \\
\hline
&1100100,00010100
\end{aligned}$$

$$1100100_{(2)} = 104_{(10)} = N_k = \rho - 1.$$

Таким чином однозначність перетворень НСЗК доведена.

7.5. Досконалі форми СЗК.

Основною операцією, яка збільшує часову складність алгоритмів перетворень в цілочисельній та нормалізованій СЗК, є процедура множення при отриманні парних добутків $b_j \cdot B_j$ та $[b_j] \cdot m_j$. Тому її виключення з аналітики названих перетворень дозволяє добитися суттєвих спрощень, як в алгоритмічному, так і в апаратному аспектах.

Для рішення цієї задачі запишемо вираз для зворотнього перетворення СЗК з одиничними ваговими коефіцієнтами

$$N_k = \begin{cases} \text{res} \sum_{j=1}^k b_j (\pm 1) \frac{\rho}{P_j} (\text{mod } \rho), & \dot{N}_k \leq 0; \\ \rho - \text{res} \sum_{j=1}^k b_j (\pm 1) \frac{\rho}{P_j} (\text{mod } \rho), & \dot{N}_k > 0. \end{cases} \quad (7.27)$$

Зворотнє перетворення для таких СЗК у випадку (7.27) визначається виразом

$$\pm \dot{N}_k = \text{res} \sum_{j=1}^k b_j (\pm 1) \frac{\rho}{P_j} (\text{mod } \rho), \quad (7.28)$$

а пряме перетворення виконується згідно виразу:

$$b_j = \begin{cases} \text{res} \dot{N}_k (\text{mod } P_j), & N_k \leq 0; \\ \text{res}(\rho - \dot{N}_k) (\text{mod } P_j), & N_k > 0. \end{cases}$$

Умови (7.27) та (7.28) дозволяють сформулювати вимоги до наборів модулів СЗК, в яких виключається надлишковість подання $b_j \cdot B_j$ при заданих $\{m_j\}$, тобто:

$$\begin{aligned} 0) \{m_j\} &= \{1, 1, \dots, 1\}; \quad \{P_j > m_j\}; \\ 1) \{m_j\} &= \{1, 1, \dots, P_k - 1\}; \quad P_k = m_k + 1; \\ 2) \{m_j\} &= \{1, 1, \dots, P_{k-1} - 1, P_k - 1\}; \quad P_{k-1} = m_{k-1} + 1; P_k = m_k + 1; \\ &\dots \dots \dots \\ k) \{m_j\} &= \{P_j - 1\}; \quad P_j = m_j + 1. \end{aligned} \quad (7.29)$$

Розглянемо випадок (7.29.0).

Підставивши $m_j = 1$ у вираз для B_j (7.16) визначимо умову існування СЗК з набором m_j (7.29), перейшовши від порівняння до лінійного рівняння

$$\sum_{j=1}^k \prod_{i=j}^k P_i = r_1 \prod_{j=1}^k P_j + 1, \quad (7.30)$$

де $r_1 = 0, 1, 2, \dots$ - ранг СЗК.

При зміні параметра $k = 1, 2, \dots$ згідно останнього виразу маємо:

$$\begin{aligned} k = 1; \quad P_1 &= r_1 P_1 + 1; \\ k = 2; \quad P_1 + P_2 &= r_1 P_1 \cdot P_2 + 1; \\ k = 3; \quad P_1 \cdot P_2 + P_1 \cdot P_3 + P_2 \cdot P_3 &= r_1 P_1 P_2 P_3 + 1; \\ &\dots \dots \dots \end{aligned}$$

Очевидно, що при $k = 1$ і $k = 2$ не може бути знайдено ні однієї СЗК з властивістю (7.29). В інших випадках (7.30) доцільно привести до вигляду

$$x_k \sum_{j=1}^{k-1} \prod_{i \neq j}^k P_i + \prod_{i \neq k}^k P_i = r_1 x_k \prod_{j \neq k}^k P_j + 1,$$

або кінцево:

$$x_k = \frac{\prod_{i \neq k}^k P_i - 1}{r_1 \prod_{j \neq k}^k P_j - \sum_{j=1}^k \prod_{i \neq j}^k P_i}, \quad (7.31)$$

де $x_k = P_k$ -й модуль СЗК, який задовольняє умові (7.30) і обчислюється на основі відомих $k-1$ модулів.

Наприклад:

Нехай заданий набір модулів СЗК

$$\{P_j\} = \{2, 3, 7, x\}.$$

Необхідно знайти таке x , щоб $\{m_j\} = \{1, 1, 1, 1\}$. Підставляючи числові значення цього прикладу в (7.31) отримаємо:

$$x = \frac{42 - 1}{42 - (6 + 14 + 21)} = \frac{41}{42 - 41}$$

і при $r_1 = 1$, $x = 41$.

Так як отримане значення x просте число, рішення є єдиним.

Таким чином отримуємо набір модулів досконалої СЗК

$$\{P_j\} = \{2, 3, 7, 41\}, \quad \rho = 1722$$

Для перевірки правильності рішення (7.31) розрахуємо ортогональні базиси СЗК з таким набором модулів

$$\dot{B}_1 = \frac{1722}{2} m_1 \equiv 1(\text{mod } 2); m_1 = 1;$$

$$\dot{B}_2 = \frac{1722}{3} m_2 \equiv 1(\text{mod } 3); m_2 = 1;$$

$$\dot{B}_3 = \frac{1722}{7} m_3 \equiv 1(\text{mod } 7); m_3 = 1;$$

$$\dot{B}_4 = \frac{1722}{41} m_4 \equiv 1(\text{mod } 41); m_4 = 1.$$

Отже

$$\dot{N}_k = (\dot{B}_1 + \dot{B}_2 + \dot{B}_3 + \dot{B}_4)(\text{mod } \rho) = 1,$$

Тобто для нашого прикладу

$$\dot{N}_k = (861 + 574 + 246 + 42)(\text{mod } 1722) = 1,$$

що відповідає умові $\{m_j\} = \{1, 1, 1, 1\}$.

На рис.7.4 графічно показані рішення (7.31), розраховані для різних наборів модулів $\{P_j\}$ та значень r_1 і k . Для виразності ілюстрації вказані графіки подані в логарифмічному масштабі.

На рисунку показано, що для $k = 3$ існує єдиний набір модулів $\{2, 3, 5\}$, якому відповідає набір $\{m_j\} = \{1, 1, 1\}$. Зі збільшенням k кількість такого класу СЗК зростає.

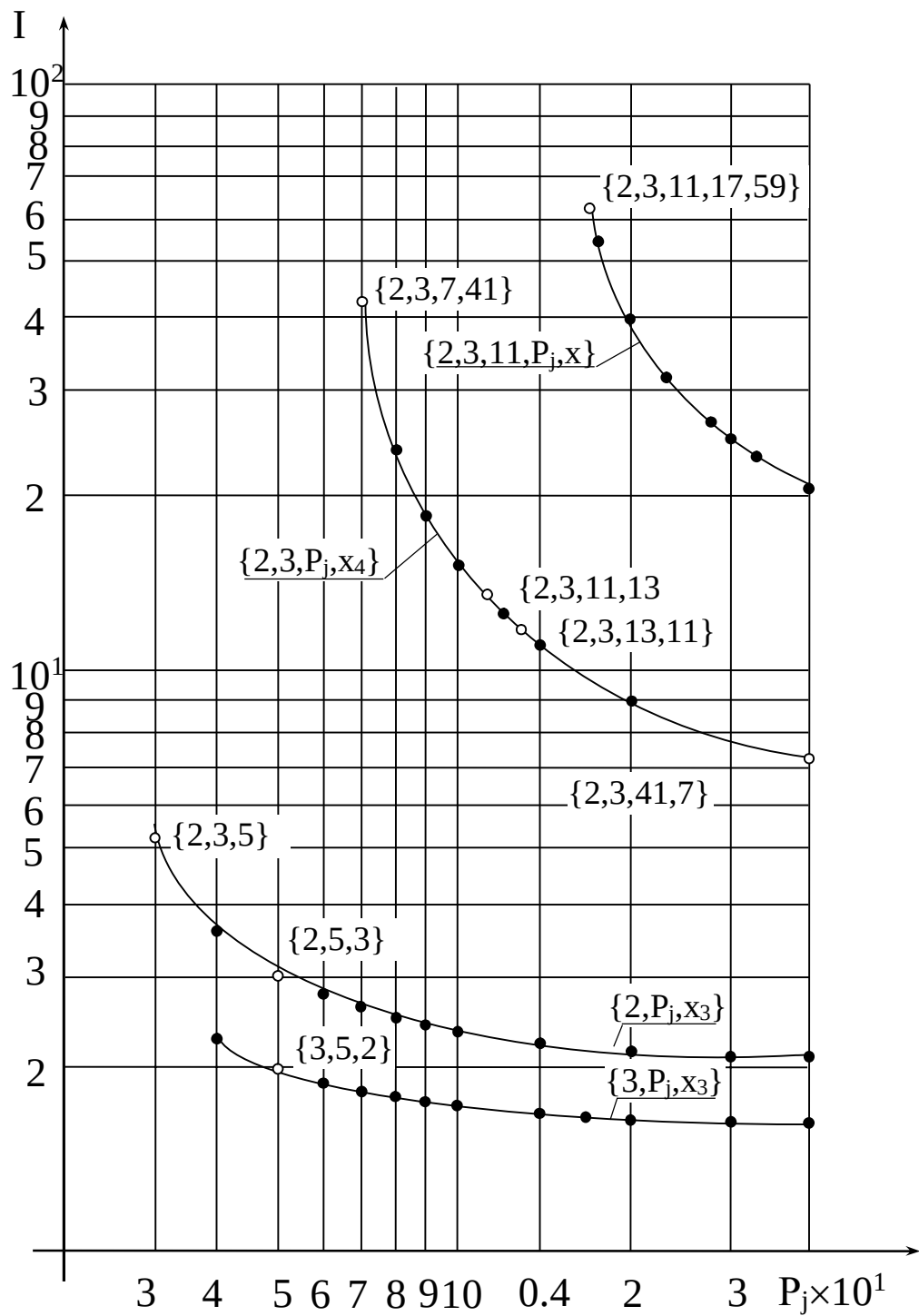


Рис. 7.4. Набори модулів досконалої форми СЗК.

Аналогічно знайдемо вирази існування СЗК для умов (7.29.1-к):

$$\begin{aligned}
 1) & \sum_{j=1}^{k-1} \prod_{i \neq j}^{k-1} P_i - \prod_{j \neq k}^k P_i = \pm r_1 \prod_{j=1}^k P_j + 1; \\
 2) & \sum_{j=1}^{k-2} \prod_{i \neq j}^{k-2} P_i - \sum_{j=k-1}^k \prod_{i \neq j}^k P_i = \pm r_1 \prod_{j=1}^k P_j + 1; \\
 & \dots \dots \dots \\
 k) & \sum_{j=1}^{k-2} \prod_{i \neq j}^{k-2} P_i - \sum_{j=1}^k \prod_{i \neq j}^k P_i = \pm r_1 \prod_{j=1}^k P_j + 1.
 \end{aligned} \tag{7.32}$$

Дослідження отриманих виразів показує, що умова (7.32) повністю співпадає з (7.30) Є, а також відображає параметри існування СЗК з одиничними ваговими коефіцієнтами $\{m_j\} = \{-1, -1, \dots, -1\}$.

Приведення СЗК досконалої форми дозволяє повністю виключити модульні операції у його зворотньому перетворенні.

Умову утворення досконалої без рангової СЗК можна отримати у вигляді нерівності, підставивши в (7.16) граничні значення $b_j = m_j = P_{j-1}$, тобто

$$\sum_{j=1}^k (P_j - 1) \frac{(P_j - 1) \cdot \rho}{P_j} < \prod_{j=1}^k P_j, \quad (7.33)$$

Звідси після елементарних перетворень отримаємо

$$\sum_{j=1}^k (P_j - 2 + \frac{1}{P_j}) < 1.$$

Очевидно, що у цьому випадку не існує ні одного набору цілочисельних взаємопростих модулів $P_j \geq 2$, $k \geq 2$, які утворюють досконалу без рангову СЗК.

Враховуючи, що коли $m_j = P_j - 1$, Є то згідно (7.29.к) можна замінити $m_j = -1$ і направленість (7.33) замисати у вигляді

$$\sum_{j=1}^k (P_j - 1) \frac{\rho}{P_j} > - \prod_{j=1}^k P_j$$

або

$$\sum_{j=1}^n (1 - \frac{1}{P_j}) < 1$$

Звідки згідно (7.33) при $k \geq 2$ та коли не існує ні однієї без рангової СЗК.

В якості одного з найбільш простих способів представлення СЗК до досконалої безрангової форми можна застосувати метод надлишкового розширення системи модулів початкової СЗК.

Умова існування досконалої без рангової СЗК для розширеної системи модулів визначається згідно виразу

$$\sum_{j=1}^{k-e} \prod_{i \neq j}^{k-e} P_i - \sum_{j=e}^{k-v} P_i < \pm \prod_{j=1}^k P_j,$$

де e , v - відповідно кількість основних і надлишкових модулів.

Розглянемо приклад:

Нехай маємо набір модулів досконалої СЗК:

$$\{P_j\} = \{2, 3, 5\}; \quad \rho = 30; \quad B_1^0 = 15; \quad B_2^0 = 10; \quad B_3^0 = 6;$$

$$m_1 = m_2 = m_3 = 1; \quad N_k = 26_{(10)} = 11010_{(2)}.$$

Представимо число N_k у СЗК, тобто:

$$N_k \begin{cases} \rightarrow \text{res}26(\text{mod}2)=0 \\ \rightarrow \text{res}26(\text{mod}3)=2 \\ \rightarrow \text{res}26(\text{mod}5)=1 \end{cases} = (0, 2, 1) \quad (b_1, b_2, b_3).$$

Запишемо залишки b_1, b_2, b_3 , у нормалізованій формі:

$$[b_1]_0 = 0/2 = 0; \quad [b_2]_0 = 2/3 = 0,66; \quad [b_3]_0 = 1/5 = 0,2.$$

Виконуємо зворотнє перетворення досконалої нормалізованої форми СЗК згідно виразу

$$[N_k]_0 = (0 + 0,66 + 0,2) \bmod 1 = 0,86.$$

Отримаємо цілочисельне значення $[N_k]_0$

$$N_k = \hat{E}[\lfloor N_k \rfloor_0 \cdot \rho] = \hat{E}[0,86 \cdot 30] = 26.$$

Аналогічно у двійковій системі числення виконаємо операції перетворень ДНСЗК:

$$\lfloor b_1 \rfloor_0 = 0,00000_{(2)}; \quad \lfloor b_2 \rfloor_0 = 0,10101_{(2)}; \quad \lfloor b_3 \rfloor_0 = 0,00110_{(2)};$$

$$\begin{array}{r} 0,00000 \\ 0,10101 \\ \hline \lfloor N_k \rfloor = \frac{0,00110}{0,11011} \end{array} \quad \begin{array}{r} \times 0,11011 \\ \hline 11110 \\ 110110 \\ 11011 \\ 11011 \\ 11011 \\ \hline N_k = \hat{E}[1100101010] = 26_{10} \end{array}$$

Труднощі пошуку досконалої форми СЗК, а також їх відносна рідкість вимагають розвитку інших методів отримання досконалих перетворень базису Крестенсона-Галуа.

Одним з таких методів є отримання скороченої СЗК з ваговими коефіцієнтами $\{m_j\} = \{1, 1, \dots, 1\}$ шляхом виключення одного або кількох модулів у перетвореннях СЗК.

Умова існування таких СЗК має вигляд

$$\sum_{j=1}^{k-v} \prod_{i \neq 1}^{k-v} P_i = \prod_{j=1}^k P_j - 1 - \sum_{j=v}^k \prod_{i \neq j}^k P_i$$

Наприклад виберемо в якості початкового набору модулів СЗК наступні

$$\{P_j\} = \{5, 7, 8, 9, 19\},$$

для якого $\{m_j\} = \{1, 1, 1, 1, 8\}$, і виключимо з нього модуль $P_5 = 19$ прийнявши $b_5 = 0$.

Тоді всі операції перетворення в СЗК можуть виконуватись по алгоритмах безрангових досконалих форм СЗК.

Іншим універсальним практичним методом приведення до досконалої форми СЗК для будь яких наборів модулів $\{P_j\}$ є використання властивості (7.34).

Суть такого методу приведення в СЗК до досконалої форми полягає в тому, що при виконанні зворотнього перетворення СЗК апріорно задається набір $\{m_j\} = \{1, 1, \dots, 1\}$ для всіх модулів $\{P_j\}$. При цьому легко показати, що однозначність перетворення СЗК не порушується, але при виконанні зворотнього перетворення необхідно коректувати значення добутих N_k залишків b_j з врахуванням конкретних значень $\{m_1, m_2, \dots, m_j\}$ для заданого набору $\{P_1, P_2, \dots, P_j\}$ згідно виразу:

$$b_j = \text{res } \overset{0}{b_j} \cdot \text{mod } P_j,$$

де $\overset{0}{b_j}$ - залишок отримання із числа $\overset{0}{N_k}$, тобто

$$\overset{0}{b_j} = \text{res } \overset{0}{N_k} (\text{mod } P_j),$$

а число $\overset{0}{N_k}$ досконалого перетворення СЗК розраховується згідно виразу

$$\overset{0}{N_k} = \left(\overset{0}{B_1} \overset{0}{b_1} + \overset{0}{B_2} \overset{0}{b_2} + \dots + \overset{0}{B_k} \overset{0}{b_k} \right) (\text{mod } \rho),$$

де $\overset{0}{B_0} = \rho / P_j \cdot 1$. Тобто: всі $m_j = 1$; $j \in \overline{1, k}$.

Нехай заданий набір модулів СЗК: $\{P_j\} = \{P_1, P_2, \dots, P_k\}$, $\rho = \prod_{j=1}^k P_j$,

якому відповідають набори:

$$\{m_j\} = \{m_1, m_2, \dots, m_k\}, \quad \{B_j\} = \left\{ \frac{\rho}{P_j} \cdot m_j \right\},$$

причому, не виконується ні одна з умов (7.35) і ні одним з розглянутих раніше методів СЗК з таким набором модулів не може бути приведене до досконалої форми.

Тоді, якщо прийняти всі $\left\{ \overset{0}{m_j} \right\} = \{1, 1, \dots, 1\}$ незалежно від їх фактичних значень $\{m_j\}$, то

однозначність перетворень СЗК не порушується при виконанні їх згідно виразів

$$b_j = \text{res} \left[\text{res} \overset{0}{N_k} (\text{mod } P_j) \cdot m_j \right] \text{mod } P_j.$$

$$\overset{0}{N_k} = \text{res} \left(\overset{0}{B_1} \cdot b_j + \overset{0}{B_2} \cdot b_2 + \dots + \overset{0}{B_k} \cdot b_k \right) \text{mod } \rho$$

В нормалізованій досконалій формі СЗК аналогічні перетворення виконуються згідно виразів:

$$\left[\overset{0}{b_j} \right]_0 = b_j / P_j; \quad \left[\overset{0}{N_k} \right]_0 = \text{res} \sum_{j=1}^k \left[\overset{0}{b_j} \right]_0 (\text{mod } 1);$$

$$\overset{0}{N_k} = \hat{E} \left[\left[\overset{0}{N_k} \right]_0 \cdot \rho \right];$$

$$b_j = \check{E} \left[\left[\overset{0}{b_j} \right] \cdot \text{res} P_j \cdot m_j (\text{mod } P_j) \right].$$

Операція обчислення залишків $\overset{0}{b_j}$ з врахуванням $\{m_j\}$ виконується шляхом циклічного зсуву на величину нормалізованих значень з модулів P_j .

Наприклад:

Нехай маємо набір модулів СЗК:

$P_1=5; P_2=7; P_3=11$; тоді $\rho = 5 \cdot 11 \cdot 13 = 385$;

$$B_1 = 77 \cdot m_1 (\text{mod } 5); \quad m_1 = 3; \quad B_1 = 231; \quad \overset{0}{B_1} = 77;$$

$$B_2 = 55 \cdot m_2 (\text{mod } 7); \quad m_2 = 6; \quad B_2 = 230; \quad \overset{0}{B_2} = 55;$$

$$B_3 = 35 \cdot m_3 (\text{mod } 11); \quad m_3 = 6; \quad B_3 = 210; \quad \overset{0}{B_3} = 35.$$

Таблиці значень залишків b_j для заданої (а) та досконалої форми СЗК (б) зваженої цілочисельної форми СЗК мають наступний вигляд:

b_i	P_1	P_2	P_3	$\overset{o}{b_i}$	P_1	P_2	P_3
0	0	0	0	0	0	0	0
1	1	1	1	1	3	6	6
2	2	2	2	2	4	1	7
3	3	3	3	3	0	2	8
4	4	4	4	4	1	2	9
5	—	5	5	5	—	4	10
6	—	6	6	6	—	5	1
7	—	—	7	7	—	—	2
8	—	—	8	8	—	—	3
9	—	—	9	9	—	—	4
10	—	—	10	10	—	—	5

а)

б)

Таким чином у цілочисельній формі СЗК число $N_k=1$ обчислюється згідно виразу

$$N_k = (231 \cdot 1 + 330 \cdot 1 + 210 \cdot 1) \pmod{385} = 1$$

або

$$N_k = (77 \cdot 3 + 55 \cdot 6 + 35 \cdot 6) \pmod{385} = 1,$$

Аналогічно складаються таблиці нормалізованих значень $\begin{bmatrix} 0 \\ b_j \\ 0 \end{bmatrix}$ у десятковій та двійковій

системах числення.

7.6. Міжбазисні перетворення на основі розмежованої СЗК

Теоретичні основи та принципи реалізації розмежованої СЗК викладені в опублікованих одноосібних та сумісних з автором наукових працях О.І. Волинського, де викладені методи між базисних перетворень Радемахера-Крестенсона та Крестенсона-Радемахера.

Для створення повної функціонально-ефективної розмежованої СЗК (РСЗК) потрібно спростити існуючі алгоритми виконання операцій порівняння чисел. Умову створення швидкісного пристрою в РСЗК задовольняють алгоритми обчислення всіх арифметичних операцій крім ділення. Для реалізації операції ділення потрібна НДСЗК. Теоретичною основою даної форми є рівняння (7.16), підставивши в яке $m_1=m_2=m_k=1$, отримаємо базове рівняння перетворення НДСЗК у вигляді:

$$[N_k]_0 = \text{res} \sum_{i=1}^k [b_i]_0 \pmod{1}. \quad (7.34)$$

З рівняння (7.34) видно, що з перетворення НДСЗК виключена операція множення і саме перетворення виконується у вигляді сумування нормалізованих залишків $[b_i]_0$ по $\text{mod}1$, що відповідає операції відкидання цілої частини результату.

Розглянемо приклад застосування НДСЗК.

Таблиця 7.1.

Перетворення векторів багатомірного дискретного простору залишків у одновимірний дискретний простір Радемахера.

N_k	$[b_1]_0$	$[b_2]_0$	$[b_3]_0$	$[N_k]_0$
0	0	0	0	0
1	0.5	0.33	0.2	0.03
2	0	0.66	0.4	0.06
3	0.5	0	0.6	0.1
4	0	0.33	0.8	0.13
5	0.5	0.66	0	0.16
6	0	0	0.2	0.2
7	0.5	0.33	0.4	0.23
...	...	...	...	...
29	0.5	0.66	0.8	0.96

На основі табл. 7.1 запропонований спрощений алгоритм перетворення чисел з базису Крестенсона в базис Радемахера, що суттєво упростило здійснення операції порівняння, яка є базовою при здійсненні операцій віднімання та ділення.

Розглянемо приклад порівняння двох чисел в НДСЗК. Нехай: $N_{k1}=5$, $N_{k2}=7$.

Дані числа представлені згідно табл. 7.1 у вигляді: $N_{k1} = (0,5;0,66;0)$ і $N_{k2} = (0,5;0,33;0,4)$ і можуть бути нормалізовані у вигляді: $[N_{k1}]_0 = 0,16$ і $[N_{k2}]_0 = 0,23$.

Графічно операцію вибірки залишків з табл. 7.1 можна представити у формі графа (рис.7.5).

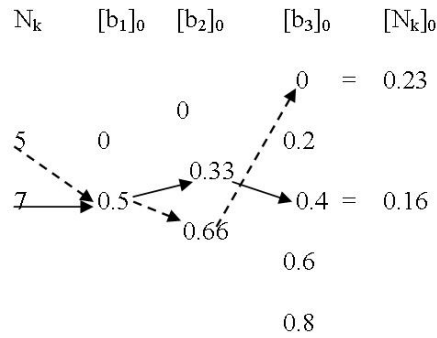


Рис. 7.5. Граф перетворення чисел з НДСЗК у базис Радемахера.

Витративши ще один такт на сумування залишків, отримаємо нормалізоване число, з яким потрібно зробити однотактну операцію віднімання. Далі порівнюємо результат з нулем і визначаємо знак результату, який необхідний для реалізації операції ділення в НДСЗК: $[N_{k1}]_0 - [N_{k2}]_0 \stackrel{>}{<} 0$.

7.6.1. Розрахунок набору модулів для реалізації 16-бітного та 1024-бітного процесора.

У таблиці 7.1 приведений приклад набору модулів, які відповідають досконалій формі СЗК. Ця умова обмежує можливості реалізації процесорів певної розрядності з оптимальним набором модулів. Оскільки довільному набору взаємопростих модулів відповідає довільний набір коефіцієнтів m_i .

Запропонований алгоритм приведення будь-якої цілочисельної форми СЗК до НДСЗК, який реалізується шляхом корекції графа визначення залишків згідно розрахованого набору коефіцієнтів m_i .

Для реалізації 16-бітного процесора оптимальний набір модулів з максимальним діапазоном кодування задається масивом чисел:

$$p_i = (3, 5, 7, 8, 11, 13)$$

Виконаємо розрахунок параметрів даної СЗК.

Діапазон кодування: $P = 3 \cdot 5 \cdot 7 \cdot 11 \cdot 13 = 120120$

Базисні числа:

$$B_1 = \frac{120120}{3} \cdot m_1 = 1(\bmod 3), m_1 = 2; B_1 = 80080;$$

$$B_2 = \frac{120120}{5} \cdot m_2 = 1(\bmod 5), m_2 = 4; B_2 = 96096;$$

$$B_3 = \frac{120120}{7} \cdot m_3 = 1(\bmod 7), m_3 = 5; B_3 = 85800;$$

$$B_4 = \frac{120120}{8} \cdot m_4 = 1(\bmod 8), m_4 = 7; B_4 = 105105;$$

$$B_5 = \frac{120120}{11} \cdot m_5 = 1(\bmod 11), m_5 = 7; B_5 = 76440;$$

$$B_6 = \frac{120120}{13} \cdot m_6 = 1(\bmod 13), m_6 = 4; B_6 = 36960.$$

Скоректований граф обчислення залишків згідно розрахованого набору коефіцієнтів m_i буде мати вигляд:

Таким чином у загальному вигляді зворотнє перетворення РСЗК аналітично описується виразом:

$$N_k = \text{res} \sum_{r=1}^n \sum_{i=1}^k \text{res}(b_{1i} + b_{2i} + \dots + b_{ri} + \dots + b_{ni}) \bmod P_i \cdot B_i \bmod P. \quad (7.35)$$

Застосування міжбазисного перетворення при такому розмежуванні базису Радемахера дозволяє схемотехнічно реалізувати міжбазисне перетворення на основі дешифратора по відповідному модулю.

В задачах цифрового опрацювання сигналів широко використовуються 16-бітні та 24-бітні сигнальні процесори. Ці процесори виконують більше ніж 90% базових операцій додавання, множення та порівняння чисел, при використанні процедур цифрової згортки, фільтрації, цифрової томографії та інше.

Тому можливості реалізації швидкодіючого процесора в РСЗК розглянемо на основі 16-бітного процесора.

Діапазон кодування чисел РСЗК розраховується згідно виразу:

$$P = \prod_{i=1}^k p_i, \text{ де } 0 \leq N_k \leq P-1,$$

N_k – двійковий код числа, p_i – система взаємопростих модулів, а розрядність процесора

$$n_k = \hat{E}[\log_2(P-1)].$$

З структури розмежованого процесора зрозуміло, що вона потребує обчислення залишків для кожного компонента згідно виразу:

$$b_{ij} = \text{res} N_{ij} \pmod{p_i}.$$

При цьому, процедура обчислення загального залишку виконується згідно виразу:

$$b_i = \text{res}(b_{i1} + b_{i2} + \dots + b_{in}) \bmod p_i.$$

При 4-бітній розрядності компонентів 16-бітного процесора оптимальний набір модулів з максимальним діапазоном кодування задається масивом чисел:

$$p_i = (3, 5, 7, 8, 11, 13),$$

що відповідає діапазону кодування:

$$P = 120120.$$

Суттєвою перевагою РСЗК є значне спрощення алгоритму перевodu чисел з базису Крестенсона в базис Галуа, що схемотехнічно реалізується за допомогою матричних дешифраторів. В табл. 7.2 приведений приклад реалізації дешифратора 4-бітних компонентів 16-бітного процесора залишкових класів по модулю 5.

Таблиця 7.2.

Схема обчислення залишку по модулю 5.

Модуль p_i																	Число векторов	
5	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	79	
	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1		2^3
	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1		2^2
	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1		2^1
	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1		2^0
																	$b_y = \text{res} N_y \pmod{5}$	
																	2^2	
																	2^1	
																	2^0	

Незважаючи на можливості міжбазисного перетворення Радемахера–Крестенсона на основі дешифратора при степені розмежування двійкових чисел на N – розрядів ($N=2^k$, $k=2, 4, \dots$) таке розмежування характеризується недоліком, оскільки при виконанні арифметичних операцій в РСЗК необхідно реалізувати наскрізні переноси між N – розрядними групами бітів базису Радемахера.

7.6.3. Бінарно-розмежована СЗК.

При бінарному розмежуванні двійкових чисел базису Радемахера, тобто $k=0$, структура розмежування має наступний вигляд:

$$\left| \begin{array}{c} 32 \\ \hline N32k \end{array} \right| \dots \left| \begin{array}{c} i \\ \hline N1k \end{array} \right| \dots \left| \begin{array}{c} 3 \\ \hline N3k \end{array} \right| \left| \begin{array}{c} 2 \\ \hline N2k \end{array} \right| \left| \begin{array}{c} 1 \\ \hline N1k \end{array} \right|$$

В результаті такого розмежування двійкового числа $(X_{n-1}, X_{n-2}, \dots, X_i, \dots, X_1, X_0)$ формується матриця залишків кожного i -того розряду у системі взаємопростих модулів $P_1, P_2, \dots, P_j, \dots, P_k$ (табл.7.3).

Таблиця 7.3.

Матриця залишків числа X .

	X_{n-1}	X_{n-2}	...	X_i	...	X_1	X_0
P_1	$b_{n-1,1}$	$b_{n-2,1}$	...	$b_{i,1}$	...	$b_{1,1}$	$b_{0,1}$
P_2	$b_{n-1,2}$	$b_{n-2,2}$	...	$b_{i,2}$	...	$b_{1,2}$	$b_{0,2}$
P_3	$b_{n-1,3}$	$b_{n-2,3}$	...	$b_{i,3}$	...	$b_{1,3}$	$b_{0,3}$
P_4	$b_{n-1,4}$	$b_{n-2,4}$	...	$b_{i,4}$	...	$b_{1,4}$	$b_{0,4}$
...	...	...	...	...	...	...	...
P_j	$b_{n-1,j}$	$b_{n-2,j}$	...	$b_{i,j}$	...	$b_{1,j}$	$b_{0,j}$
...	...	...	...	...	...	...	...
P_k	$b_{n-1,k}$	$b_{n-2,k}$	...	$b_{i,k}$	...	$b_{1,k}$	$b_{0,k}$

Для переходу в базис Крестенсона над елементами стрічок матриці поданої в табл. 9.3 виконується наступна операція:

$$res(b_{n-1,j} + b_{n-2,j} + \dots + b_{i,j} + \dots + b_{1,j} + b_{0,j}) \bmod P_i \quad (7.36).$$

Для підвищення швидкості модульної операції (7.36) доцільно застосувати пірамідальний алгоритм сумування згідно рис.7.8.

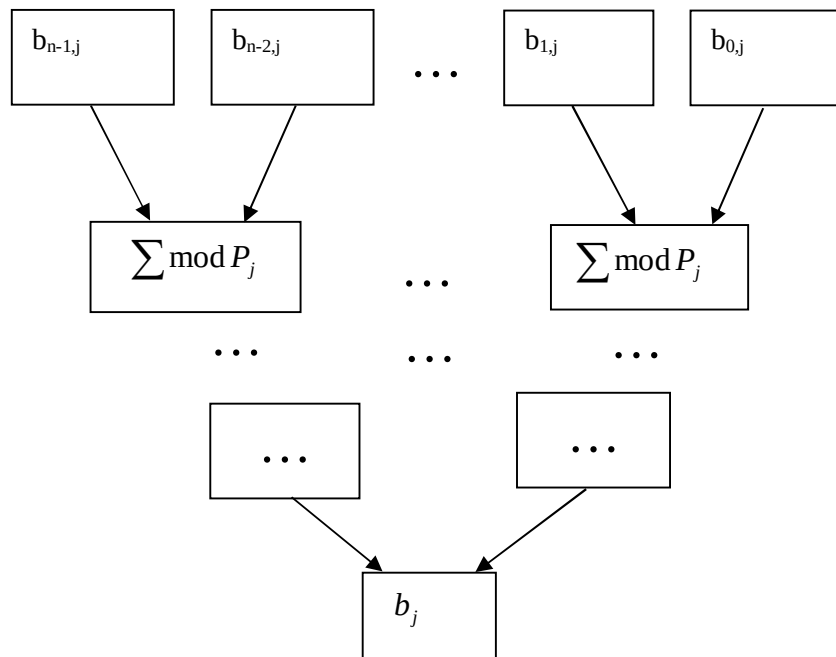


Рис. 7.8. Пірамідальний алгоритм сумування залишків в РСЗК.

Швидкодія такого пірамідально-модульного суматора розраховується за формулою:

$$m = \log_2 n^i,$$

n – розрядність процесора базису Радемахера.

Висока швидкодія такого компонента міжбазисного перетворення Радемахера–Крестенсона потребує великої кількості суматорів в залежності від розрядності процесора число яких розраховується за формулою:

$$S = n + n/2 + n/4 + \dots + n/n.$$

Таким чином загальний об'єм апаратного обладнання даного міжбазисного перетворення можна оцінити згідно виразу

$$Q = K \cdot S,$$

де K – число взаємопростих модулів базису Крестенсона.

Об'єм мікроелектронного обладнання, міжбазисного перетворювача (МБП), можна суттєво зменшити на основі паралельної інформаційної технології та структури показаної на рис.7.9.

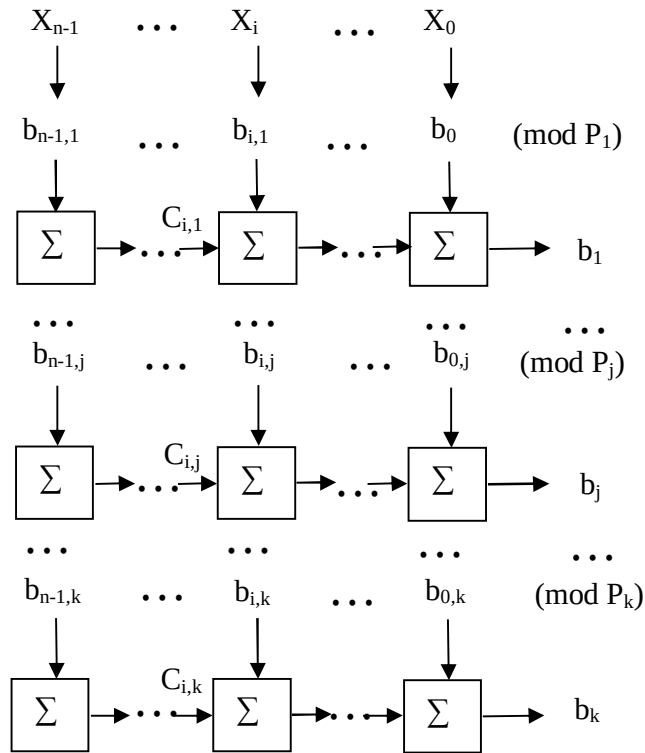


Рис. 7.9. Структура міжбазисного перетворювача Радемахера–Крестенсона.

7.6.4. Оцінка швидкодії суматора по модулю P_j .

Згідно архітектури пірамідального та лінійного міжбазисних перетворень по модулю P_j , показаних архітектур на рис.7.8 та 7.9, порівняння їх характеристик приведено на рис.7.10, звідки видно, що лінійна архітектура потребує в два рази менше обладнання по відношенню до пірамідальної.

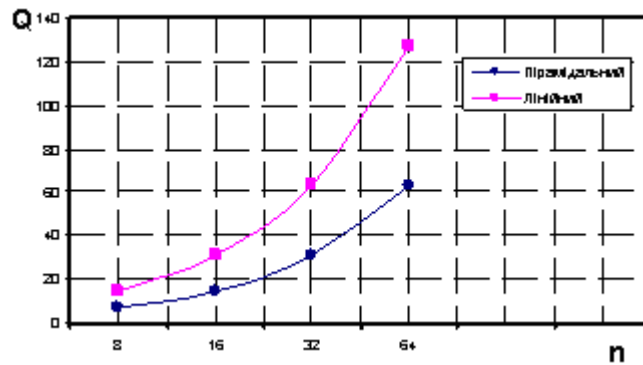


Рис. 7.10. Об'єм мікроелектронного обладнання міжбазисного перетворення Радемахера–Крестенсона.

Результати аналізу швидкодії двох досліджуваних архітектур міжбазисного перетворення при унітарному кодуванні залишків (рис. 7.11) розраховується згідно виразів:

$$V_p = \frac{1}{2 + \log_2 n}; \quad V_l = \frac{1}{n};$$

де n – число розрядів процесора; V_p – швидкодія пірамідального МБП; V_l – швидкодія лінійного МБП.

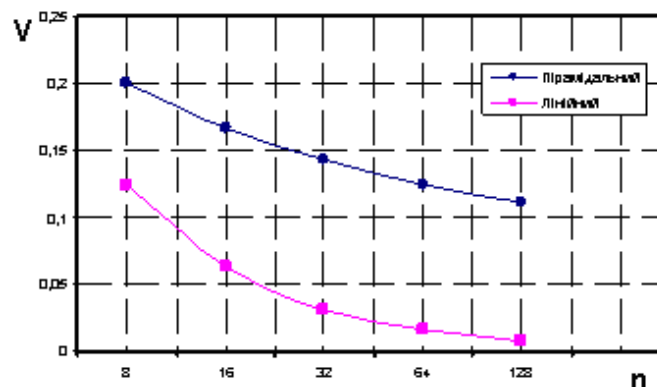


Рис. 7.11. Швидкодія міжбазисного перетворення Радемахера–Крестенсона.

7.6.5. Оцінка складності арифметики у базисах Радемахера, Крестенсона та Галуа.

Порівняльна оцінка функціональних можливостей арифметики названих ТЧБ подана в табл. 7.4.

Таблица 7.4.

Функціональні можливості ТЧБ.

№	Базові операції	Радемахер	Крестенсон	Галуа	БРСЗК
1	Додавання	$2nv$	v	$3v$	v
2	Зсув	v	-	$2v$	v
3	Множення	$2v(2n+1)$	v	?	?
4	Рівності	v	v	v	v
5	Знакова(старшинства)	nv	?	?	nv
6	Віднімання	$(3n+5)v$	?	?	$2nv$
7	Ділення	n^2v	?	-	?
8	Модульна	n^2v	$2nv$	?	$2nv$

В табл.7.4. n - розрядність представлення чисел, v – часова складність виконання операцій.

Оцінка існування та складності алгоритмів міжбазисних перетворень ТЧБ приведена в табл. 7.5.

Таблиця 7.5.

Міжбазисні перетворення досліджуваних ТЧБ.

Теоретико-числові бази	Алгоритм міжбазисних перетворень
Радемахера-Крестенсона	$N_k = (a_{n-1}, \dots, a_i, \dots, a_0); a_i \in \overline{0,1}; N_k = \sum_{i=0}^{n-1} a_i \cdot 2^i;$ $N_k \rightarrow \begin{matrix} \rightarrow b_1 \\ \rightarrow \dots \\ \rightarrow b_i \\ \rightarrow \dots \\ \rightarrow b_k \end{matrix} \quad b_i = \text{res} N_k (\text{mod} p_i);$ $0 \leq N_k \leq P, \quad p_i \equiv p_j.$ $N_k = a_i p_i + b_i, \quad P = \prod_{i=1}^k p_i;$
Радемахера-Галуа	$N_k \rightarrow \sum_{i=0}^{n-1} i \rightarrow G_0, G_1 \dots G_{i-1}; G_0 = (11 \dots 1); G_{i+1} = G_i \oplus G_{i-n}; i \in \overline{0, n-1}.$
Крестенсона-Радемахера	$N_k = \text{res} \sum_{i=1}^k b_i \cdot B_i (\text{mod } P), \quad B_i = \frac{P}{p_i} \cdot m_i \equiv 1 (\text{mod } p_i).$
Крестенсона-Галуа	$N_k = (b_1, \dots, b_i, \dots, b_k) \rightarrow \sum_{i=0}^{n-1} i \rightarrow G_i \dots G_{i-n};$ $G_0 = (11 \dots 1); G_{i+1} = G_i \oplus G_{i-n}; i \in \overline{0, n-1}.$
Галуа-Радемахера	$G_i, G_{i-1} \dots G_{i-n} \rightarrow \sum_{i=0}^{n-1} i \rightarrow$ $N(a_{n-1}, \dots, a_i, \dots, a_0) \rightarrow N_k; a_i \in \overline{0,1}.$
Галуа-Крестенсона	$G_i, G_{i-1} \dots G_{i-n} \rightarrow \sum_{i=0}^{n-1} i \rightarrow N_k = (b_1, b_2, \dots, b_i, \dots, b_k).$

Сумісне використання двовимірного базису Радемахера, розмежованої, цілочисельної та досконалої нормалізованої форм СЗК дозволяє реалізувати виключно високопродуктивні мультибазисні процесори згідно алгоритму, представленого на рис.7.12.

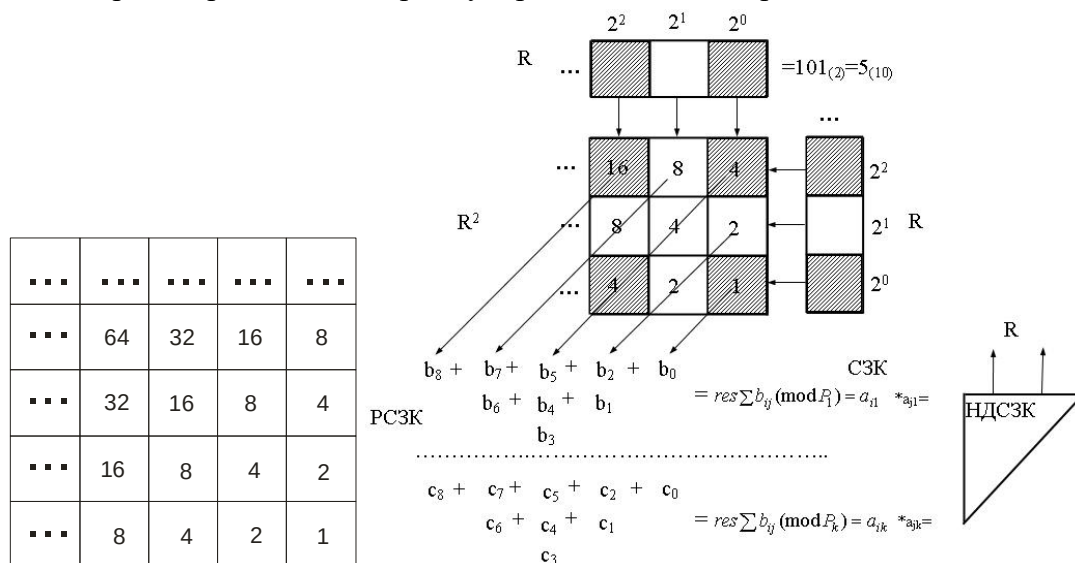


Рис 7.12. Мультибазисні перетворення Радемахера-Крестенсона-Галуа (R-C-G).

Аналіз сучасного стану реалізації базисних функцій ТЧБ орієнтованих на створення високопродуктивних спецпроцесорів опрацювання великорозрядних чисел дозволяє зробити висновок, що успішний розвиток теорії ТЧБ Крестенсона-Галуа, з метою реалізації всіх базових функцій арифметики процесорів, є доцільним та перспективним.

Необхідно зауважити, що досконала та розмежована форми СЗК є особливо перспективними для створення високопродуктивних мультибазисних процесорів. При цьому новим досягненням в галузі теорії систем числення є розробка двовимірного матричного представлення базису Радемахера.

7.7. Оптимізація міжбазисного перетворення Галуа-Крестенсона.

Міжбазисне перетворення Радемахера-Галуа та Крестенсона-Галуа може виконуватися на основі проміжного перетворення в Унітарний базис методом одностороннього наближення (рис.7.14).

В той же час такий спосіб міжбазисного перетворення характеризується найбільшою часовою складністю і відповідно низькою швидкістю його реалізації за допомогою спецпроцесорів чи програмних засобів.

Методи адаптивного сходження та інкрементної збіжності за ключем більш ефективні, проте є множина кодових послідовностей, які за допомогою адаптивного сходження декодувати неможливо, а метод збіжності за ключем повністю не використовує інформаційну потужність ключа.

В результаті аналізу структури та особливостей розміщення елементів кодових послідовностей Галуа у фрагментах двійкових рекурентних кодів поля Галуа С.І.Мельничуком запропоновано метод позиційної збіжності за ключем.

Суть методу полягає в проведенні перебору фрагментів кодової послідовності від прийнятої (m_{var}) до ключової (m_{const}) інкрементним або декрементним зсувом зі змінним кроком (k), величина якого визначається позицією останнього біту, що не співпадає в поточному та ключовому фрагменті і опосередковано від довжини породжуючого ключа (L) коду Галуа:

$$k = f(m_{var}, m_{const}).$$

Величина зміщення кроку (k) відносно поточного кодового фрагменту визначається на основі побітного порівнювання елементів прийнятого фрагменту послідовності (m_{var}) з елементами ключа (m_{const}). З метою оптимізації порівнювання доцільно здійснювати з кінця кодового фрагменту зменшуючи (k) при рівності відповідних елементів. У випадку нееквівалентності (m_{var}) _{k} та (m_{const}) на основі (k) генерується новий кодон Галуа, після чого порівняння повторюється.

Описана процедура проводиться поки ($k \neq 0$), до співпадання усіх елементів (m_{var}) _{k} та (m_{const}). Напрямок зміщення, в кінець чи початок, визначає інкрементний чи декрементний закон сходження до ключа. Реалізація псевдопаралельного сходження, тобто одночасно інкрементного та декрементного зміщення, здійснюється аналогічно до методу псевдопаралельного наближення.

Доцільно зазначити, що описаний метод ефективний тільки у випадку використання монотонного породжуючого ключа, тобто коли усі його елементи (біти) однакові "0" чи "1". Якщо ключем обрано іншу послідовність то декодування здійснюється до моменту досягнення монотонного фрагменту, після чого необхідно виконати операцію віднімання (чи додавання) величини його зміщення відносно вибраного ключа.

Декодування методом інкрементної позиційної збіжності за ключем на прикладі поля Галуа $GF(2^4)$ зі стартовим кодомом (1111) та довжиною породжуючого ключа $L=4$ біти подано на рис.7.13.

№	Код	k_i	№	Код	k_i
0	1111	0	8	1001	3
1	1110	4	9	0010	4
2	1101	3	10	0100	4
3	1010	4	11	1000	4
4	0101	3	12	0001	3
5	1011	2	13	0011	2
6	0110	4	14	0111	1
7	1100	4	0	1111	0

Рис. 7.13. Міжбазисне перетворення Галуа-Радемахера методом інкрементної збіжності за ключем.

Порівняльні характеристики по кількості необхідних операцій порівняння методів одностороннього наближення, адаптивного сходження, збіжності за ключем та методом позиційної збіжності за ключем подано на рис. 7.14.

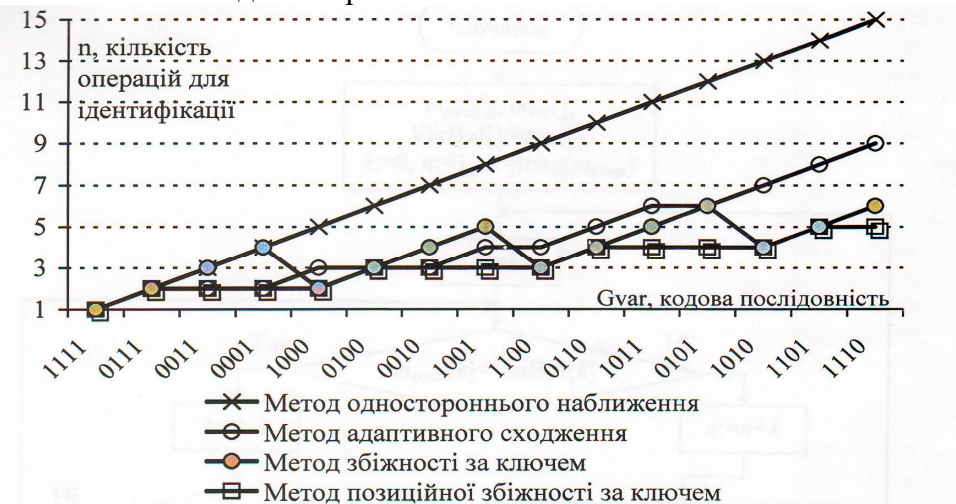


Рис.7.14. Характеристики часової складності між базисного перетворення Галуа-Радемахера.

Як можна побачити, повна (побітна) перевірка поточного та базового (ключового) фрагментів здійснюється лише один раз - при досягненні ключа, в усіх інших випадках ця операція припиняється раніше, що фактично не реалізовано в жодному з вище згаданих методів. Такий підхід дозволяє відмовитись від математичних операцій, а також дещо зменшити кількість операцій порівняння, що виконуються в процесі декодування за рахунок переходів, як показано на рис.7.13.

Декодування рекурентних кодів поля Галуа різних довжин, в порівнянні з методом одностороннього наближення, методом позиційної збіжності за ключем дозволяє зменшити кількість операцій порівняння на 6-67%. Методи збіжності за ключем та адаптивного сходження, в аналогічних умовах, забезпечують зменшення лише на 6-60% та 6-40% відповідно.

Блок-схему алгоритму декодування запропонованого методу подано на рис. 7.15.

В програмній реалізації методу збіжності за ключем, функцію визначення наступної послідовності (фрагменту коду) для порівняння з ключовою $f(k, VarBuf[])$ найдоцільніше зробити циклічною - послідовно формувати k біт коду без проведення порівнювання з $m_{const}[]$.

Утворення кожного наступного елементу (біту) кодової послідовності можна реалізувати за наступними аналітичними виразами:

$$X_i = X_{i-d} \oplus X_{i-h}, \quad \overline{X}_i = \overline{X}_{i-d} \oplus \overline{X}_{i-h},$$

де X_i - наступний біт коду Галуа; d, h - величини зсуву для формуючого ключа.

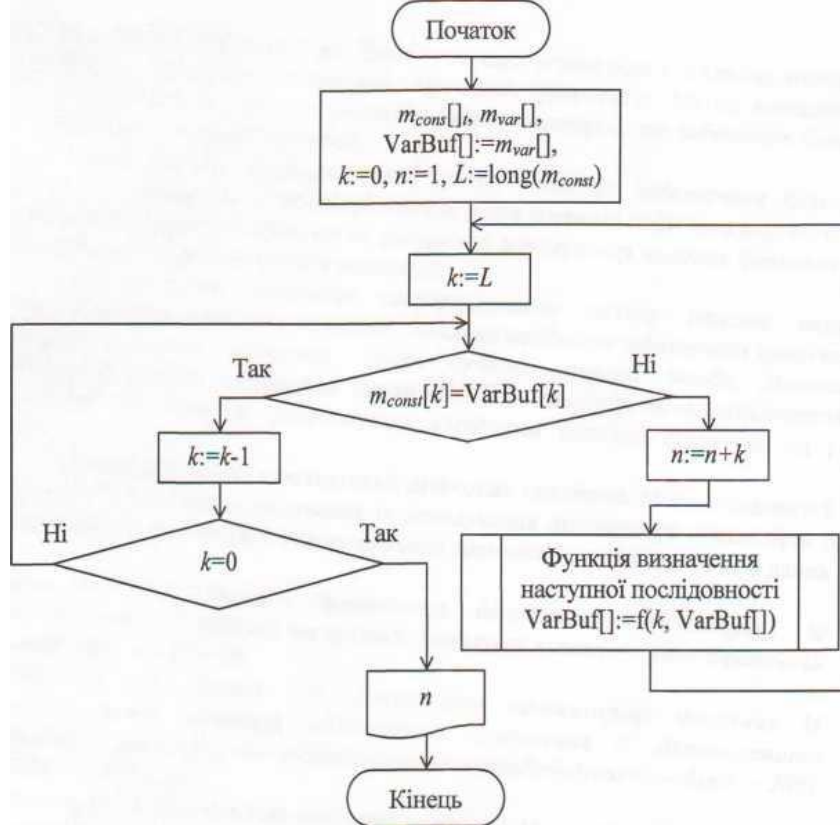


Рис.7.15. Блок-схема алгоритму міжбазисного перетворення Галуа-Крстенсона.

В методах одностороннього та псевдопаралельного наближення кількість операцій порівняння відповідає загальній довжині пошуку. В методах адаптивного сходження та збіжності за ключем кількість таких операцій менша за рахунок використання переходів. Проте операція порівняння в згаданих методах здійснюється над усіма елементами отриманих фрагментів. Метод позиційної збіжності за ключем дозволяє уникнути зайвих перевірок, що забезпечує більш ефективне використання переходів.

Таким чином, описаний метод дозволяє забезпечити більшу ефективність за рахунок оптимізації використання операцій побітного порівняння і, як наслідок, отримати збільшення швидкості декодування кодових фрагментів рекурентних послідовностей Галуа при програмній реалізації.

7.8. Теорія арифметичних операцій в кодах поля Галуа.

У кінцевих полях Галуа правила виконання арифметичних операцій базуються на властивостях ортогональності за симетрії індекса і аргумента у просторі дискретно-постійних функцій ($\varphi(z)$). Таким чином, функції Галуа утворюють базис асимптотично ортогональних рядів, які представляють системи булевих функцій і визначаються правилами додавання, віднімання, множення і ділення над відповідними поліномами по модулю P .

При цьому операція множення реалізується у вигляді згортки двох поліномів:

$$A(x) = a_{k-i}x^{k-i} + a_{k-2}x^{k-2} + \dots + a_1x_1 + a_0;$$

$$H(x) = h_{z-1}x^{z-1} + h_{z-2}x^{z-2} + \dots + h_1x_1 + h_0.$$

Операція згортки послідовно виконується починаючи із старших розрядів за k -тактів.

Результат згортки подається у вигляді:

$$A(x) \cdot H(x) = a_{k-i} h_{z-1} x^{k+z-2} + (a_{k-2} h_{z-2} + a_{k-i} h_{z-2}) \cdot x^{k+z-3} + (a_{k-3} h_{z-1} + a_{k-2} h_{z-2} + a_{k-1} h_{z-3}) x^{k+z-4} + \dots + (a_0 h_2 + a_1 h_1 + a_0 h_0) x^2 + (a_0 h_1 + a_1 h_0) x + a_0 h_0.$$

Перевагою такої рекурсивної процедури є проста реалізація процесора на регістрах зсуву. Але його швидкодія невелика. Досягнути більшої швидкодії в арифметиці Галуа

дозволяє метод безпосереднього обчислення логічного добутку поліномів на основі матричного логічного процесора та відповідних апаратних засобів.

Автором розроблення теоретична основа та запропонований метод реалізації арифметики Галуа на основі каскадного кодування. Суть рішення полягає у тому, що кожен елемент поля в кільці $GF(2^m)$ представляється у вигляді логічного виразу рекурсії відповідної операції над попередніми елементами.

Наприклад, у полі Галуа $GF(2^4)$, з ключем $G_{i+1} = G_i \oplus G_{i-4}$, послідовність елементів додавання $a_0, a_2, \dots, a_{15}$ (1111010110010000), які кодують у базисі Галуа відповідні числа $0, 1, 2, \dots, 15$ записується у вигляді:

$b_1, b_2, b_3, b_4, b_1 \oplus b_4, b_1 \oplus b_2 \oplus b_4, b_1 \oplus b_2 \oplus b_3 \oplus b_4, b_1 \oplus b_2 \oplus b_3, b_2 \oplus b_3 \oplus b_4, b_1 \oplus b_3, b_2 \oplus b_4, b_1 \oplus b_3 \oplus b_4, b_1 \oplus b_2, b_2 \oplus b_3, b_3 \oplus b_4, \emptyset$.

Особливістю такої паралельної арифметики у ТЧБ Галуа є різна форма подання двох операндів. Перший операнд записується у вигляді коду залишків по відповідному модулю P , а другий операнд задається у вигляді відповідного логічного рівняння над залишками $b_1, b_2, \dots, b_4$, причому цей вираз визначає логічні операції по модулю P над кодом першого операнду при обчисленні залишків коду результату сумування кодів Галуа.

Покажемо це на прикладі. Нехай маємо операнди:

$X = 5_{(10)}, Y = 7_{(10)}$. Обчислимо їх суму, використовуючи КППГ $GF(2^4)$, тобто

	b_1	b_2	b_3	b_4
$X =$	1	0	1	1
$Y =$	$b_1 \oplus b_2 \oplus b_3$	$b_2 \oplus b_3 \oplus b_4$	$b_1 \oplus b_2$	$b_2 \oplus b_4$
$X+Y =$	0	0	0	1

Це відповідає коду Галуа числа $12_{(10)}$.

У загальному випадку нумерацію чисел, які відповідають КППГ можна виконувати будь-яким чином.

Наприклад:

$\overbrace{\quad 5 \quad} \quad \quad \quad \overbrace{\quad 13 \quad}$
 0 0 0 0 1 1 1 1 0 1 0 1 1 0 0 1 0 0 0 ...
 $\underbrace{\quad 7 \quad}$
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 0 1 2

У цьому випадку операція додавання двох чисел $X = 5_{(10)}$, то $Y = 7_{(10)}$ буде виконуватися наступним чином:

	b_1	b_2	b_3	b_4
$X =$	1	1	0	1
+				
$Y =$	$b_2 \oplus b_3 \oplus b_4$	$b_1 \oplus b_3$	$b_2 \oplus b_4$	$b_1 \oplus b_3 \oplus b_4$
$X+Y =$	0	0	1	0

Очевидно, що приведена арифметика коду Галуа у полі $GF(2^4)$, однозначно реалізує арифметику 16-ї системи числення з цифрами 0, 1, 2, ..., A, B, C, D, E, F і може бути ефективно використана при асемблерному програмуванні адресних кодів пам'яті, на основі 16-ї системи числення.

Важливими перевагами приведеної операції додавання чисел у кодах Галуа по відношенню до виконання операцій додавання двійкових чисел у базисі Радемахера є наступні:

- відсутність наскрізних переносів між розрядами коду Галуа і можливість розпаралелення та паралельного виконання операцій у всіх розрядах одночасно;
- однаковість числа розрядів представлення чисел у базисах Радемахера та Галуа;
- зменшення апаратної складності та підвищення швидкодії процесорів у базисі Галуа порівняно з процесорами у базисі Радемахера та Крестенсона.

Таким чином у загальному вигляді математична операція сумування чисел в кодах Галуа за довільним модулем P виконується наступним чином:

$$\text{Операнди представляються поліномами } C_{(x)} = \sum_{i=0}^{n-1} c_i x_i \bmod P,$$

де

$$C_{(x)} = (a_{n-1}d_{n-1}^{n-1} + a_{n-2}d_{n-2}^{n-2} + \dots + a_i d_1^{n-1} + a_i d_0^{n-1})x^{n-1} \bmod P + (a_{n-1}d_{n-1}^{n-2} + a_{n-2}d_{n-1}^{n-2} + \dots + a_{n-2}d_{n-2}^{n-2} + \dots + a_1 d_1^{n-2} + a_0 d_0^{n-2})x^{n-2} \bmod P + \dots + (a_{n-1}d_{n-1}^1 + a_{n-2}d_{n-2}^1 + \dots + a_1^2 d_1^1 + a_0 d_0^1)x \bmod P + (a_{n-1}d_{n-1}^0 + a_{n-2}d_{n-2}^0 + \dots + a_1 d_1^0 + a_0 d_0^0)x \bmod P.$$

Для спрощення виразів модульної арифметики при обчислення $C(x)$ доцільно скористатися рівняннями:

$$(Pb_i) \bmod P = 0; [(P+1)b_i \bmod P = b_i], \text{ які слідують з теорії чисел.}$$

Розглянемо приклад операції сумування у полі $GF(3^2)$, яке описує розряди вісімкової системи числення.

З врахуванням останніх рівнянь, складемо таблицю кодів $GF(3^2)$, згідно ключа $G_{i+1} = (G_i \oplus G_{i-1}) \bmod 3$ та виразів їх рекурсії

A=8	$GF\left(\begin{smallmatrix} 2 \\ 3 \end{smallmatrix}\right)$	$b_i \bmod 3$
0	2	b_1
1	2	b_2
2	$X = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$b_1 \oplus b_2$
3		$b_1 \oplus 2b_2$
4	$Y = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	$2b_1$
5		$2b_2$
6	2	$2b_1 \oplus 2b_2$
7	0	$2b_1$

Нехай $X = 2_{(3)}, Y = 4_{(3)}$, тоді

X=	b_1	b_2
+	1	0
Y=	$2b_1$	$2b_2$
X+Y=	2	0

Це відповідає числу $6_{(8)}$

Операція множення чисел $X \cdot Y$ в кодах поля Галуа може бути виконана двома способами:

1. Шляхом x -кратного додавання чисел Y при $Y > X$ або Y -кратного додавання чисел X при $X > Y$;
2. Шляхом послідовного додавання подвоєних значень числа X згідно двійкового представлення числа Y у базисі Радемахера.

Очевидно, що можлива реалізація процесорів у базисі Крестенсона – Галуа в системі взаємо-простих модулів $(p_1 p_2 \dots p_n)$ при обчисленні кореляційних та спектральних функцій та цифрової фільтрації сигналів, які містять базові операції сумування та множення.

РОЗДІЛ 8

ЗМЕНШЕННЯ НАДЛИШКОВОСТІ ДАНИХ В ТЧБ ГАЛУА

8.1. Стиснення даних на основі логіко-статистичних інформаційних моделей та кодів полів Галуа.

Дослідження систем зі стисненням даних може проводитись експериментальним та аналітичним методами. У першому випадку результати дослідження цілком визначаються умовами досліду, параметрами та структурою системи, типом визначеного процесу, які важко змінювати. У зв'язку з цим, більш перспективними є аналітичні методи, хоча вони і вимагають чіткої математичної постановки вихідної задачі: ймовірнісний опис повідомлень, вибір операторів його перетворення, вибір критерію якості.

Реальні фізичні процеси можна описувати на макрорівні, коли використовуються характеристики джерела за великий інтервал часу, або на мікрорівні, коли використовуються короткочасні властивості ДІ, а також на структурному рівні, коли окремі елементи процесу можна класифікувати на основі виділення специфічних ознак. Перші два види опису базуються на представленні процесу у вигляді експериментальної кривої (з неперервним або дискретним часом). Структурний опис дозволяє виявити дискретну природу об'єкту і представити його у вигляді ланцюжка символів із заданого алфавіту. При цьому на вихідний опис джерела впливає характер подальшої обробки повідомлення.

Макроопис доцільно використовувати для процедур стиснення даних фіксованої структури (неадаптивних), мікроопис – для адаптивних алгоритмів, зменшення надлишковості, структурне представлення – для методів, що базуються на синтаксичному підході до розпізнавання образів.

Таким чином, розробка моделі джерела інформації повинна відповідати вимозі універсальності для узгодження вказаних видів опису і адекватно відображати реальні нестационарні фізичні процеси.

Важливу роль для проектування сенсорів та техногенно-екологічних комп'ютерних систем безпеки відіграє побудова в реальному масштабі часу розроблених логіко-статистичних (ЛСІМ) та інших інформаційних моделей.

Особливо це стосується кодування семантичних станів ДІ, коли необхідно відслідковувати критичні ситуації технологічних об'єктів, до яких належать аварії, викиди нафти, коротке замикання.

Кодування нульовими та одиничними бітами унітарного базису вимагає додатково фіксувати час, оскільки ми не можемо відразу визначити момент часу, в який відбулась зміна.

Використання бітів Галуа дозволяє вдосконалити існуючу систему, позбавити її вказаного недоліку. Нульовим значенням ставляться у відповідність прямі біти Галуа, а одиничним – інвертовані. Це дозволяє визначити час, коли відбулася зміна. Для визначення моменту часу необхідно декодувати останніх n_g біт послідовності Галуа, що передують інвертованому біту $\bar{G}$ (де n_g – розрядність кодону Галуа).

Перша ЛСІМ фіксує вихід за границі апертури “норми” амплітудних значень характеристичних параметрів P_{ijk} (рис.8.1) згідно аналітичного виразу:

$$g = \begin{cases} G, & P_{ijk} > E_5 \\ \bar{G}, & P_{ijk} \leq E_5 \end{cases},$$

де E_1 – границя апертури амплітудних значень;

G_i - біт коду Галуа.

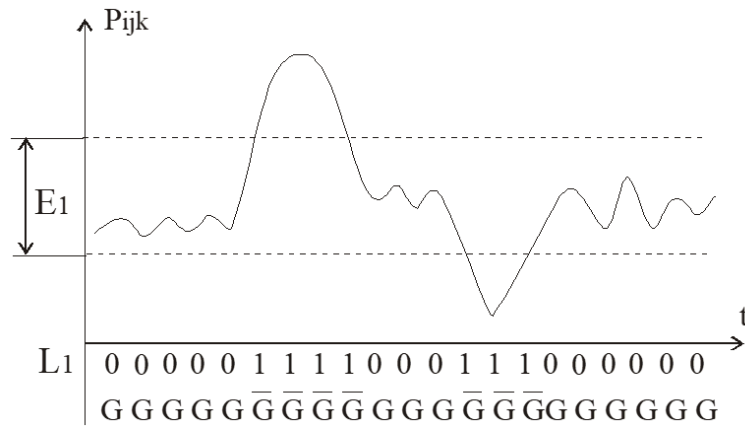


Рис.8.1. Перша ЛСІМ.

Друга ЛСІМ фіксує зміну динаміки характеристичного параметру P_{ijk} , амплітуда якого не виходить за границі апертури E_1 (рис.8.2) згідно наступного аналітичного виразу:

$$L_2 = \begin{cases} G, & E < \varepsilon_0 \\ \bar{G}, & E \geq \varepsilon_0 \end{cases},$$

де $C_{pp}(\tau)$ – автоструктурна кореляційна функція P_{ijk} ; τ – часове зміщення;

$$C_{pp}(\tau) = \frac{1}{n} \sum_{i=1}^n \sum_{j=0}^n (P_{ijk} - (P_{ijk} + \tau))^2.$$

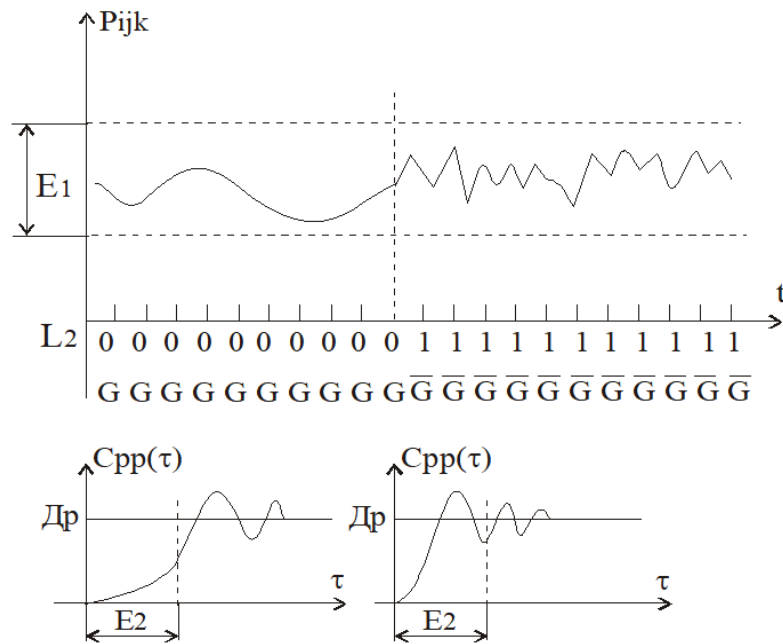


Рис.8.2. Друга ЛСІМ.

Третя ЛСІМ фіксує зміну знаку коефіцієнта взаємкореляції між двома характеристичними параметрами P_{ijk} і q_{ijk} з амплітудами, що не виходять за межі апертури (рис.8.3), один з яких q_{ijk} вважається контрольним або еталонним згідно виразу:

$$L_3 = \begin{cases} G, & \rho_{pq} < 0 \\ \bar{G}, & \rho_{pq} \geq 0 \end{cases}.$$

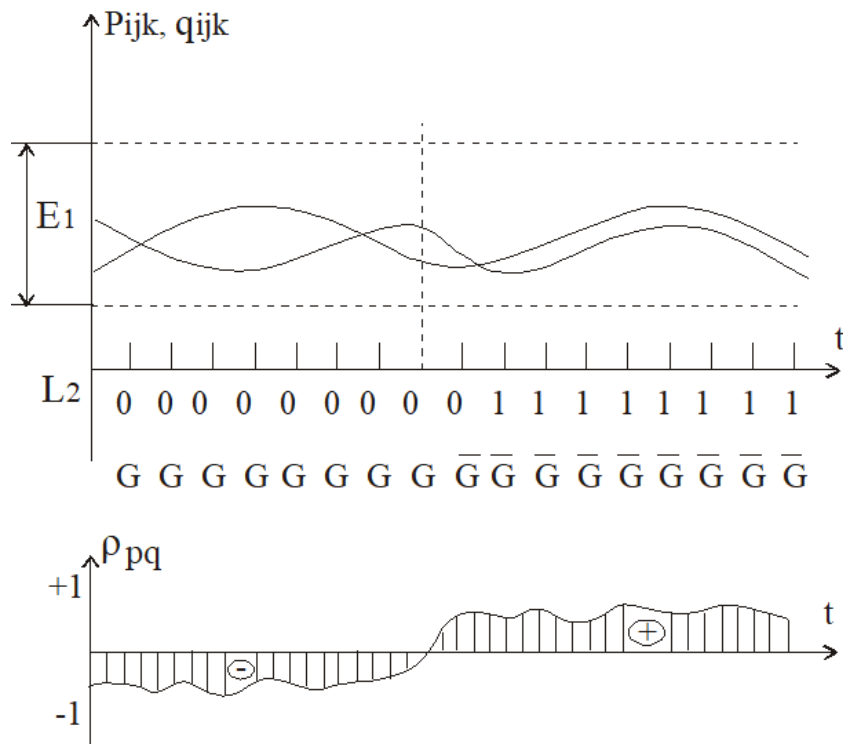


Рис.8.3. Третя ЛСІМ.

Четверта ЛСІМ базується на спектральному аналізі контрольованого сигналу. Множина E_1 визначає спектральний склад сигналу, який відповідає нормальному режиму роботи ДІ (рис.8.4):

$$L_4 = \begin{cases} G, & E_1 = w, \\ \bar{G}, & E_1 \neq w. \end{cases}$$

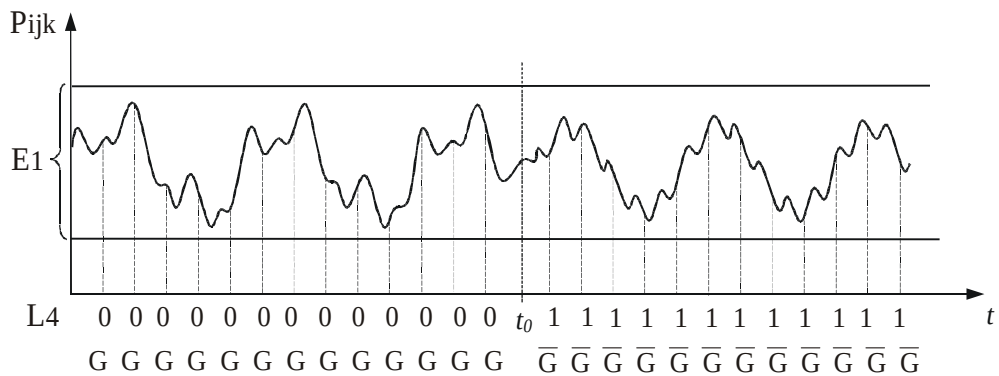


Рис.8.4. Четверта ЛСІМ.

П'ята ЛСІМ базується на матриці коефіцієнтів взаємкореляції сигналів в каналах ДІ і оцінки глобальної дисперсії D_G . Вона дозволяє зафіксувати зменшення значень D_G нижче встановленої апертури E_5 , що відповідає руйнуванню кореляційних зв'язків і деградації системи в цілому. На відміну від попередніх ЛСІМ дана модель виражається одною булевою змінною g (рис.8.5), яка оцінює загальний стан багатоканального ДІ:

$$g = \begin{cases} G, & D_G > E_5 \\ \bar{G}, & D_G \leq E_5 \end{cases}.$$

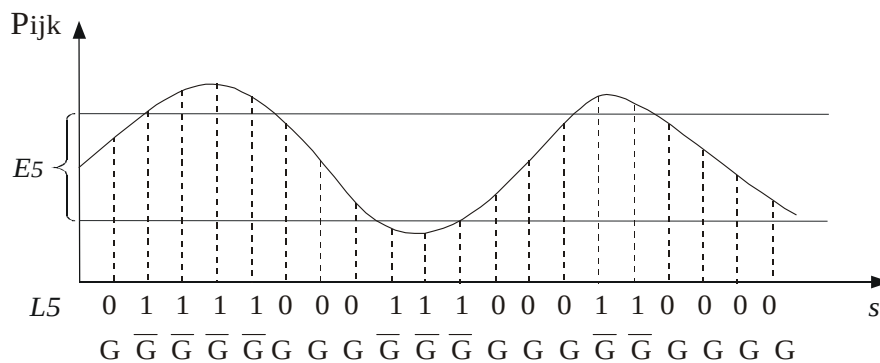


Рис. 8.5. П'ята ЛСІМ на основі глобальної дисперсії.

Шоста ЛСІМ базується на оцінці відхилення значення контрольованого параметру від еталонного. Задається ε_0 , обчислюється величина відхилення сигналу, після цього здійснюється порівняння обчисленого і заданого значення.

У разі, якщо відхилення реального сигналу від передбачуваного не перевищує заданого значення, у канал зв'язку поступають прямі біти Галуа, при перевищенні допустимого відхилення біт Галуа інвертується. Інвертовані біти Галуа вказують на номер відліку (або ж час), коли відбулось дане відхилення. При нормалізації процесу у канал зв'язку знову поступають прямі біти Галуа.

Якщо відомо, що для ДІ на протязі тривалого часу відхилення сигналу не перевищує заданого значення, то доцільно використати модифікований метод, суть якого полягає в наступному: якщо відхилення сигналу є в межах норми, то в канал зв'язку нічого не передається, але біти Галуа постійно генеруються на передавальній стороні, останні n_g бітів записуються в буфер. У кожний момент слідування наступного біту Галуа вміст буферу оновлюється. У разі недопустимого відхилення значень параметра від еталона реєструється інвертований біт Галуа.

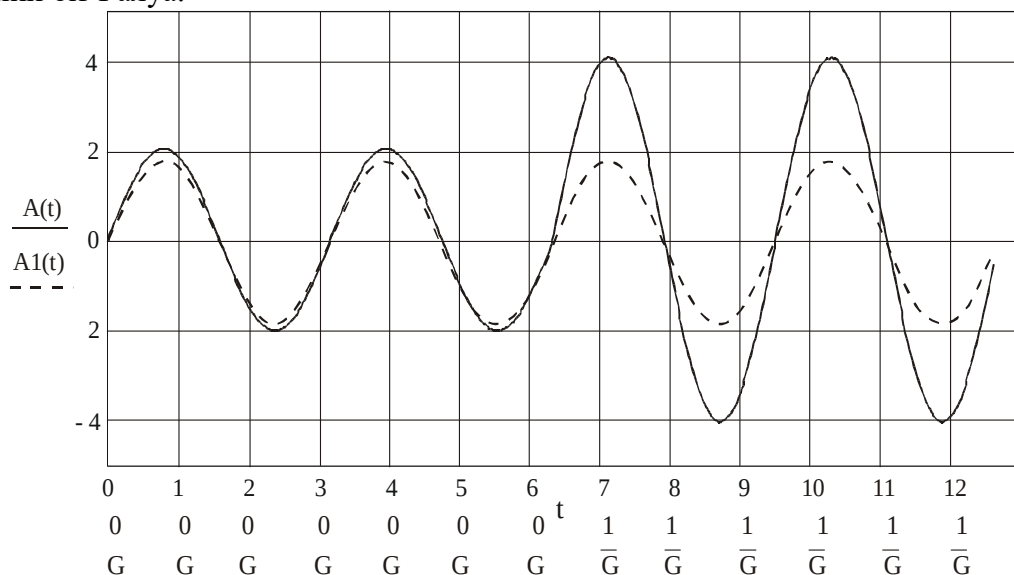


Рис.8.6. Шоста ЛСІМ.

Значення шостої ЛСІМ визначаються за формулою:

$$L_6 = \begin{cases} G, & \varepsilon < \varepsilon_0 \\ \bar{G}, & \varepsilon \geq \varepsilon_0 \end{cases}.$$

В загальному випадку застосування ЛСІМ дозволяє ефективно зменшити обсяг даних. Якщо діапазон квантування контрольованого параметру A , то розрядність коду для представлення одного відліку визначається за формулою:

$$n = \hat{E}[\log_2 A].$$

Початковий об'єм даних при такому представленні:

$$I_0 = m \cdot \hat{E}[\log_2 A],$$

де m – кількість відліків.

Застосування ЛСІМ дозволяє зменшити об'єм даних до величини $I = m$, оскільки для кодування кожного відліку використовується один біт.

Отже, коефіцієнт стиснення дорівнює

$$k_c = \frac{\hat{E}[\log_2 A] \cdot m}{m} = \hat{E}[\log_2 A].$$

Аналогічні ЛСІМ можуть бути побудовані на основі відхилень статистичних, кореляційних, спектральних, ентропійних, кластерних, Хеммінгових та інших інформаційних моделей ДІ.

Важливу інформацію про стан екологічної безпеки несуть інтегральні моделі семантичних та інформаційних станів ДІ. При цьому досягається суттєве збільшення інформації про аномальні перевищення характеристичних параметрів P_{ijk} границь норми при великому числі сенсорів та низькій швидкості їх опитування $\Delta t > \Delta t_0$, де Δt_0 - розрахунковий інтервал часової дискретизації.

8.2. Теорія та застосування базисних функцій кодів поля Галуа.

Найповніше дискретизовані і квантовані функції досліджені в базисі Радемахера. Причому, розроблені теоретичні і методологічні основи генерації базових елементарних функцій: $y_i = n \cdot x_i$; x_i^n ; $\sqrt[n]{x_i}$; $\log_n x_i$; e^{x_i} ; $\cos x_i$; $\sin x_i$ і т. д, а також відповідні структури спецпроцесорів для їх генерації та цифрової обробки.

Метод дельта-модуляції, розроблений академіком Харкевичем, знайшов широке застосування в техніці кодування та стиснення технологічних даних.

В основу принципу дельта-модуляції покладена процедура кодування на основі вибору такого кроку квантування сигналів по рівню δ та кроку дискретизації Δt , при якому виконується умова:

$$\Delta_i = \begin{cases} 1, & x_i - x_{i-1} = +1 \\ 0, & x_i - x_{i-1} = -1 \\ 0, 1, \dots & x_i - x_{i-1} = 0 \end{cases}, \quad (8.1)$$

де x_i - поточне значення відліку стану джерела інформації.

Метод дельта-модуляції не характеризується повнотою базисних функцій, коли градієнт наростання значень X_i на інтервалі дискретності перевищує умову (8.1).

Неповнота системи базисних функцій методу дельта-модуляції обумовлює головний недолік названого методу, який полягає в тому, що при невиконанні умови (8.1) спостерігається відставання або часове запізнення фактичних станів джерела інформації при декодуванні стиснених даних.

В той же час в досліджуваному базисі Галуа така робота не проведена. Тому є актуальною задача розробки каталогу названих дискретизованих і квантованих функцій в базисі Галуа.

Наявність такого каталогу створить основу та допоможе теоретично дослідити потенційні можливості стиснення даних в базисі Галуа.

Розглянемо лінійну функцію $y_i = n \cdot x_i$.

Перші елементарні базисні лінійні функції Галуа нульового порядку $y_i = G_i$ показані на рис. 8.7.

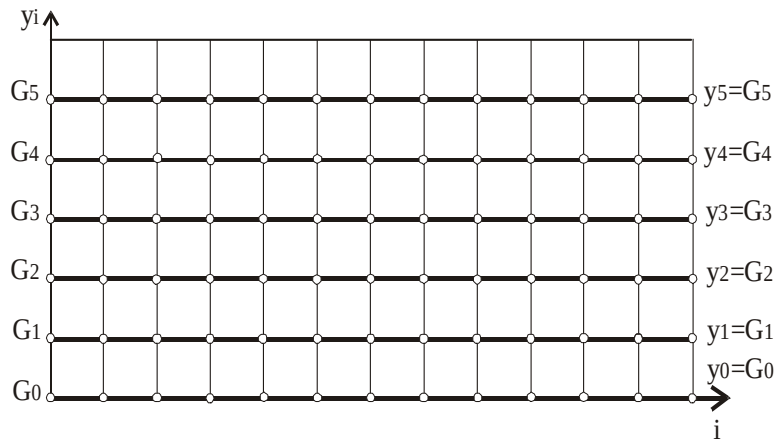


Рис. 8.7. Елементарні базисні лінійні функції Галуа.

З рис 8.7 видно, що існує інваріантність представлення різних функцій однаковими послідовностями бітів Галуа, тому даний метод приводить до неоднозначності кодування аналогічно представленню інтегралів з невизначеною величиною константи C .

Для того, щоб уникнути даного недоліку, кожна з функцій кодується послідовністю Галуа, яка генерується різним ключем: для $n_g = 4$ (табл.8.1).

Таблиця 8.1

Фазові базисні функції.

Назва	Кодон	Послідовність Галуа
G_0	0000	0000101001101111
G_1	1000	1000010100110111
G_2	1100	1100001010011011
G_3	1110	1110000101001101
G_4	1111	1111000010100110
G_5	0111	0111100001010011
G_6	1011	1011110000101001
G_7	1101	1101111000010100
G_8	0110	0110111100001010
G_9	0011	0011011110000101
G_{10}	1001	1001101111000010
G_{11}	0100	0100110111100001
G_{12}	1010	1010011011110000
G_{13}	0101	0101001101111000
G_{14}	0010	0010100110111100
G_{15}	0001	0001010011011110

Такий спосіб кодування інформаційних потоків в базисі Галуа забезпечує коефіцієнт стиснення даних

$$k_{c1} = \hat{E}[\log_2 A],$$

де A – діапазон квантування, при $A=1024$, $k_{c1}=10$.

Застосування даного методу можливо при виконанні умови, що досліджуваний об'єкт перебуває у певному стані на протязі $n_g + 1$ тактів, де n_g – розрядність кодону Галуа.

Без врахування вказаної умови коефіцієнт стиснення визначається за формулою:

$$k_{c2} = \frac{n \cdot m}{(n+1) \cdot f_a + (m - f_a)}.$$

Приклад квазістаціонарного процесу, який однозначно можна представити біт-орієнтованою базисною функцією Галуа $y_i = G_i$, що описується лінією, показаний на рис.8.8.

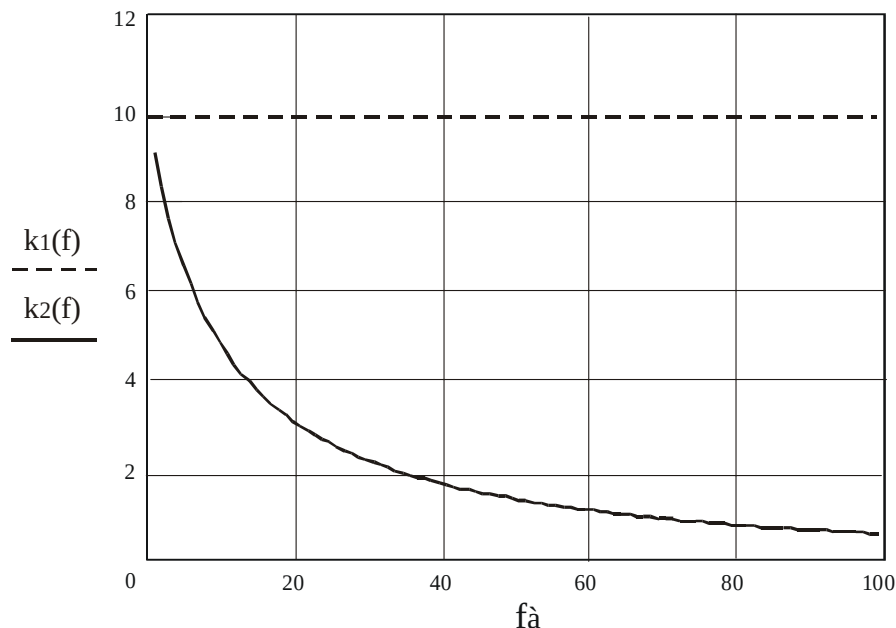


Рис.8.8. Залежність коефіцієнта стиснення від кількості активних відліків.

Практичним застосуванням запропонованого методу є кодування станів об'єктів керування.

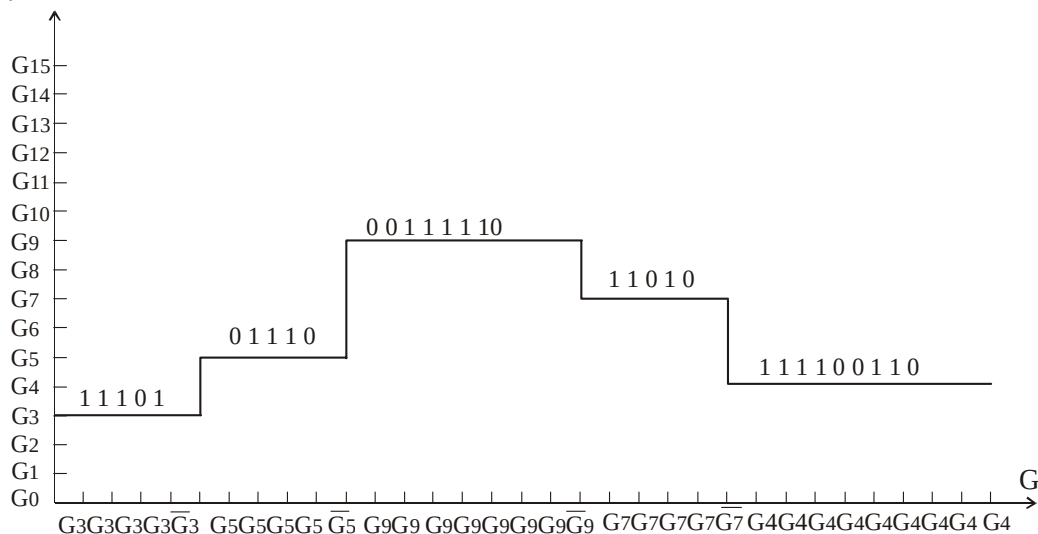


Рис.8.9. Приклад кодування на основі базисних функцій Галуа нульового порядку.

Базисні функції Галуа нульового порядку доцільно використовувати для кодування станів об'єктів керування. Розглянемо приклади: установка буріння має 4 семантичні, 8 технологічних та 5 інформаційних станів. Якщо у відповідність кожному стану поставити певний кодон Галуа, то отримаємо для кожного технологічного стану наступний код:

0001101 –	Буріння;
0010110 –	Аварія;
0110001 –	Викид;
1100010 –	Ліквідація.

Якщо об'єкт керування в даний момент часу перебуває у стані буріння, то послідовність бітів має вигляд 0001101. У разі його переходу в інший стан останній біт, на момент зміни стану вказаної вище послідовності, інвертується 0001100 і генерується послідовність на основі нового кодону, який вказує в якому стані знаходиться ОК в даний момент часу.

Цей метод дозволяє отримати істотне зменшення об'єму інформації. Якщо поставити у відповідність кожному стану ОК певний рівень (1–8), то кожний відлік потрібно кодувати 3 бітами, застосування запропонованого методу дозволяє кодувати 3 бітами тільки перший момент часу, а решту моментів часу представляти одним бітом. Тому на передаючій стороні генератор постійно формує послідовність Галуа, а передає кодон тільки в тому випадку, коли значення контрольованого параметру змінилося, тобто відліки стали активні.

Оскільки технологічні установки, як правило, перебувають у певному стані на протязі тривалого часу, що значно перевищує частоту опитування, то дане кодування є ефективним, причому, чим більша тривалість перебування ОК в кожному стані, тим вища ефективність кодування, тобто більший коефіцієнт стиснення.

У випадку, коли зміна значення функцій відбувається на протязі менше n тактів, необхідно вводити базисні функції Галуа першого порядку (рис.8.10).

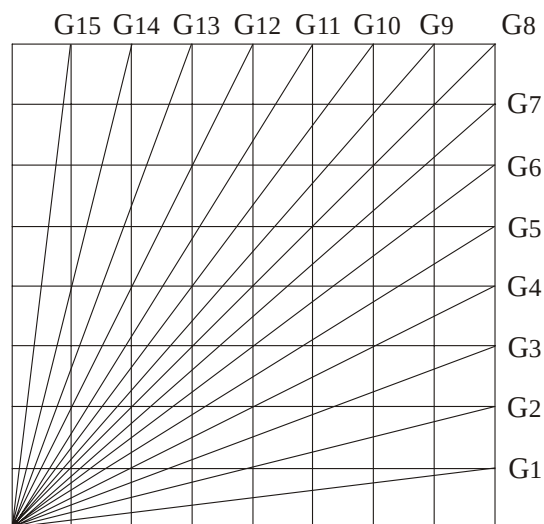


Рис.8.10. Базисні функції Галуа першого порядку.

При цьому біт-орієнтоване кодування дискретних функцій в базисі Галуа однозначно можливе для стрибкоподібних і лінійно-наростаючих функцій, що ілюструється прикладом (рис.8.11).

Кодування даних за допомогою фазових базисних функцій Галуа є більш ефективне, ніж кодування за допомогою різних кодових ключів.

Метод кодування з використанням базисних функцій Галуа першого порядку доцільно використовувати для кодування інтегрованих значень параметрів об'єктів керування. Кут нахилу ліній визначає швидкість зростання інтегралу.

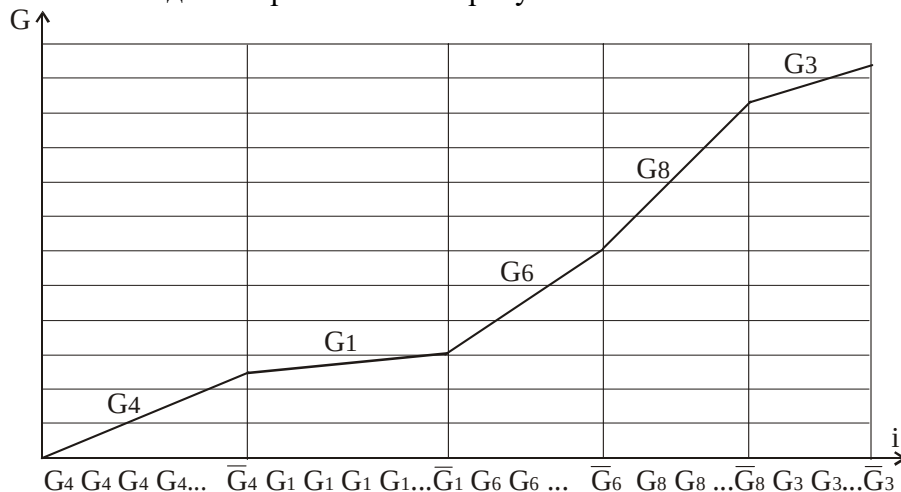


Рис.8.11. Кодування на основі базисних функцій першого порядку.

Кожному куту нахилу відповідає кодон Галуа, що дозволяє однозначно декодувати значення інтегралу функції. Запропонований метод позбавлений недоліку, що полягає у постійному зростанні розрядності відліків, оскільки кут нахилу ліній знаходиться у визначеному діапазоні.

8.3 Стиснення алфавітно-цифрової інформації у базисі Галуа.

Стиснення інформації застосовується для прискорення та зниження витрат на її оброблення, зберігання й пошук, а також для зменшення пам'яті, зайнятої в ЕОМ.

Під стисненням інформації розуміється операція, внаслідок якої певному коду чи повідомленню ставиться у відповідність код або повідомлення меншої довжини

Існує багато практичних алгоритмів стиснення даних. Однак, в основі цих методів лежать три теоретичних алгоритми: алгоритм RLE (Run Length Encoding); алгоритми групи KWE (KeyWord Encoding); алгоритм Хафмана. В основі алгоритму RLE лежить ідея виявлення послідовностей даних, що повторюються, та заміни цих послідовностей більш простою структурою, в якій вказується код даних та коефіцієнт повторення. В основі алгоритму KWE покладено принцип кодування лексичних одиниць групами байт фіксованої довжини. Результат кодування зводиться в таблицю, утворюючи так званий словник. В основі алгоритму Хафмана лежить ідея кодування бітовими групами. Після частотного аналізу вхідної послідовності символи сортуються за спаданням частоти входження. Чим частіше зустрічається символ, тим меншою кількістю біт він кодується. Результат кодування зводиться в словник, що необхідний для декодування [1].

Застосування базису Галуа для стиснення алфавітно-цифрової інформації виконується згідно наступної формули: $G_{j+i} = G_i \oplus G_{i-n}$ (табл.8.3.)

Таблиця 8.3

Приклад коду Галуа з об'ємом коду $V=N$:

1	2	3	4	5	6	7	8
0	0	0	1	0	1	1	1

Використання восьмибітного слова дає змогу закодувати $2^8=256$ знаків, тоді як реальні алфавіти з урахуванням цифр і деяких допоміжних символів містять до 50-60 знаків, тобто для кодування їх потрібні п'яти-шестибітові комбінації та аналогічні структури пам'яті. Спробуємо

зменшити кількість біт, необхідних для кодування одного символу, поділивши всі символи на чотири групи, що показані в табл.8.4.

Останні три біти послідовності Галуа визначають номер групи, в якій знаходиться заданий символ:

00011 - друга група; 00111 - третя група; 01111 - четверта група.

Якщо ми знаходимось в певній групі, а хочемо закодувати символ латинського алфавіту, потрібно ще раз вказати код заданої групи. Для кодування великої літери будемо використовувати послідовність 00000. Позначимо кожен символ 5 бітами Галуа, Кожен наступний символ позначимо бітами попереднього, зсунутого вліво на 1 біт. Комбінацію тих літер, що зустрічаються часто, розташуємо поруч. Тепер замість кожного символу тексту запишемо відповідні 5 біт коду. Якщо наступний символ тексту відповідає наступному символу кодової таблиці, то запишемо відповідний біт, в іншому випадку - інвертований біт Галуа.

Табляця 8.4

Кодова таблиця Галуа з поділом на групи.

1	1	1	1	1	0	0	1	1	0	1	0	0	1	0	0	0	0	1	0	1	0	1	1	1	0	1	1	0	0	0	1
				X	T	h	E	S	i	п	d	0	г	a	m	P	1	У	с	U	ь	j	F	g	W	k	V	Q	z		-
				д	К	а	Й	м	0	в	І	р	н	я	с	т	У	е	ш	ц	л	г	Ч	З	Х	п	и	Ь	б		-
				ф	Щ	ю	Ж	ъ	Ы	Э	Є	ї	ё	=	+	-	0	1	2	3	4	5	6	7	8	9	*	/	A		-
				ч			?	і	·	U	№	%	@	(	)	~	'	U	#	\$	&	1	\	<	>	{	}	[	]		-

Спробуємо даний метод для кодування символів з різних груп. Так, слово Windows98 буде зашифровано наступним чином:

W	G	i	П	d	0	G	W	G	в	G	III группа	9	G	8
00000 01110	0	00110	1	0	0	0	01110	0	10011	1	00111	11101	0	01110

Дане слово займає 72 біт пам'яті. Після упаковки його за допомогою 5-бітової послідовності Галуа воно займатиме 48 біт. Отже $K_c=1,5$.

Мінімальний коефіцієнт стиснення даних, тобто при умові, що в тексті немає жодної пари символів, які б в кодовій таблиці Галуа знаходилися поруч буде розраховуватися за формулою:

$$K_{cmin} = \frac{8 \cdot n}{(6 \cdot n + 5 \cdot n_v + 5 \cdot n_r)} \quad (8.2)$$

де n - кількість символів в файлі, що кодується, n_v — кількість великих символів, n_r - кількість переходів з однієї групи кодової таблиці Галуа в іншу.

Якщо ж враховувати, що в більшості текстів будуть зустрічатися пари символів, які в кодовій таблиці Галуа знаходяться поруч, тоді попередня формула матиме вигляд:

$$K_{cmin} = \frac{8 \cdot n}{(6 \cdot n - 5 \cdot n_p + 5 \cdot n_v + 5 \cdot n_r)}$$

де n_p - кількість символів, які в таблиці Галуа знаходяться поруч.

Для того, що запропонований метод стиснення був більш ефективним, необхідно враховувати статистичні характеристики алфавіту, з якого складаються тексти, які слід обробляти. Бажано також враховувати ймовірність різних сполучень деяких знаків.

Розглянемо приклад застосування базису Галуа для стиснення алфавітно-цифрової інформації.

Нехай заданий наступний текст:

Теорема множення ймовірностей: ймовірність добутку двох подій дорівнює ймовірності однієї з них, помноженій на умовну ймовірність другої при наявності першої. Теорема додавання ймовірностей: ймовірність суми двох подій дорівнює сумі ймовірностей події мінус ймовірність їх добутку.

Проаналізуємо частоту появи складів у цьому тексті. Найчастіше зустрічаються сполучення літер те, ня, вір, ймо, су, ює, до, єї.

Необхідно 255 біт Галуа, щоб позначити всі літери, цифри та символи, що можуть зустрічатися в текстах. Тому прогенеруємо послідовність Галуа з ключем 1 0 0 0 1 1 1 0. Отримаємо:

```

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

```

Позначимо кожен символ 8 бітами Галуа. Кожен наступний символ будемо позначати бітами попереднього, зсунутого вліво на 1 біт. Комбінацію тих літер, що зустрічаються часто, розташуємо поруч. Тепер замість кожного символу тексту запишемо відповідні 8 біт коду. Якщо наступний символ тексту відповідає наступному символу кодової таблиці, то запишемо відповідний біт, в іншому випадку – інвертований біт Галуа. Розглянемо фрагмент кодової таблиці:

Caps Lock								к	а	й	м	о	в	і	р	н	я	с	у	т	є	Д	о	є	ї	л	ь	
1	1	1	1	1	1	1	1	0	0	1	0	0	0	0	1	0	1	0	0	1	1	1	1	1	0	1	0	1

Слово “ймовірність” буде зашифроване наступним чином:

										м	о	в	і	р	н	$\overline{G}$	I											$\overline{G}$					
Й										1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0							
С										$\overline{G}$	Т										$\overline{G}$												
0 0 0 0 1 0 1 0										1	0	0	1	0	1	0	0	1	0														
Ь										$\overline{G}$																							
1 1 1 1 1 0 1 0 0																																	

Дане слово займає 88 біт пам'яті. Після упаковки його за допомогою послідовності Галуа воно займатиме 51 біт. Отже, $K_c=1,73$.

Спробуємо упаковати заданий текст згідно з методом і оцінити коефіцієнт стиснення

До моменту стиснення текст займав 2354 біт. Після стиснення - 1929.

Коефіцієнт стиснення дорівнює:

$$K_c = 2354 / 1929 = 1,22$$

Якщо спробувати упаковати цей самий текст за допомогою архіватора PKZIP, то він займатиме 2216 біт, тобто $K_c=1,06$. Як бачимо, даний метод є доволі ефективним для стиснення текстових даних.

Проведемо деякі дослідження залежності коефіцієнта стиснення від зміни параметрів.

Нехай, задана послідовність чисел, що складається зі 100 символів. Оскільки всі числа, в тому числі й пробіл, належать до однієї групи, то значення змінних $n_v = n_r = 0$. А коефіцієнт стиснення, без врахування кількості розташованих поруч цифр буде рівний $K_c = \frac{8 \cdot n}{6 \cdot n} = \frac{8 \cdot 100}{6 \cdot 100} = 1,33$. Дана величина є сталою, незалежно від кількості символів у файлі.

Якщо ж одна п'ята частина символів заданої послідовності знаходиться поруч в кодовій таблиці, то коефіцієнт стиснення збільшиться до

$$K_c = \frac{8 \cdot n}{6 \cdot n - 5 \cdot n_p} = \frac{8 \cdot 100}{6 \cdot 100 - 5 \cdot 20} = 1,6$$

Як бачимо, ефективність стиснення збільшилась.

Розглянемо, як буде змінюватися коефіцієнт стиснення K_{cmin} при зміні параметрів. Якщо кількість великих літер та кількість зміни групи символів буде незмінною, а кількість символів в тексті буде зростати, то коефіцієнт стиснення буде збільшуватися. Проаналізувавши формулу (8.2) отримаємо, що ефективність стиснення буде досягнута, при умові $5 \cdot n_v + 5 \cdot n_r \leq 2 \cdot n$. Як видно з рис.8.12, якщо $n_v = n_r = 10$, то $K_{c1} > 1$, при $n > 50$.

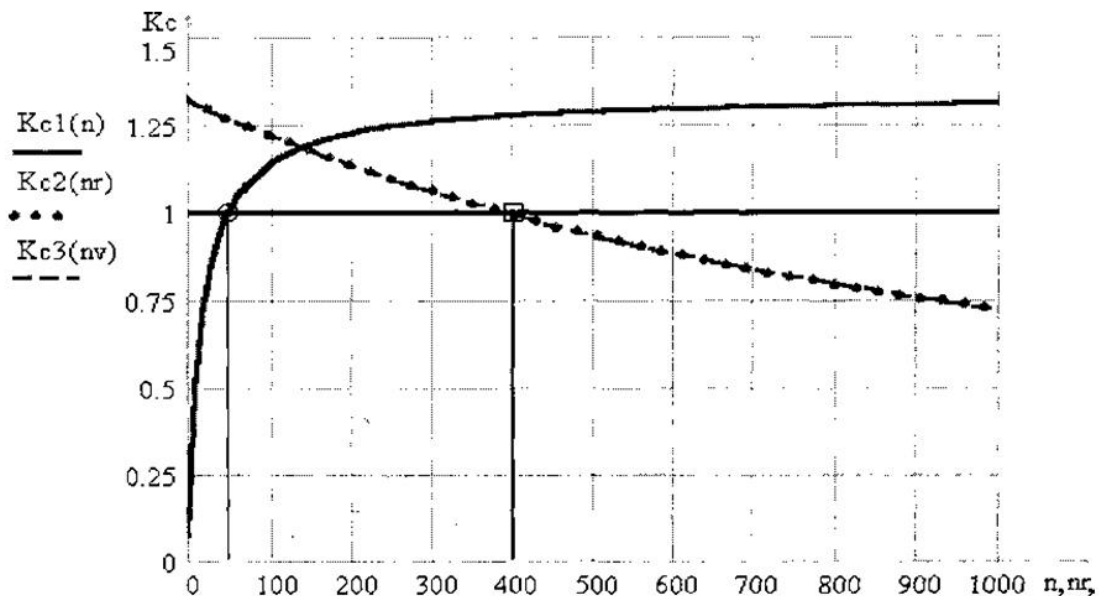


Рис.8.12. Залежність коефіцієнта стиснення від кількості символів в текстовому файлі (K_{c1}), від кількості змін групи символів (K_{c2}) та кількості великих літер (K_{c3}).

Коефіцієнти K_{c2} та K_{c3} , що представлені на рис.8.12, показують залежність коефіцієнта стиснення від кількості змін групи символів та кількості великих літер. Як бачимо, вони рівні і оберненопропорційно залежать від заданих параметрів. Якщо $n=1000$, то $K_{c1} > 1$, при n_v або $n_r < 400$.

На рис.8.13 представлено залежність коефіцієнта стиснення від кількості символів при умові, що в тексті немає жодної пари символів, які б в кодовій таблиці Галуа знаходилися поруч (K_{c1}) та залежність коефіцієнта стиснення від кількості символів, які в таблиці Галуа знаходяться поруч (K_{c2}). Аналіз отриманих залежностей дозволяє зробити висновки: якщо $K_{c1} > 1$ при умові $5 \cdot n_v + 5 \cdot n_r \leq 2 \cdot n$ і зі збільшенням n наближається до величини 1,33, то $K_{c2} > 1$ при умові $5 \cdot n_v + 5 \cdot n_r - 5 \cdot n_p \leq 2 \cdot n$. Якщо ж кількість n_p досягає значення $0,5n$, то ефективність стиснення збільшується в 2,25 раз.

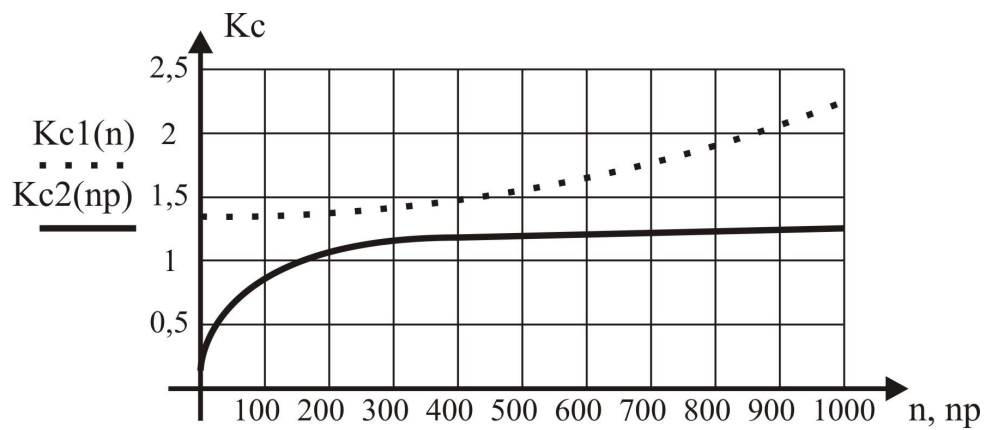


Рис.8.13.Залежність коефіцієнта стиснення від кількості символів (K_{c1}) та від кількості символів, які в таблиці Галуа знаходяться поруч (K_{c2}).

Отже, для одноalfавітних текстових даних з мінімальною кількістю великих літер та максимальною кількістю символів, розташованих поруч в кодовій таблиці даний метод дає найбільший ефект.

Описаний метод є більш ефективним для одноalfавітних з невеликою кількістю великих літер та максимальною кількістю символів, розташованих поруч у кодовій таблиці Галуа.

РОЗДІЛ 9

ХАРАКТЕРИСТИКИ КОРЕКТУЮЧИХ КОДІВ

9.1. Класифікація класів коректуючих кодів

Коди, які виявляють і виправляють помилки комп'ютерів зарекомендували себе, як зручний об'єкт, ніж системи передачі даних, оскільки причини появи помилок тут є збої і відмова апаратури, і у випадку відсутності джерела зовнішніх перешкод можна впевнено прогнозувати характер і конфігурації очікуваних помилок. Одним із шляхів вирішення задачі підвищення правильності передавання даних на сьогоднішній час є використання спеціальних процедур, заснованих на застосуванні перешкодостійких (коректуючих) кодів [3,4].

Класи кодів. Коди виявлення і виправлення помилок: коди поділяються на перешкодонестійкі та перешкодостійкі (коректуючі) коди, показані на рис.9.1.

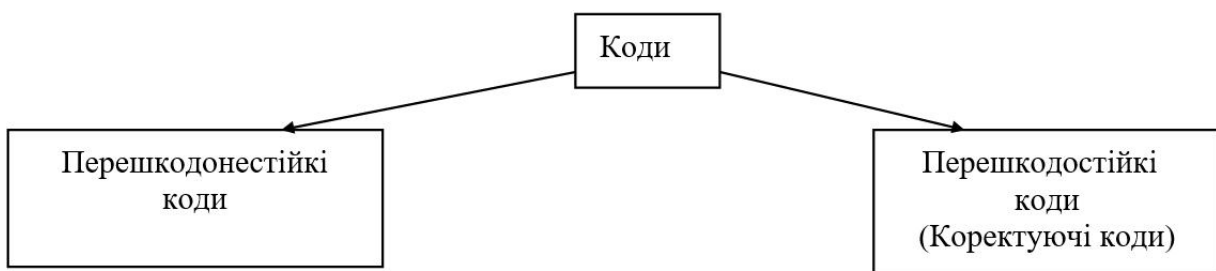


Рис. 9.1. Класи кодів

Перешкодонестійкі коди: двійковий код, одинично-десятковий (число-імпульсний) код, двійково-десятковий код, код ASCII, код Грея, штрих-коди, показані на рис.9.2.

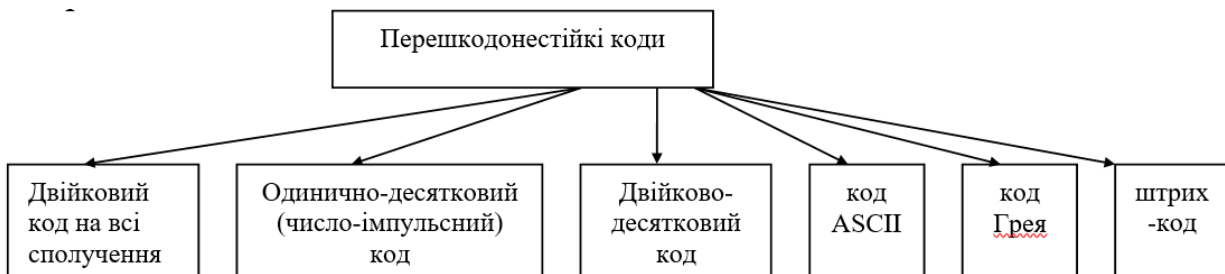


Рис. 9.2.Перешкодонестійкі коди

Перешкодостійкі (коректуючі) коди можна розділити на два великих класи: блокові коди і неперервні – згортаючі коди, показані на рис.9.3., а також коректуючі коди можна розбити на два підкласи: виправляючі незалежні помилки і виправляючі пакети помилок (рис.9.3) [3,4].

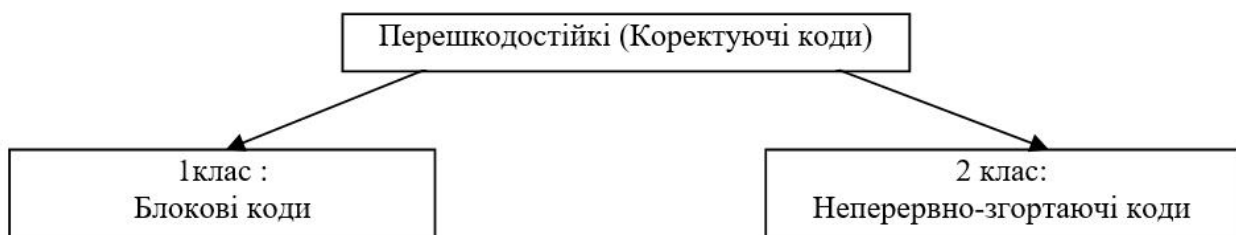


Рис. 9.3. Перешкодостійкі (коректуючі коди)

Класифікація перешкодостійких (коректуючих) кодів

ПСК– коди, які дозволяють виявляти і (або) виправляти помилки в кодових словах, котрі виникають при передачі по каналах зв'язку. Відомо велике число ПСК, які класифікуються за різноманітними ознаками. ПСК можна розділити на два великих класи: БК і НРК, ЗГК. Класи розділяються на підкласи, підкласи на типи кодів. На рис.9.10 показана спрощена класифікація ПСК. ЛБК, в яких інформаційний потік символів розбивається на відрізки і кожен із них перетворюється у визначену послідовність (блок) кодових символів. У БК кодування при передачі (формування перевірочних елементів) і декодування при прийнятті (виявленні і виправленні помилок) виконуються у межах кожної кодової комбінації (блока) окремо за відповідним алгоритмом. Неперервні (рекурентні) коди утворюють послідовність символів, не розділену на окремі кодові комбінації. Кодування і декодування неперервно відбувається на послідовності елементів без ділення їх на блоки. Формування перевірочних символів ведеться за рекурентними (зворотнім) правилам, тому неперервні коди часто називаються рекурентними ланцюговими. [5,7]

1 клас: Блокові коди

При блочному кодуванні послідовність елементарних повідомлень джерела розбивається на відрізки і кожному відрізку ставиться у відповідність визначена послідовність (блок) кодових символів, що називається звичайною

ковою комбінацією. Множина всіх кодових комбінацій, можливих при даному способі блочного кодування, і є блоковим кодом. В блокових кодах послідовність елементарних повідомлень розбиваються на блоки символів ($B_1, B_2, B_3, \dots, B_n$) фіксованої довжини K , кожному з яких ставиться в відповідності певна комбінація символів кодового слова ($b_1, b_2, b_3, \dots, b_n$). Блокові коди діляться на рівномірні і нерівномірні, показані на рис.9.4.

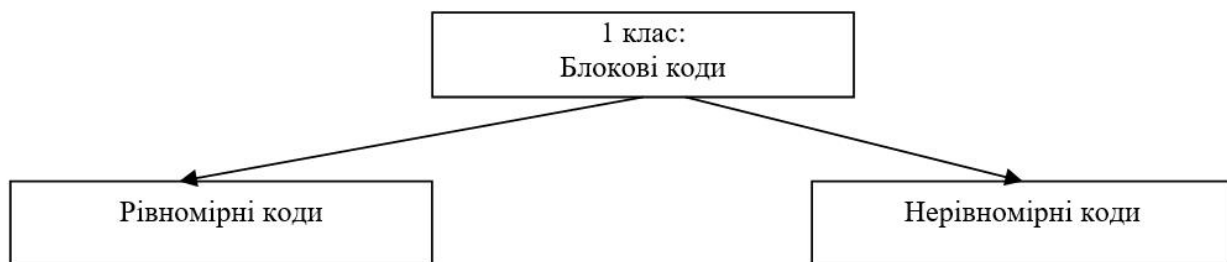


Рис. 9.4. 1 клас: блокові коди.

В рівномірних кодах, на відміну від нерівномірних, всі кодові комбінації містять однакове число n -символів (розрядів) з постійною тривалістю t_0 імпульсів – символів коду. Рівномірні коди в основному застосовуються в системах зв'язку, так як це спрощує техніку передачі і прийому даних. Блокові коди поділяються на рівномірні коди та нерівномірні – коди Морзе.

Рівномірні коди поділяються на роздільні і нероздільні – рівновагові коди, показані на рис.9.5.



Рис. 9.5. Рівномірні коди

Роздільні коди поділяються на систематичні (групові) коди лінійні загального виду і на несистематичні нелінійні коди – рівновагові, ітеративні, каскадні коди, турбо – коди, коди Бергера, Адамара, Хаара. Ітеративні коди дозволяють отримати так звані потужні коди, тобто коди з довгими блоками і великою кодовою відстанню при порівняно простій процедурі

декодування. Ітеративні коди можуть будуватися як комбінаційні безпосередньо добутками двох або більше систематичних кодів, показані на рис.9.6. [5].

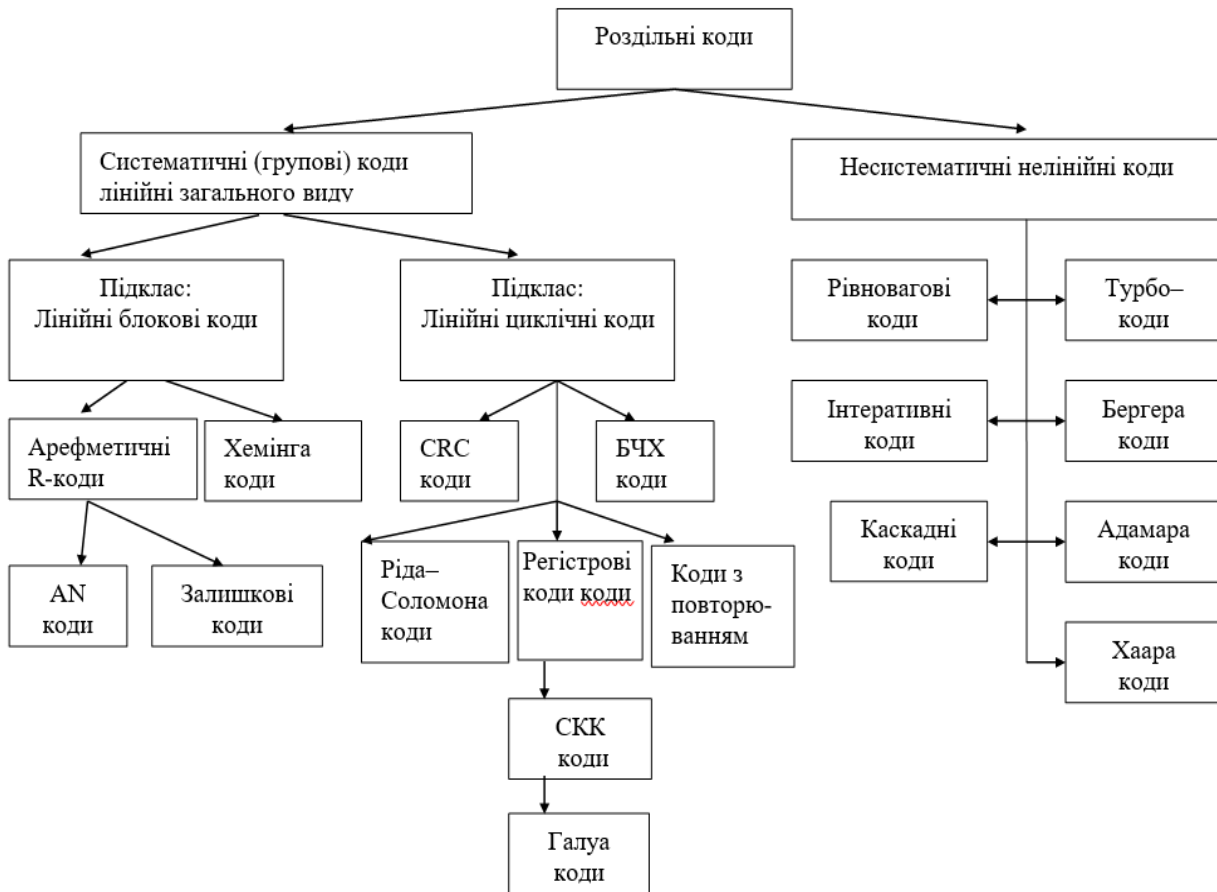


Рис. 9.6. Роздільні коди

Підкласи кодів.

ЛКЗВ, поділяються на два підкласи:

Підклас: ЛБК -коди з визначенням і виправленням помилок. В свою чергу поділяються на коди з по елементним формування перевіркою групи – ЛАК: AN- коди і залишкові коди , коди Хеммінга $[d \min=2]$, $[d \min=3]$, $[d \min=4]$ [6].

Підклас: ЛЦК – з визначенням і виправленням помилок. ЛЦК: утворюючі поліноми, формування кодової послідовності; декодування; визначення помилок; виправлення помилок. Коди: CRC , БЧХ, РСК, РК, СКК, КП, КПП.

Типи перешкодостійких (коректуючих) кодів.

Ці коди будуються таким чином, що для передачі повідомлення використовується лише частина кодових слів, котрі відрізняються один від

другого більш ніж в одному символі. Ці кодові слова називаються дозволеними. Всі решта кодові слова не використовуються і відносяться до числа заборонених. Застосування перешкодостійких кодів для підвищення правильності передачі даних зв'язано з розв'язанням задачі кодування і декодування. Прості коди характеризуються тим, що для передачі інформації використовується всі кодові слова (комбінації), кількість яких рівна $N=q^n$ (q - основа коду, а n - довжина коду). В загальному випадку вони можуть відрізнятися один від другого одним символом (елементом). Тому навіть один помилково прийнятий символ приводить до заміни одного кодового слова іншим і, відповідно, до неправильного прийому повідомлення в цілому. Розрізняють такі типи ПСК 1 класу: БК: підклас: ЛБК, АК, КХ, підклас: ЛЦК, CRC, БЧХ, СКК, НРК 2 класу: неперервні коди: ЗГК [7,8].

Лінійні коди загального виду.

Підклас: ЛБК– такий код, що чисельність його кодові слова утворюють k -мірний лінійний підпростір C в n -мірному лінійному просторі, ізоморфний простір k -бітних векторів.

Це значить, що операція кодування відповідає множенню вихідного k – бітного вектора на невироджену матрицю G , називається порожньою матрицею.

Арифметичні коди

Для виявлення і виправлення помилок в роботі виконавчих пристроїв були створені спеціальні коди, які прийнято називати арифметичними. Відомі два найбільш вивчені конструкції арифметичних кодів: AN –коди і залишкові коди. В загальному випадку використання AN –кодів пропонує попередньо множення вихідних чисел на деякі заздалегідь вибрані числа. Виконання арифметичної операції над числами виду AN і наступне декодування, виправлення або виявлення помилок, виконується шляхом визначення по модулю A . AN –код формується цілими числами $[6,7] \ 0 \leq N_i < B$ де A – позитивне ціле число, яке називається породжуючим числом, або генератором кода, а B визначає кількість чисел в коді, або потужність кода. Цілі числа, які належать коду, називаються кодовими словами; AN_i , $i=1,2,\dots,B$, є кодовим словом для кодованого числа N_i , отримане множенням останнього на генератор A . Легко бачити, що $AN_1 + AN_2 = A(N_1 + N_2)$ тобто сума кодових слів для N_1 і N_2 є кодовими словами сумами цих чисел (якщо $N_1 + N_2 < B$); тому декоди AN – коди називають ЛАК. ЗК – систематичні. Арифметичні операції виконуються паралельно над вихідними операндами і їх найменшими залишками по деяких вибраних модулях, і після завершення операцій обчислюються залишок результатів операції по цим модулям порівнюють результати операції по цим модулям зрівнюється з результатами аналогічних операції над відрахуваннями операндів. Можливе неспівпадіння вказує на існування помилок в отриманому результаті. Паралельно з розвитком теорії AN – кодів увагу привертала інша конструкція арифметичних кодів, витoki якої слід шукати в ранній роботі Пітерсона. Коди, які в кінці кінців отримали звання залишкових, були запропоновані Ю.Г.Дадаєвим і наслідок в окремому вигляді перевідкриті Рао[].

Захист даних надлишковим кодуванням.

Арифметичні коди – R –коди. Базуються на системі числення залишкових класів. $F(x)=G(x) \cdot b(x)$ якщо дописуємо один залишок, можна лише виявити помилки, а подвійна передача дозволяє виправити помилки [11,12,13]. Для реалізації кодів, які виявляють і виправляють помилки комп'ютерів зарекомендували себе, як зручний об'єкт, ніж системи передачі даних, поскільки причини появи помилок тут є збої і відмова апаратури, і у випадку відсутності джерела зовнішніх перешкод можна впевнено прогнозувати характер і конфігурації очікуваних помилок. Одним із шляхів вирішення задачі підвищення правильності передавання даних на сьогоднішній час є використання спеціальних процедур, заснованих на застосуванні ПСК [3,4].

Хеммінгові коди

В цих КХ використовується принцип перевірки на парність визначеного ряду інформаційних символів. Перевірювальна група з r символів формується по елементно у відповідному алгоритму. Коди, з $d_{min}= 3$, дозволяють виправляти одну помилку. Розширені коди будуються в результаті доповнення кодів $cd_{min}= 3$ шляхом загальної перевірки кожної з кодових комбінацій на парність, тобто ще одним перевірювальним символом. Це дозволяє збільшити мінімальну кодову відстань до $d_{min} = (4,6, 9,10)$.

Циклічні-коди

Теорія побудови ЛЦК базується на розділах вищої алгебри, яка вивчає властивості двійкових многочленів. Особливу роль в цій теорії відіграє так звані незвідні багаточлени, тобто поліноми, які не можуть бути представлені у вигляді добутку многочленів нижчих ступенів [9].

Для підкласу ЛЦК можна записати відповідний їм аналітичний вираз, чи деяке логічне співвідношення, яке визначається правилами створення цих кодів. Найбільш зручною формою представлення ЛЦК – використання алгебраїчного виразу [84].

$$G(x) = a_{n-1} \times x^{n-1} + \dots + a_1 \times x + a_0$$

де a_i – числа, що дорівнюють «0» чи «1», які визначають відповідні значення розрядів кодових комбінацій. Таким чином дія над ЛЦК зводиться до дії над відповідними

математичними виразами. Коефіцієнти однакових степенів додаються за модулем 2. Дані коди є одними з найбільш досконалою упаковкою інформації.

Таблиця 9.1.

Поліноми. Лінійні циклічні коди		
Назва кода	Ступінь	Поліноми
CRC-12	12	$x^{12} + x^{11} + x^3 + x^2 + x + 1$
CRC-16	16	$x^{16} + x^{15} + x^2 + 1$
CRC-CCITT	16	$x^{16} + x^{12} + x^5 + 1$
CRC-32	32	$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$

Коди CRC

Коди CRC (ceclic redundancy check–циклічна надлишкова перевірка) є систематичними кодами, призначеними не для виправлення помилок, а для їх

виявлення. Вони реалізуються способом систематичного кодування, викладений вище : «контрольна сума» обчислюється шляхом ділення $x^n - k_u(x)$ на $g(x)$. В зв'язку з тим, що виправлення помилок не вимагається , перевірка правильності передачі може виконуватися так само. Таким чином, вид полінома $g(x)$ задає конкретний код CRC[10] .Приклади найбільш популярних поліномів, табл.9.1.

БЧХ коди

Коди БЧХ- є підкласом двійкових циклічних кодів. Їх відмінні властивості – можливість побудови кода БЧХ з мінімальною відстанню не менше заданого . Коди відрізняються спеціальним вибором породжуючого циклічний код полінома, що приводить до простої процедури декодування. В циклічних кодах " r " перевірочних символів, доданих до початкового "k" інформаційним, можуть бути отримані відразу, тобто в результаті множення початкової, кодової комбінації Q(x) простого кода на одночлен x^r і доданих до цього добутку залишку R(x), отриманого в результаті ділення добутку на породжуючий поліном P(x) [7,8,10].

Типи заводо захищених кодів:

Систематичні коди (коди Хемінга) $X \oplus Y$. Циклічні коди (БЧХ) Шляхом зсуву кодової комбінації, коди базису Галуа. $X \rightarrow Y$. Арифметичні коди X, $\text{res}X(\text{mod}P)$.

Для будь-яких цілих чисел m і t існує код з довжиною кодової комбінації $n=2^{m-1}$ і мінімальною кодовою віддаллю між комбінаціями $d=2t+1$. При цьому число надлишкових символів не перевищуватиме $mt=n-k$ де k – число надлишкових символів. Даний код дозволяє виявити пачки помилок довжиною $n-k < r$ виявляє 2t кратні помилки і дозволяє t кратні помилки. Параметри заводо захищених систем. При використанні групових КХ надлишковість зменшується при використанні блоків даних [10,12].

Таблиця 9.2

Ефективність БЧХ кодів

Групові	n	3	7	15	31	63	127
Інформ. Розр.	m	1	4	11	26	57	120
Контр	k	2	3	4	5	6	7

Таблиця 9.3

Групові	n	3	7	15	31	63	127	511	1023
Інформ. Розр.	m	2	3	11	25	57	120	502	1013
Контр	k	1	4	4	6	6	7	9	10

БЧХ коди є основними для захисту від помилок. Процедура утворення: Циклічний БЧХ утворюється логічним множенням n -елементного коду (n -число інформаційних розрядів) $G(x)$ на утворюючий поліном $P(x) \otimes G(x) \equiv F(x)$ $G(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \dots + a_n$. $P(x)$ - неприводимі поліноми, які утворюються:

$$\begin{aligned} P_1 &= x+1; \\ P_2 &= x^2+x+1; \\ P_3 &= x^3+x^2+1 \\ P_4 &= x^4+x+1 \\ P_5 &= x^4+x^3+1 \\ P_6 &= x^4+x^3+x^2+x+1 \end{aligned}$$

Приклад захисту коду БЧХ $F(x) = x^{19} + x^{10} + x^7 + x^3 + (x+1)$
 $F(x) = 1000100101\ 00011$

Коди корекції помилок Ріда-Соломона

РСК – недвійкові циклічні коди, дозволяють виправляти помилки в блоках даних. Елементами кодового вектора є не біти, а групи бітів (блоки). Дуже поширені РСК, які працюють з байтами (октетами). Математичні РСК близькі до кодів БЧХ. Хоча блокові коди, як правило, добре справляються з рідкими, але великими пачками помилок, їх ефективність при частих, але невеликих помилках. Досліджувалися дві конструкції складових кодів, які умовно можна позначити наступним чином

$$A_0 N, (A_0 N \bmod A_1), A_{i+1}, (A_0 N \bmod A_2) A_{i+2}, \dots, (A_0 N \bmod A_i) A_{i+m} \} \quad [7,8].$$

Регістровані коди

Коди РГК максимального періоду, причому і в арифметичному варіанті залишаються еквідистантними (визначеними для арифметичних кодів)[7].

Коди з виправленням помилок.

Складені і композиційні коди

В основі конструкції СКК лежить ідея побудови дециклічних кодів для передачі повідомлень. Множина слів виду $(C, f_1, f_2, \dots, f_i)$ названо складовим арифметичним кодом.

Мінімальна кодова відстань такого коду визначається як мінімальна сума ваги компоненти слів кода, тому що побудований код є систематичним [6,7]. Досліджувалися дві конструкції складових кодів, які умовно можна позначити наступним чином

$$\{A, N, A_0 N \bmod A_1, A_1 N \bmod A_2, \dots, A_{i-1} N \bmod A_i\},$$

Коректуючі коди поля Галуа

КППГ за загальною класифікацією відносяться до 1 класу блокових кодів підкласу лінійних циклічних кодів, які володіють усіма основними властивостями завадозахищених кодів. КППГ – службові коди для виявлення або виправлення помилок, виникаючих при передачі інформації під впливом перешкод, а також при їх зберіганні.

КППГ – такі коди, в яких лише деяка частина всіх можливих 2^n комбінацій, отриманих з допомогою n двійкових знаків, використовується для представлення інформації. [10] Для генерування КППГ використовуються не приводимі алгебраїчні багаточлени [12,12]. Важливою перевагою кодової послідовності Галуа, є їх проста генерація на основі рекурентного рівняння.

Найпростіші ключі кодів полів Галуа описуються виразом (9.1) та представлені в табл.9.4.

$$G_{i+1} = G_i \oplus G_{i-(n-1)} \quad (9.1)$$

де n – довжина кодона,

$\oplus$ – операція додавання „за модулем два”

при $n = 4$ отримаємо код (рис.9.7):

Рис. 9.7. Формування коду Галуа при $n=4$ (*- біт стаффінг).



Арифметичні операції над кодами поля Галуа:

$$\begin{array}{r} \text{xi} = (1100) \ 1 \\ \underline{1} \\ 1001 \end{array}$$
$$\begin{array}{r} \text{xi} = \begin{array}{c} \text{0(1100)} \\ + \quad \quad \quad \text{1} \\ \hline \text{0110} \end{array} \end{array}$$

$C1 = f(a1, a2, a3, a4, b1)$ f – складання по модулю 2 ($\oplus$):

$$\begin{array}{cccc} x_i = & (a_1, & a_2, & a_3, & a_4) \\ x_j = & (b_1, & b_2, & b_3, & b_4) \\ \hline & C_1 & C_2 & C_3 & C_4 \end{array}$$

$$\begin{array}{rcl} x_i & = & (a_1, a_2, a_3, a_4, \dots, a_n) \\ x_j & = & (b_1, b_2, b_3, b_4, \dots, b_n) \\ \hline & & \begin{array}{ccccccc} C_1 & C_2 & C_3 & C_4 & \dots & C_n \end{array} \end{array}$$

4. Операція ділення відсутня!

КПП належать до класу рекурентних кодів, які широко використовують для захисту інформації від несанкціонованого доступу [17]. Для генерування кодів поля Галуа потрібно застосовувати ключі кодів поля Галуа.

2 клас: Неперервні (рекурентні) коди

Згортаючі коди. Згортаючий кодер ($k=7$, $R=1/2$). ЗГК, на відміну від блокових, не ділять інформацію на фрагмент і працюють з нею як із суспільним потоком даних. ЗГК, як правило, породжуються дискретною лінією інваріантною в часі системою. Тому, на відміну від більшості блокових кодів, згортаюче кодування – дуже проста операція, чого неможна сказати про декодування.

ЗГК ефективно працюють в каналі з білим шумом, но погано справляються з пачками помилок. Більш того, якщо декодер помиляється, на його виході завжди виникає пачка помилок. ЗГК працюють з безперервним потоком даних, кодуючи їх за допомогою регістрів зсуву з лінійним зворотним зв'язком. Декодування ЗГК проводиться, як правило, за допомогою алгоритму Вітербі. [10]

Нерівномірні коди.

Класичним прикладом нерівномірного коду є МВГ, широко застосований у минулому в телеграфії, і ХК, застосований для компресії інформації (факсимільний зв'язок, ПК). Ніяких спеціальних заходів з виправленням і виявленням помилок в МВГ не передбачається в зв'язку з великою надлишковістю самого передаваного тексту. В цьому розумінні МВГ не відноситься до класу коректуючих кодів, показано на рис.9.9.

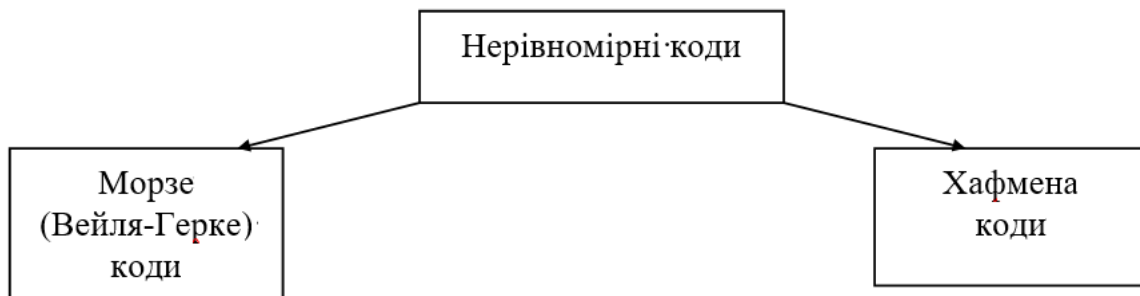


Рис. 9.9. Нерівномірні коди

Нероздільні–рівновагові коди



Рис. 9.10. Нероздільні–рівновагові коди

2 клас: Неперервні (рекурентні) коди

НРК характеризуються тим, що операції кодування и декодування виконується над неперервною послідовністю символів без розбиття її на блоки. Серед неперервних найбільше застосовуються згортаючі коди, показані на рис.9.8. Згортаючі коди ефективно працюють в каналі з білим шумом, але погано справляються з пачками помилок. Більш того, якщо декодер помиляється, на його виході завжди виникає пачка помилок. Згорткові коди працюють з безперервним потоком даних, кодуючи їх за допомогою регістрів зсуву з лінійним зворотним зв'язком. Декодування згорткових кодів проводиться, як правило, за допомогою алгоритму Вітербі.



Рис. 9.11. Неперервні (рекурентні) коди

Кодування даних

Кодування – це процес перетворення повідомлення на впорядкований набір символів, елементів, знаків. Способи кодування даних – визначається теоретико – числовими базисами які застосовуються для їх представлень, найбільш широко вживаними ТЧБ сучасних КС є наступні базиси: Унітарний, Хаара, Крейда, Радемахера, Грея, Крестенсона, Уолша, Галуа [62]. Кодування поділяється на одноелементне кодування, багатоелементне кодування, надлишкове кодування. Задача кодування завершується в отриманні при передачі для кожної k - елементної комбінації із чисельності q^k відповідного їй кодового слова довжиною n і чисельності q^n .

Схема класифікація перешкодостійких (коректуючих) кодів представлена на рис.9.12.

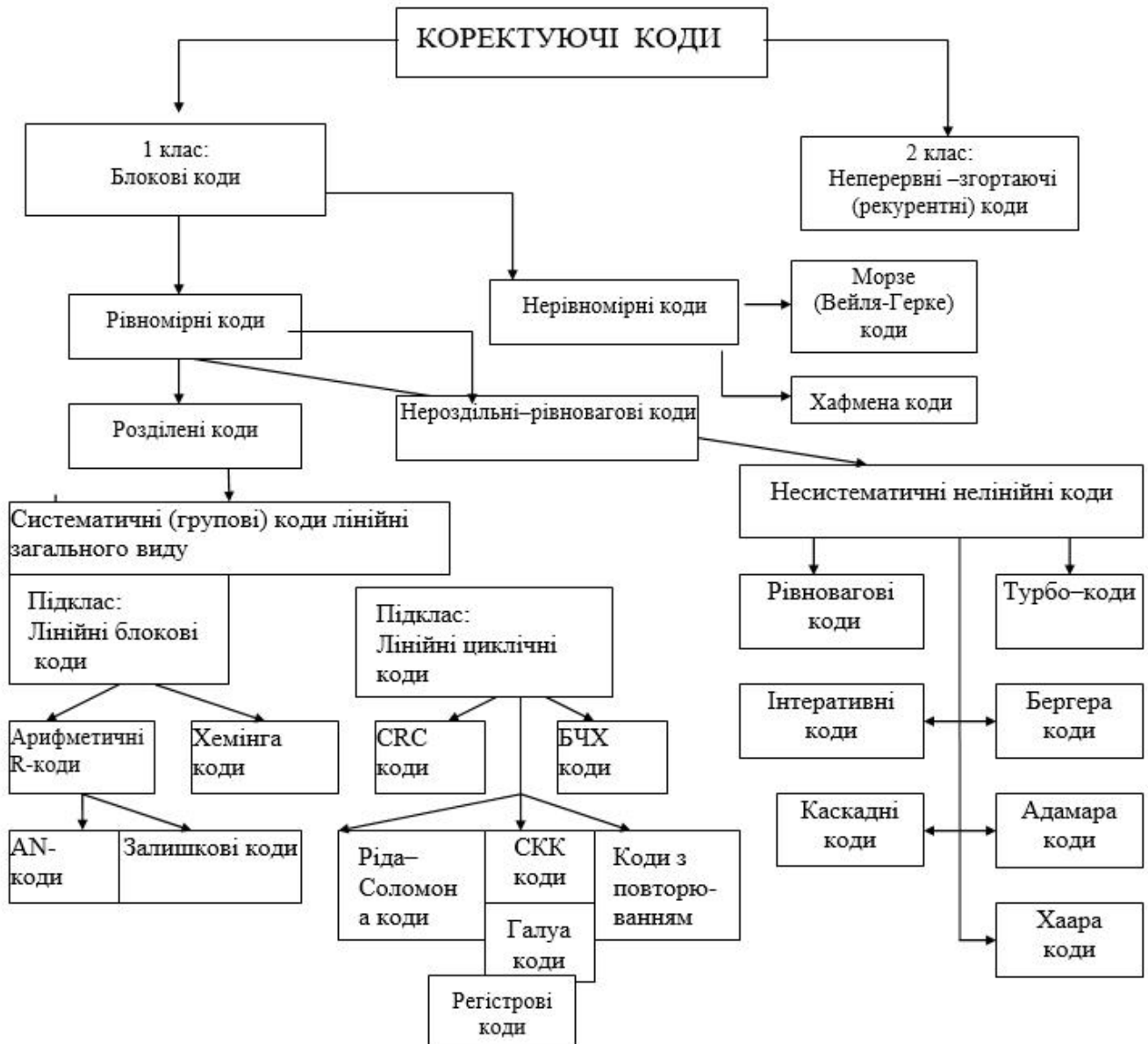


Рис.9.12. Схема класифікації перешкодостійких (коректуючих) кодів

Каскадне кодування. Турбо-коди.

Переваги різних способів кодування можна об'єднати, застосувавши ККТ. При цьому інформація спочатку кодується одним кодом, а потім другим, в результаті отримуємо код—добуток. Наприклад, популярна є наступна конструкція: дані кодуються кодом РСК, потім рандомізуються (при тому символи, розташовані близько, поміщаються далеко один від другого) і кодуються ЗГК. На приймачі спочатку декодується код, потім здійснюється зворотнє перемножування (про цьому пачки помилок на виході згортаючого декодера попадають в різні кодові слова РСК, і потім здійснюється декодування РСК. Деякі коди-добутки спеціально сконструйовані для ітеративного декодування, при якому декодування здійснюється в декілька проходів, кожний із яких використовують інформацію від попереднього. Такі коди називаються турбо – кодами і дозволяють добитися великої ефективності, однак, їх декодування вимагає великих ресурсів. Кодування згортаючим кодом виконується з допомогою регістра зсуву, відводи від якого сумуються по модулю два. Таких сум може бути дві (частіше всього) або більше [8,10].

Декодування даних.

Задача декодування полягає в отриманні k – елементної комбінації із прийнятого n – розрядного кодового слова при одночасному виявленні або виправленні помилок. Метод багатократного повторення досліджувався на прикладі коду МТК-2 з мажоритарним декодуванням [1]. Спосіб багатократного повторення аналогічний способу передачі одної і тої самої ж інформації по декількох паралельних каналах зв'язку. В цьому випадку необхідно мати не менше трьох каналів зв'язку (наприклад, з частотним рознесенням), несучі частоти яких потрібно вибирати таким чином, щоб помилки в каналах були незалежними. Перевагою таких систем є надійність і малий час затримки в отриманні інформації. Основним недоліком багатоканальних систем так, як і систем з повторенням, є нераціональне використання надлишковості. Декодування ЗГК, як правило, виконується по алгоритму Вітербі, який намагається відновити передану послідовність згідно критерію максимальної правдоподібності. [11,12]

9.2. Теоретичні основи захисту даних від помилок

В процесі зберігання даних і передачі інформації по мережах зв'язку неминуче виникають помилки. Контроль цілості даних і виправлення помилок – важливі задачі на багатьох рівнях роботи з інформацією (на фізичному, каналному, та транспортному рівнях моделі OSI). В системах зв'язку можливі декілька стратегій боротьби з помилками:

- виявлені помилки в блоках даних і автоматичний запит повторної передачі пошкоджених блоків – цей підхід застосовується в основному на каналному та транспортному рівнях;
 - виявлення помилки в блоках даних і відкидання пошкоджених блоків – такий підхід деколи застосовується в системах потокового мультимедіа, де важлива затримка передачі і немає часу на повторну передачу;
 - виправлення помилок (forward error correction) застосовується на фізичному рівні.
- Основних методів захисту даних від помилок є 3:

1. Коди парності;
2. Повторне передавання даних. Передавання зі зворотнім контролем;
3. Захист даних надлишковим кодуванням.

Повторне передавання даних;

$D1 \xrightarrow{K.З.} D1?$

- при одноразовому передаванні даних неможливо виявити помилки і провести їх корекцію без застосування сигнальних коректуючи кодів Галуа.
- Нехай дані передаються два рази. $D_1 \xrightarrow{K.З.} D_1 D_2$ $D_1=D_2$ $K_n=0,5$. Такий метод дозволяє виявити помилки але не дозволяє виправити їх.

- При передачі даних тричі: $D_1 \xrightarrow{\text{к.з.}} D_1 D_2 D_3$. Можна перевірити $D_1=D_2=D_3$, або $D_1=D_2$, $D_1 \neq D_3$, $D_2 \neq D_3$. $K_n=0,6(6)$. Дозволяє виявити однократні помилки і виправити їх. Якщо помилки виникають лиш в одному коді їх можна виправити.

- Можна застосувати чотирьохкратну передачу даних, дозволяє виявити помилки і виправити, якщо вони є лиш в одному коді.

Симетричні інверсні коди.

$D_1(1101) \rightarrow D_1 D_2(11010010)$ $K_n=0,5$. Застосування таких кодів дозволяє виявляти і виправляти помилки ($X \oplus Y$).

При передаванні інформації помилки можуть виникнути в одному і в багатьох розрядах. За рахунок впливу імпульсних завад в каналах зв'язку виникають пачки помилок. Було проведено дослідження каналів зв'язку при $C=1200$ біт/с.

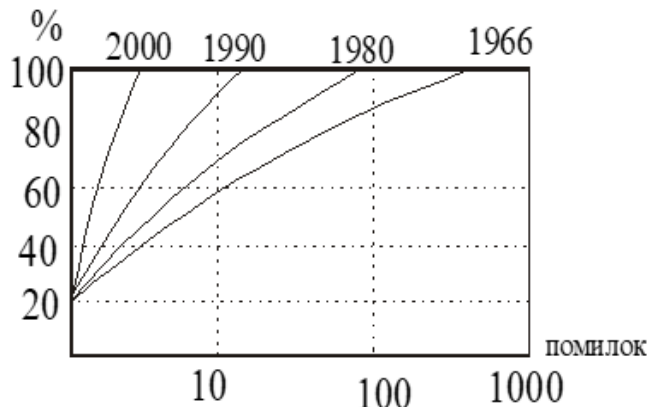


Рис. 9.13. Графік залежності повторної передачі даних

В результаті отримали залежність: метод повторної передачі даних є неефективним в зв'язку з великою надлишковістю даних. Недоліком є також відсутність інформації про надійність каналу зв'язку.

Передавання зі зворотнім контролем.

$D_1 \rightarrow D_2$
 $D_3 \leftarrow$ адаптивний і дозволяє передавати на рівні передавача характер помилок в каналах зв'язку.

$D_1=D_3$ $K_n=0,5$. Даний метод є

Характер помилок : [12]

$D_1=11010110$

$D_3=10110111$

хх х де х означає інвертування бітів даних.

$D_3=1110100110$

L L де L вставка бітів

$D_3=101010$

P p - випадання бітів даних.

$D_3=11....10$

zzzz - завмирання каналу зв'язку.

Даний метод в зв'язку з надлишковістю не набув значного поширення.

9.3. Порівняльний аналіз методів формування кодових ключів Галуа

Перевагами використання кодів унітарного базису, кодів базису Радемахера є порівняно нескладна реалізація декодування прийнятої інформації, а основними недоліками таких кодових систем є: надлишковість кодів, необхідність застосування методу пакетного передавання даних. Наведені умови ставлять підвищені вимоги до якості каналів зв'язку,

знижують швидкість передавання даних, ускладнюють апаратну та програмну реалізацію таких систем. Приклади кодування витрати та відповідна кількість біт для отримання її інтегрального значення для вище вказаних методів представлення даних первинних інформаційних джерел наведено в табл.9.5

З метою виявлення ефективності використання описаних методів кодування даних імпульсних перетворювачів проведено порівняльний аналіз розмірів інформаційних пакетів для базисів унітарних функцій $L_{(Uni)}$, базису Радемахера $L_{(Rad)}$ та базису Галуа $L_{(Gal)}$ згідно наступних аналітичних виразів :

$$\begin{aligned} L_{(Rad)} &= \text{Int}(\log_2(10^d + 1)) + B_s; \\ L_{(Uni)} &= n; \\ L_{(Gal)} &= \text{Int}(\log_2(n + 1)), \end{aligned} \quad (9.2)$$

де n –максимальне число відліків лічильника імпульсного перетворювача, B_s –число старт-стопових біт, d – число розрядів лічильника імпульсного перетворювача. Результати проведеного аналізу подано на рис.9.14.

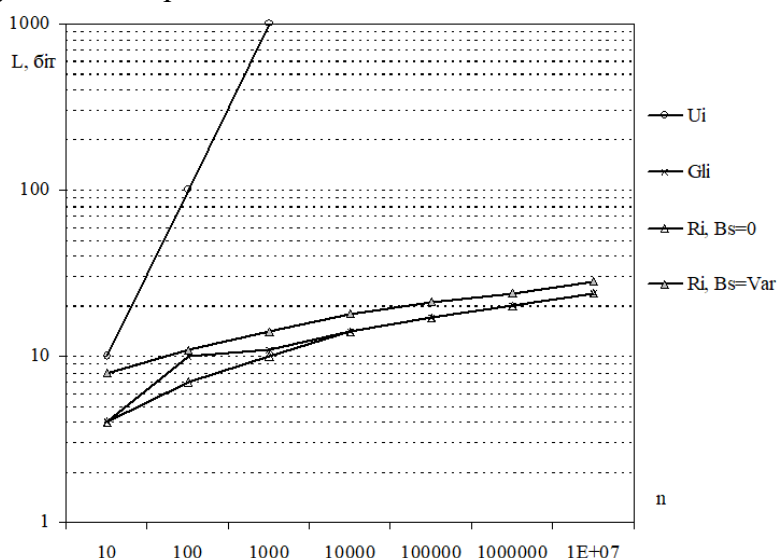


Рис.9.14 .Зміна розміру пакетів даних від числа відліків лічильника

Таблиця 9.5

Кодування даних в унітарному базисі(U_i), базисі Радемахера(R_i), базисі Галуа(G_i)

i							..	3	4	5	6	7	8	9
i	0	0	0	0	...	1	1							
	000000	000001	000010	000011	...	100010	100011							
i	..													
l _i	..													

Таким чином метод кодування даних на основі циклічних Галуа-послідовностей забезпечує мінімальний об'єм інформаційних пакетів (властивість унітарних кодів), а властивість рекурентності ефективно може бути використана для компактного кодування унікальних номерів баз даних. Отже, для формування унікальних номерів в реляційних базах даних найдоцільніше використовувати кодові послідовності, які формуються за допомогою М-послідовностей.

9.4. Методи формування кодових ключів Галуа на основі унітарного базису

В унітарному базисі зміна контрольованої величини представляється шляхом формування одиничного біту U_i , а інтегральне значення визначається як сума всіх попередньо прийнятих біт.

$$Q = \sum_{i=1}^n U_i, \quad (9.3)$$

де n – кількість попередньо прийнятих імпульсів [18, с. 42].

Використовуючи унітарний базис для формування унікального номера бачимо, що на формування унікального номеру для одного запису БД затрачається об'єм пам'яті, який прямо пропорційний кількості записів в таблиці, кількість операцій, яка необхідна для його формування дорівнює 1.

Основним недоліком такого методу кодування даних є відсутність захисту від імпульсних завад, що приводить до вставок або пропадань інформаційних імпульсів.

Розглянемо приклад реляційної бази даних, в якій в якості унікального номеру виступає унітарний базис (табл.9.6).

Таблиця 9.6

Формування унікального номеру в унітарному базисі			
№ п/п	Унікальний номер, U	Номер телефону	Унітарний код
1	1000000000	581074	1111100000 1111111100 1000000000 1111111111 1111111000 1111000000
...	...	...	...

А також унітарний базис є інформаційно неефективний, коли кількість процесорів є меншою N .

9.5. Методи формування кодових ключів на основі M – послідовності

Для того, щоб уникнути надлишковості кодів використовують кодову послідовність Галуа, яка базується на основі базису Галуа (рис. 9.15) [20].

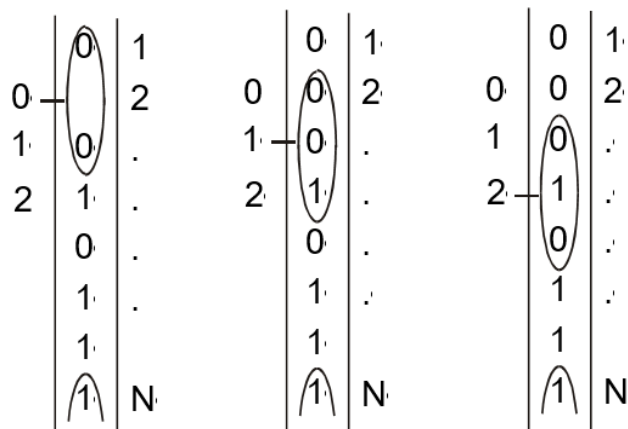


Рис. 9.15 Представлення коду Галуа.

Дані коди є одними з найбільш досконалою упаковкою інформації (рис.9.16).

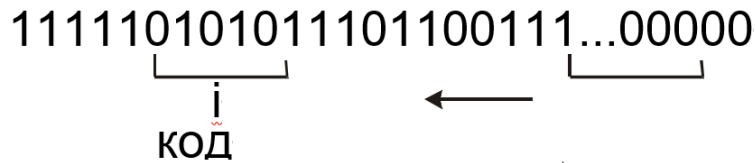


Рис. 9.16 Формування коду Галуа.

Для генерування кодів Галуа потрібно застосувати ключ:

$$G_{i+1} = G_i \oplus G_{i-(n-1)}, \quad (9.4)$$

де n – довжина кодона [20, с.66].

Наступний біт в базисі Галуа формується таким чином:

- вибираємо розмір ключа;
- перший і останній біти ключа додаємо по mod 2;
- зсуваємось на одиницю і додаємо наступний біт ключа з новоутвореним бітом;
- формування послідовності Галуа відбувається доти, поки вона не буде дорівнювати: $2^n - 1$, де n - розмір ключа.

Розглянемо приклад формування послідовності Галуа при використанні ключа, довжина якого дорівнює 4 біти. Послідовність Галуа матиме вигляд (рис 9.13).



Рис.9.17. Формування коду Галуа при $n=4$

В базисі Галуа зміна контрольованої величини передається за допомогою одного біту, як і в унітарному базисі, а інтегральне значення визначається декодуванням n попередньо прийнятих біт:

$$Q = f(Gl[n]), \quad (9.5)$$

де n – довжина породжуючого ключа М-послідовності [20, с.66].

Перевагами базису Галуа є максимальне стиснення інформації, висока захищеність від помилок, а також уникнення надлишковості інформації.

Використовуючи базис Галуа для формування унікального номера бачимо, що на формування унікального номеру для одного запису БД затрачається 1 біт пам'яті, кількість операцій, яка необхідна для його формування дорівнює 1 [20, с.66].

В автоматизованих комп'ютерних системах до теперішнього часу застосовуються І-залишкові коди, які дозволяють лише виявити помилку (так званий контроль по модулю). Процедура виявлення виконується на кожній арифметичній і логічній операції, причому її структурні і схемотехнічні рішення відпрацьовані до такої степені, що тимчасові витрати на контроль практично не впливають на загальну продуктивність. Виявлення з'явлених помилок потребує негайної реакції, яка зводиться до організації, а точніше, до збудження „зворотного каналу”. Останній може приймати різноманітні форми, однак в будь-якому випадку, як і в передачах даних, необхідними залишаються витрати пам'яті і часу. Більш того, так же як і в системах передачі даних, такий канал в комп'ютерах далеко не завжди виявляється доступним. Для бортових машин з жорсткими енергетичними і масово-габаритними обмеженнями. Альтернатива полягає у виправленні виявлених помилок методом перешкодостійкого кодування за новими корегуючими кодами в базисі Галуа. Комутація здійснюється не на рівні каналів, а на рівні пакетів даних [27].

Виконані порівняння характеристик кодування даних в базисі Радемахера і принципами кодування в базисі Галуа, які дозволили встановити суттєве зменшення об'ємів даних а також

можливості виявлення та виправлення помилок при кодуванні інформаційних індикаторів в БД.

Досліджені методи формування кодових ключів та послідовностей Галуа у різних ТЧБ і як ефективність застосування рекурентних М- послідовностей на основі загорткових рівнянь.

Досліджені методи захисту інформації в БД від помилок на основі широкого класу коректуючих кодів , показано що, використання рекурентних властивостей кодів поля Галуа забезпечує найбільш прості алгоритми кодування даних , також виявлення і виправлення помилок.

Дослідження коректуючих властивостей кодів базису Галуа, а також арифметичні операції над кодами названого класу , які характеризуються простими алгоритмами і високою швидкістю виконання операції.

РОЗДІЛ 10

СИГНАЛЬНІ КОРЕКТУЮЧІ КОДИ В ТЧБ ГАЛУА

Викладені теоретичні основи та принципи формування сигнальних коректуючих кодів в ТЧБ Галуа. Поля надходження масивів даних і виникнення помилок може відбутися у процесі у дистанційному доступу до БД. При цьому існують методи на основі CRC- кодів які забезпечують виявлені помилки і повторні передачі пакетів даних, що суттєво знижує ефективність функціонування ГРБД глобальних розподілених БД. Ефективним рівнем вирішення цієї проблеми є застосування нового класу сигнальних коректуючих кодів в ТЧБ Галуа.

10.1. Методологія формування та оцінки ефективності формування сигнальних кодів.

Методи цифрової обробки сигналів базуються на зміні системних характеристик носіїв фізичного сигналу. Якщо ці зміни є неперервні, то цей метод називається модуляцією, якщо стрибкоподібно - маніпуляцією. В загальному випадку сигнал можна представити у вигляді функціоналу, формалізованого запису аналітичного виразу в залежності від формалізованих параметрів [1,2]:

$$S_x = F(A, \psi, \omega, t, \varphi), \quad (10.1)$$

де A - амплітуда сигналу; ψ - базисна функція; ω - частота в базисі; φ - фаза сигналу; t - час.

Вказані 5 параметрів є основою для організації методів модуляції та маніпуляції в різних теоретико-числових базисах.

10.1.1. Імпульсні методи маніпуляції.

В сучасних інформаційних системах для передавання даних найширшого застосування отримали методи імпульсної, а в окремих випадках потенціальної маніпуляції [1]. [65,92,98].

Імпульсні методи маніпуляції сигналів найчастіше використовують на низових рівнях комп'ютерних мереж, в цифровій телефонії, а також комп'ютерних системах з оптичними каналами ІК. Оскільки дані методи використовують обмежену частину енергії на інтервалі тривалості сигналу, а також потребують широкої смуги частот в каналі зв'язку, ефективність їх недостатньо висока [2].

В загальному випадку надлишковість даних, які передаються згідно стандартного протоколу, можна оцінити у вигляді коефіцієнта надлишковості як відношення об'єму даних, що передаються, до об'єму вхідних даних, які підлягають передачі, згідно аналітичного виразу:

$$K_n = V_k / V_x, \quad (10.2)$$

де V_k - об'єм даних, що передаються, V_x - об'єм вхідних даних.

$$V_k = V_c + V_x, \quad (10.3)$$

$$V_c = V_\Phi + V_A + V_Y + V_{kk} + V_B \quad (10.4)$$

V_c - об'єм службових даних;

V_Φ , V_A , V_Y , V_{kk} , V_B - відповідні об'єми: кодів флага, адреса станцій, типу фрейму, коректуючого коду, символів біт-стаффінга.

З метою оцінки надлишковості існуючих протоколів типу HDLC, HP-IL, KT-1 проаналізовані структури фреймів, які формалізовані на основі аналітичних виразів:

$$K_n(\text{HDLC}) = (V_\Phi + V_A + V_Y + V_{kk} + 2^b + 2^k \cdot 8) / 2^k \cdot 8; \quad (10.5)$$

$$K_n(\text{HP-IL}) = (3 + 2^k \cdot 8 + V_{kk}) / K_{\text{FC}} \cdot 2^k \cdot 8; \quad (10.6)$$

$$K_n(\text{KT-1}) = (4 + 2^k \cdot 8 + V_{kk}) / K_{\text{FC}} \cdot 2^k \cdot 8. \quad (10.7)$$

$$K_{\text{FC}}(\text{HDLC}) = 0.5; K_{\text{FC}}(\text{HP-IL}) = 0.3; K_{\text{FC}}(\text{KT-1}) = 1.$$

де K_n - коефіцієнта надлишковості, K_{FC} - коефіцієнт форми сигналу.

В результаті отримано графік надлишковості передавання даних існуючими методами (рис.10.1).

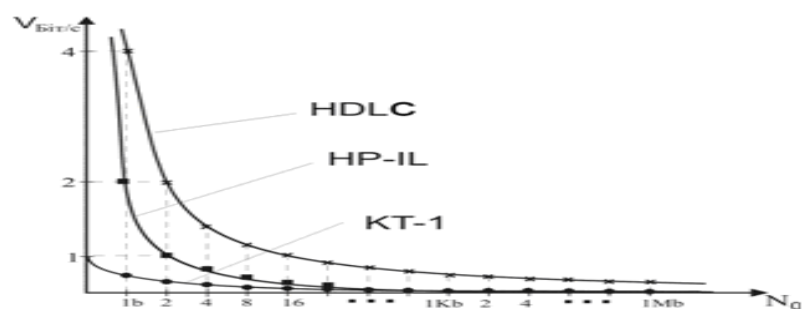


Рис. 10.1 Надлишковість передавання даних протоколами HDLC, HP-IL, KT-1.

Серед цих методів застосовують імпульсну та диференційно-імпульсно кодові модуляції. Табл.10.1.

Таблиця 10.1

Методи двовимірної імпульсної маніпуляції.

ММ	код	0	1	1	0	1	0	0	1	М
АЧМ	RZ									2
АЧМ	RB									2
АЧМ	NZ									3
АЧМ	RZB									3
АФМ	RZ-F									2
ЧІМ	RB-FM									2
ЧІМ	RB-M ² FM									2

Порівняльний аналіз методів ефективності (переваг та недоліків) різних методів двовимірної імпульсної маніпуляції сигналів поданий у табл.10.2.

Таблиця 10.2

Класифікація переваг та недоліків різних методів двовимірної маніпуляції.

Метод маніпуляції	Переваги	Недоліки
АЧМ (RZ)	простий закон маніпуляції, сигнали однополярні	низька енергія сигналу, відсутня самосинхронізація.
АЧМ (RB)	вдвічі більша енергія сигналу, по відношенню до методу RZ ;	сигнал знакозмінний; загальний недолік для RZ і RB дрейф нуля в лінії зв'язку, що призводить до зниження завадозахищеності віддалених приймачів сигналу.
АЧМ (NZ)	більш високий рівень серії бітової синхронізації	сигнал знакозмінний
АЧМ (RZB)	синхронізація кожного біту, що передається	загальним недоліком методів RZB, RZ-F, RB-FM, RB-M ² FM є використання малої кількості енергії сигнального простору
АЧМ(RZ-F)	самосинхронізація кожного біту даних	
АФМ(RB-FM)	самосинхронізація і завадозахищеність	
ЧІМ(RB-M ² FM)	захист від пропадання імпульсів в лінії зв'язку.	

Загальним недоліком всіх представлених двовимірних методів маніпуляції є широкий спектр сигналів, а також неефективне використання сигнального вікна, що приводить до зниження швидкості передавання інформації і зменшення дальності передавання повідомлень.

10.1.2. Потенціальні методи маніпуляції

Широкого застосування в сучасних комп'ютерних системах набули також потенціальні методи маніпуляції сигналу [3]. Дані методи характеризуються підвищеною енергією сигналів, що передаються. Вони бувають двох типів: без самосинхронізації (потребують додаткового каналу синхронізації) і з самосинхронізацією (табл. 10.4).

Для виявлення помилок при передаванні сигналу використовують стандартні методи на основі рекурентних надлишкових коректуючих кодів типу HDLC, HP-IL, KT-1 [3].

Таблиця 10.3

Потенціальні методи маніпуляції.

КОД	0	1	1	0	1	0	0	1	m
NRZ-1									2
NRZ-M		↑	↓		↑			↓	2
MFM		↑	↓		↑		↓	↑	2
PE	↓	↑	↑	↓	↑	↓	↓	↑	2
FT	0	1	1	0	1	0	0	1	2
FM		↑	↓		↑		↓	↑	2
HP-IL		↑	↓		↑	↓	↑	↓	3
KT-1		↑	↓		↑	↓	↑	↓	3

Серед перелічених методів, метод KT-1 є найбільш ефективним, оскільки володіє наступними позитивними якостями:

1. Виключає повторення однакових ознак сигналів і забезпечує якісну бітову синхронізацію.
2. Двократні повторення сигналів використовуються в якості start-stop сигналів, що виключає необхідність формування флагів на границях інформаційних масивів і біт-стафінгів в середині інформаційного масива.

Метод KT -1 розроблений професором Николайчуком Я.М. [3] і успішно застосований в низових рівнях комп'ютерних систем в нафтогазовій промисловості (табл.10.4).

Таблиця 10.4

Порівняльні характеристики різних методів потенційних методів маніпуляції.

Метод маніпуляції	Переваги	Недоліки
NRZ – 1	забезпечує максимальну енергію сигналів, але відсутня бітова синхронізація	дрейф нуля в лінії зв'язку
NRZ – M	забезпечує синхронізацію сигналів "1"	
MFM	забезпечує фронтом наростанням або наростання нулів і передачу потенціалу	синхронізація бітів фронтами наростання та спаду сигналу з фазовою маніпуляцією
PE	нуль передається фронтом спаду, а одиничка фронтом наростання	ефективна бітова синхронізація
FT	висока заводо захищеність і самосинхронізація	зміна швидкості передавання даних, дрейф нуля

НР – ІЛ	характеризується самосинхронізацією, відсутністю дрейфу нуля	низька швидкість передачі даних при заданій смузі частот каналу зв'язку
КТ – 1	максимальна швидкість передачі сигналів і самосинхронізація	дрейф нуля. Для ліквідації дрейфу сигнали "0", "1" і сигнал "повторення" передаються трьома частотами.

10.1.3. Квазітрійкові методи маніпуляції сигналів

У табл.10.5. показані приклади маніпуляції сигналів модифікованими методами квазітрійкової маніпуляції, які характеризуються відсутністю дрейфу нуля.

Таблиця 10.5

Формування імпульсної послідовності методом КТ-М									
код	0	1	1	0	1	0	0	1	m
FM-КТ1									3
КТ-М									3

Загальний недолік проаналізованих методів маніпуляції є широкий спектр сигналів, як наслідок цього, відбувається зниження швидкості передавання інформації і високий рівень завад між каналом зв'язку. Тому доцільно використовувати методи квазітрійкової маніпуляції сигналів покращеної форми з вузьким спектром (табл.10.6).

Таблиця 10.6

Квазітрійкові методи маніпуляції сигналів покращеної форми.									
ФОРМА сигналу	Метод	Бітова послідовність							
		0	1	1	0	1	0	0	1
Трапецеїдальні	КТ-Т								
Дзвоноподібні	КТ-Д								
$\frac{\sin x}{x}$	КТ-С								
$\sin^2 x$	КТ-Х								

Ефективність методу маніпуляції сигналу також залежить від реалізації блокової синхронізації.

При цьому аналіз ефективності методу маніпуляції сигналів з врахуванням блокової синхронізації можна вирахувати за допомогою наступного виразу:

$$K_e = (n+m)/n \quad (10.8)$$

де n – число біт-інформаційної кодової послідовності

m – число біт коду блокової синхронізації.

На рис.10.2.наведено приклад виконання блокової синхронізації при використанні стандартного протоколу обміну даними в комп'ютерних системах HDLC, методу маніпуляції НР-ІЛ та КТ-1і без врахування коефіцієнта форми сигналу та ступеня використання енергії сигнального вікна.

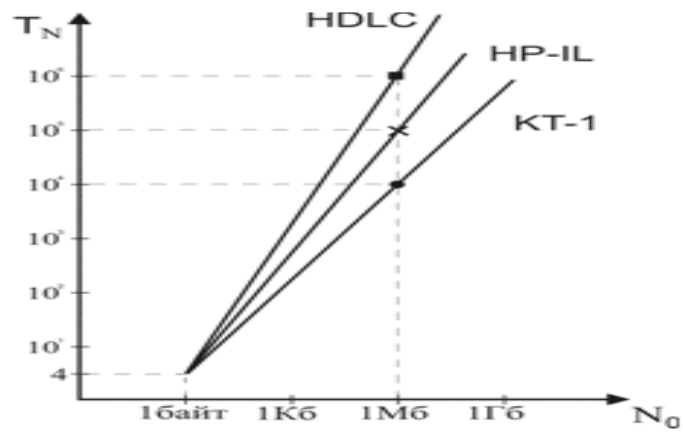


Рис.10.2. Характеристика швидкості передавання даних в стандартних протоколах:
 N_0 – об'єм даних; T_N – швидкість передавання в залежності від об'єму даних.

Серед методів маніпуляції найбільш широко використовується Манчестерський код, рис.10.3.

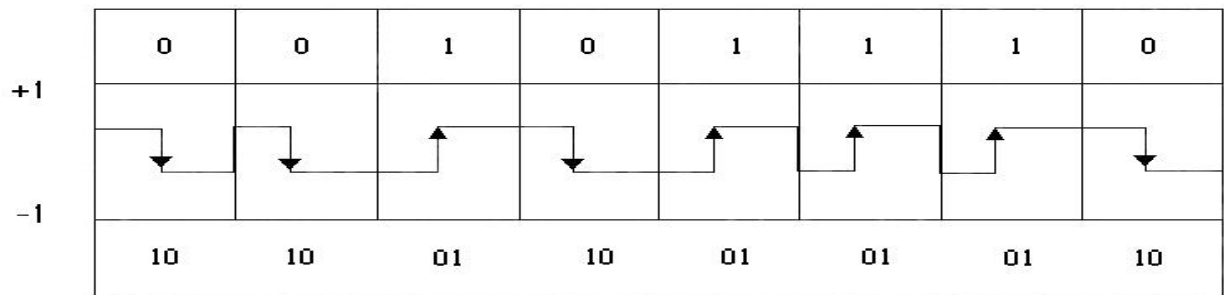


Рис.10.3. Кодування сигналу на основі Манчестерського коду.

Аналіз рис.10.3. показує, що незважаючи на те, що при передаванні байту даних (8 біт), в каналі є присутні ще 2 байти двійкових сигналів - цей флаг пояснюється тим, що для передавання Манчестерським кодом "1" і "0" використовуються не всі позиції сигнального простору. рис.10.4.

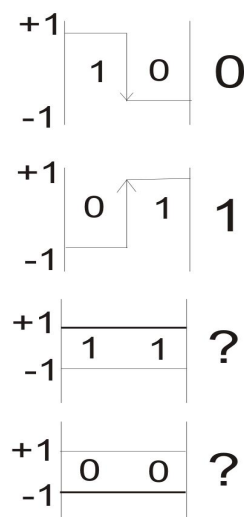


Рис.10.4. Представлення кодової послідовності з використанням бітів даних.

З рис.10.4. видно, що дві позиції "00" і "11" в каналах зв'язку фактично не використовуються.

Існують інші методи маніпуляції, де ці коди використовують для повторення нулів і одиниць, що підвищує ефективність передачі інформації (рис.10.5).

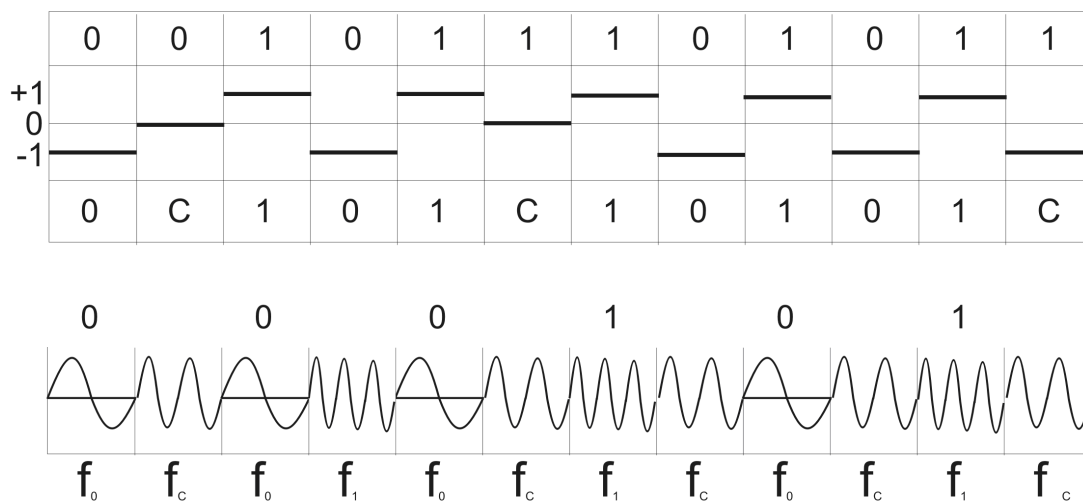


Рис.10.5. Метод кодування КТ-1.

С – синхронізація, f_0 – частота передавання нулів, f_1 – одиниць, f_c – синхронізації.

10.1.4. Частотні, фазові та амплітудно-фазові сигнали, двовимірні та багатовимірні сигнальні простори.

Для порівняння ефективності різних методів маніпуляції необхідно виконати їх розрахунок на основі сигнальних моделей.

Представлення методів модуляції відбувається в одиничному колі рис.10.6.

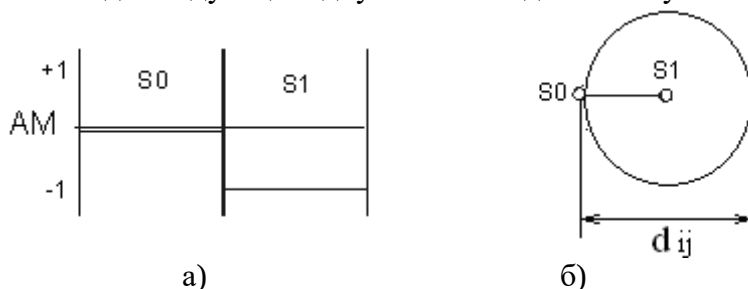


Рис.10.6. Сигнальне коло при амплітудній (а) та фазній модуляціях (б): S_i, S_j - сигнали в сигнальному просторі, d_{ij} - кодова віддаль, $E_i = R_i$, E - енергія сигналу.

Чим більше d_{ij} тим ефективність методу модуляції більша табл.10.7

Таблиця 10.7

Систематизація сигнальних кіл при різних ознаках сигнального вікна.

M=2			
M=3			

M=4			
M=5			
M=6			
M=7			
M=8			
M=9			

де M – число ознак сигнального вікна (число маніпуляцій).

10.2. Сигнальні коди на основі теорії кодів Галуа.

Коди поля Галуа [8-10] за загальною класифікацією відносяться до підкласу циклічних блокових кодів, які володіють всіма основними властивостями завадозахищених кодів. В блокових кодах послідовність елементарних повідомлень розбиваються на блоки символів ($B_1, B_2, B_3, \dots, B_n$) фіксованої довжини K , кожному з яких ставиться в відповідності певна комбінація символів кодового слова ($b_1, b_2, b_3, \dots, b_n$).

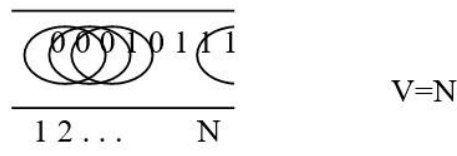


Рис. 10.7 – Представлення коду Галуа (V – об'єм виродженої матриці).

Циклічні коди відносяться до класу систематичних кодів. Для даних кодів можна записати відповідний їм аналітичний вираз, чи деяке логічне співвідношення, яке визначається правилами створення цих кодів. Найбільш зручною формою представлення циклічних кодів – використання алгебраїчного виразу [84]

$$G(x) = a_{n-1} \times x^{n-1} + a_{n-2} \times x^{n-2} + \dots + a_1 \times x + a_0, \quad (10.9)$$

де $a_0 - a_{p-1}$ – числа, що дорівнюють «0» чи «1», які визначають відповідні значення розрядів кодових комбінацій.

Таким чином дія над циклічними кодами зводиться до дії над відповідними математичними виразами. Коефіцієнти однакових степенів додаються за модулем 2. Дані коди є одними з найбільш досконалою упаковкою інформації.рис.10.8.

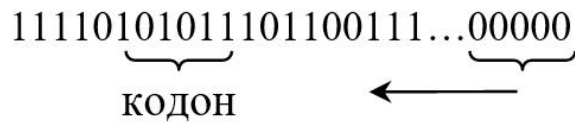


Рис.10.8. Формування коду Галуа.

Найефективніше переваги даного базису можна використати при кодуванні інтегральних значень, оскільки при інтегруванні кожне наступне значення збільшується на одиницю. Тому, на відміну від базису Радемахера, кожне дискретне значення інтеграла функції $x(t)$ замість p -розрядного двійкового коду фіксується одним бітом Галуа.

Для генерування кодів Галуа використовуються неприводимі алгебраїчні багаточлени (табл.10.8).

Таблиця.10.8

Таблиця ключів кодів Галуа

Розрядність коду	Ключ коду Галуа
4	$x_1 \oplus x_4$
5	$x_2 \oplus x_5$
6	$x_1 \oplus x_6$
7	$x_3 \oplus x_7$
8	$x_3 \oplus x_8$
9	$x_4 \oplus x_9$
10	$x_3 \oplus x_{10}$
11	$x_2 \oplus x_{11}$
12	$x_1 \oplus x_4 \oplus x_6 \oplus x_{12}$
13	$x_1 \oplus x_3 \oplus x_4 \oplus x_{13}$
14	$x_1 \oplus x_6 \oplus x_{10} \oplus x_{14}$
15	$x_1 \oplus x_{15}$
16	$x_1 \oplus x_3 \oplus x_{12} \oplus x_{16}$
17	$x_3 \oplus x_{17}$
18	$x_7 \oplus x_{18}$
19	$x_1 \oplus x_2 \oplus x_5 \oplus x_{19}$
20	$x_1 \oplus x_{18}$
21	$x_2 \oplus x_{21}$
22	$x_1 \oplus x_{22}$
23	$x_5 \oplus x_{23}$
24	$x_1 \oplus x_2 \oplus x_7 \oplus x_{24}$
25	$x_1 \oplus x_4$
26	$x_1 \oplus x_2 \oplus x_3 \oplus x_7$
27	$x_1 \oplus x_2 \oplus x_3 \oplus x_6$

28	$x_1 \oplus x_4$
29	$x_1 \oplus x_3$
30	$x_1 \oplus x_2 \oplus x_5 \oplus x_7$
31	$x_1 \oplus x_7$
32	$x_1 \oplus x_3 \oplus x_7 \oplus x_8$

Важливою перевагою кодової послідовності Галуа (КПГ) є їх проста генерація на основі рекурентного рівняння. Найпростіші ключі кодів Галуа описуються виразом :

$$G_{i+1} = G_i \oplus G_{i-(n-1)}, \quad (10.10)$$

де n – довжина кодона, $\oplus$ - операція додавання “по модулю два”

При $n=4$ отримаємо код:

1111.0101.1001.0000*

Рис.10.10. Формування коду Галуа при $n=4$ (* - біт-стаффінг).

Коди поля Галуа належать до класу рекурентних кодів, які широко використовуються для захисту інформації від несанкціонованого доступу (рис.10.11).

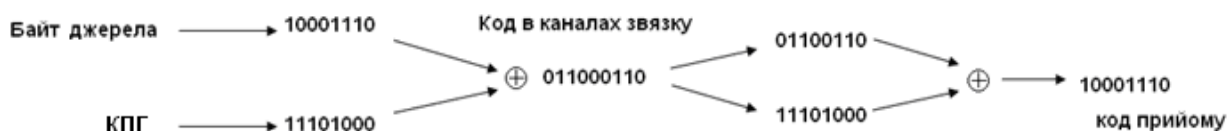


Рис.10.11 Байт, який передається на основі кодової послідовності Галуа.

Таким чином дія над циклічними кодами зводиться до дії над відповідними математичними виразами. Дані коди є одними з найбільш досконалою упаковкою інформації.

Найбільш ефективно переваги даного базису можна використати при кодуванні інтегральних значень, оскільки при інтегруванні кожне наступне значення збільшується на одиницю. Тому на відміну від базису Радемахера кожне дискретне значення інтеграла функції замість n –розрядного двійкового коду фіксується одним бітом Галуа (рис. 10.8).

У загальному випадку код поля Галуа генерується згідно рівнянь незвідних поліномів (табл. 9.9) на основі узагальненого рекурентного виразу:

$$X_{i+1} = (x_i \cdot a_i \oplus x_{i-1} \cdot a_{i-1} \oplus \dots \oplus x_{i-n} \cdot a_{i-n}), \quad (10.11)$$

де $a_i \in 0,1$ - двійкові значення незвідного алгебраїчного поліному, що формує код рекурентного ключа для послідовності.

В окремих випадках код Галуа G_2^n формується згідно рекурентної послідовності:

$$G_{i+1} = G_i \oplus G_{i-n}, \quad (10.12)$$

де n – розрядність кодона

Таблиця 10.9

Незвідні поліноми $\pi(x) = x^r + f(x)$ степенів r і характеристик p .

p	r	$\pi(x); [x^r = f(x)]$
2	2	$x^2 + x + 1$
	3	$x^3 + x + 1; x^3 + x^2 + 1$
	4	$x^4 + x + 1$

	5	$x^5 + x^2 + 1$
	6	$x^6 + x + 1$
	7	$x^7 + x + 1; x^7 + x^3 + 1$
	8	$x^8 + x^4 + x^3 + x^2 + 1$
	9	$x^9 + x^4 + 1$
	10	$x^{10} + x^3 + 1$
	11	$x^{11} + x^2 + 1$
	12	$x^{12} + x^6 + x^4 + x + 1$
	13	$x^{13} + x^4 + x^3 + x + 1$
	14	$x^{14} + x^{10} + x^6 + x + 1$
	15	$x^{15} + x + 1$
	16	$x^{16} + x^{12} + x^3 + x + 1$
	17	$x^{17} + x^3 + 1$
	18	$x^{18} + x^7 + 1$
	19	$x^{19} + x^5 + x^2 + x + 1$
	20	$x^{20} + x^3 + 1$
	21	$x^{21} + x^2 + 1$
	22	$x^{22} + x + 1$
	23	$x^{23} + x^5 + 1$
	24	$x^{24} + x^7 + x^2 + x + 1$
	25	$x^{25} + x^3 + 1$
	26	$x^{26} + x^6 + x^2 + x + 1$

При передаванні та прийманні інформації на основі сигнальних кодів використовуються маніпульовані сигнали сформовані на основі чотирьох ознак, які поставлені у відповідність до елементів інформаційного повідомлення відповідно до кодів поля Галуа.

Відомий спосіб передавання, та приймання інформації на основі модифікованої частотної модуляції (MFM). В модифікованій частотній модуляції використовуються 4 сигнальні ознаки: фронт наростання (), фронт спаду (), які відповідають символу «1», і потенціал «+», потенціал «-», які відповідають символу «0». При повторенні символу «0» для бітової синхронізації також використовують фронт наростання () чи спаду ().

Проте такий спосіб не дозволяє виявляти і виправляти помилки.

Інший спосіб передавання та приймання інформації [9] дозволяє виявити та виправити помилки при прийманні біт-орієнтованих даних за рахунок коректуючих властивостей кодів Галуа без додаткового передавання контрольних сум.

При передаванні та прийманні інформації використовуються маніпульовані сигнали сформовані на основі чотирьох ознак, які поставлені у відповідність до елементів інформаційного повідомлення відповідно до кодів поля Галуа. Такий принцип формування сигнального коду полягає в тому, що біти одиниць в пакеті даних нумеруються рекурентним кодом Галуа G^2_k . Причому для одиниць в пакеті даних біт Галуа «1» передається фронтом наростання (), а біт Галуа «0» передається фронтом спаду (). Біти нулів в пакеті даних також нумеруються рекурентним кодом Галуа G^2_k . Причому для нулів в пакеті даних біт Галуа «1» передається потенціалом «+», а біт Галуа «0» передається потенціалом «-». В якості чотирьох ознак маніпуляції на сигнальному рівні може бути відповідно використані набори з чотирьох фаз, частот, шумоподібних сигналів та їх комбінацій.

Приклад маніпульованого за допомогою сигнальних коректуючих кодів сигналу, при якому об'єм коду Галуа відповідає об'єму даних наведено на рис.10.12. З таблиці видно, що в

блоці об'ємом $N=2^4$ завершення послідовності нулів відповідає коду Галуа 1101 і завершується символами ++--+ ∨∨∨∨, тобто $N=6$, згідно G_2^4 . А завершення послідовності одиниць в коді Галуа відповідає символам ∧∨∨∧, тобто коду Галуа 0110, $N=10$.

Таким чином забезпечується ефективно симетричне кодування у вигляді кодів Галуа послідовності нулів і одиниць блоку даних з однозначним визначенням їх числа $N_0+N_1=N$, яке використовується для виявлення та виправлення помилок після передавання даних.

На рис.10.12 зображена реалізація потоку даних маніпульованих за допомогою сигнальних коректуючих кодів, з виявленням помилок на сигнальному рівні. В таблиці приведено приклад виникнення помилок на сигнальному рівні в 7-ій та 17-ій позиції нулів, а також 10-ій та 21-ій позиції одиниць.

Виявлення помилок ґрунтується на біт-орієнтованій нумерації послідовності нулів і одиниць, які передаються за допомогою кодових послідовностей Галуа.

Позиції бітів	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12	d13	d14	d15	d16
Біти даних	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Біти Галуа $G_4^2(1)$	1	1	1	1	0	1	0	1	1	0	0	1	0	0	0	0
Символьний код	∧	∧	∧	∧	∨	∧	∨	∧	∧	∨	∨	∧	∨	∨	∨	∨
Сигнальний код																
Біти даних	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Біти Галуа $G_4^2(0)$	1	1	1	1	0	1	0	1	1	0	0	1	0	0	0	0
Символьний код	+	+	+	+	—	+	—	+	+	—	—	+	—	—	—	—
Сигнальний код																
Біти даних	1	1	0	1	1	0	1	1	1	0	0	0	1	1	0	1
Біти Галуа $G_4^2(1)$	1	1		1	1		0	1	0				1	1		0
Біти Галуа $G_4^2(0)$			1			1				1	1	0			1	
Символьний код	∧	∧	+	∧	∧	+	∨	∧	∨	+	+	—	∧	∧	+	∨
Сигнальний код																

Рис. 10.12. Приклад формування сигнального коректуючого коду для інформаційного повідомлення

Якщо помилка виявлена, використовується формула, де рекурентним шляхом перевіряється, в якій саме позиції відбулася заміна символу нуля(одиниці), в процесі передавання даних і даний символ замінюється на правильний [9].

10.2.1. Спіральні властивості коректуючих кодів в ТЧБ Галуа

Теоретична основа підвищення завадозахищеності сигнальних коректуючих кодів показана в роботі, на основі виявленої наявності в послідовності Галуа черезрівневої рекурентної властивості.

Нехай маємо код Галуа, який представляється рекурентним кодом:

$$G_2^4 = 1111010110010001111010110... \quad (10.13)$$

Можна показати, що код генерований на основі виразу (10.13) можна упакувати в спіраль (рис.10.14), причому по кожній з чотирьох твірних формується рекурентна послідовність, яка має відповідні рекурентні властивості коду в базисі Галуа.

Отримана спіраль закодована рекурентним кодом розкручується зі збереженням рекурентності через 12 символів, що дозволяє виправляти помилки по твірних спіралі згідно виразу:

$$G_{i+1} = G_i \oplus G_{i-13} \quad (10.14)$$

З врахуванням спіральних властивостей сигнальний рекурентний код можна ефективно використати при виявленні пакетів помилок. Оскільки спочатку перевіряється і виправляється помилки, а потім - по твірних спіралі.

На рис.10.13 показано виявлення помилок при використанні спіральних властивостей сигнальних коректуючих кодів поля Галуа. Як видно з рис.9.14 при виникненні помилок в п'яти підряд позиціях виявлення помилок завдяки рекурсивному виразу (10.14) стає неможливим, оскільки можлива неправильна корекція, тому потрібно виявляти дані помилки завдяки спіральним властивостям, тобто виразу (10.15).

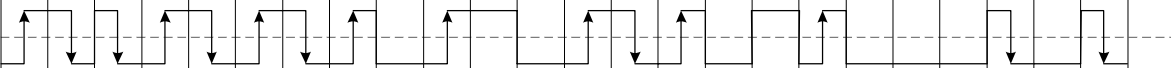
Номер позиції бітів	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	...
Біти даних	1	0	0	1	0	1	0	1	1	1	0	1	1	0	1	1	0	1	1	1	1	0	1	0	...
Біти Галуа $G_4^2(1)$	1			1		1		1	0	1		0	1		1	0		1	0	0	0		0		...
Біти Галуа $G_4^2(0)$		1	1		1		1				0			1			0					1		1	...
Сигнальний код																									
Помилка *							*			*							*				*				...
Символьний код з помилкою	Λ	∨	∨	Λ	∨	Λ	Λ ⁺ ₊	Λ	-	± _∨	+	-	Λ	∨	Λ	-	Λ _∨	-	-	-	Λ ₊ _∨	∨	Λ	∨	...

Рис. 10.13. Реалізація потоку даних за допомогою сигнальних коректуючих кодів, з виявленням помилок на сигнальному рівні

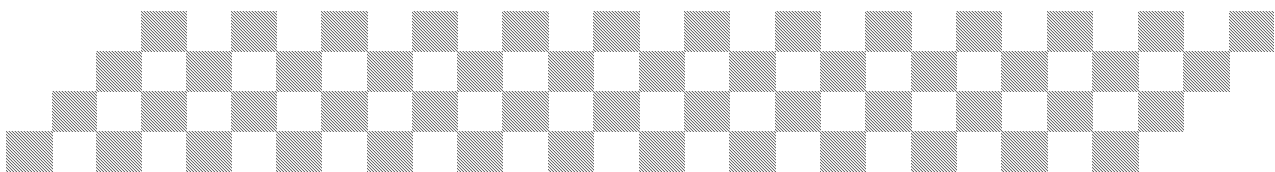


Рис. 10.14. Сигнальний коректуючий код, упакований у вигляді спіралі

10.2.2. Метод формування позиційно-сигнального коду (ПСК).

Суть методу безнадлишкового сигнального коду базується на принципі створення таких кодів з можливістю виявлення та виправлення помилок, які не призводять до збільшення числа сигналів при передаванні біт-орієнтованих потоків даних.

Суть методів безнадлишкового сигнального кодування з можливістю виявлення та виправлення помилок полягає в тому, що при формуванні такого класу кодів використовується до п'яти сигнальних ознак наступного типу:

- фронт наростання $\text{—}\uparrow\text{—}$ (Λ);
- фронт спаду $\text{—}\downarrow\text{—}$ (∨);
- додатній потенціал $\text{—}|$ (+);
- від'ємний потенціал $\text{—}|$ (-);
- нульовий потенціал $\text{—}|$ (S);

Зауважимо, що всі ці ознаки різними способами використовуються в стандартних потенціальних методах маніпуляції (табл.10.7) [3], при чому послідовність бітів даних додатково модулюється кодом поля Галуа.

Встановлено чотири можливих способи формування такого класу кодів:

- позиційно-сигнальний код (ПСК);
- несиметричний рекурентний сигнальний код (НРСК);
- рекурентний симетричний сигнальний код (РССК);
- квазі-символьний сигнальний код (КССК). [8].

Введемо наступну систему представлення сигнальних та кодових ПСК:

$$G_0^1 \Rightarrow «+» ; G_0^0 \Rightarrow «-» ; G_1^1 \Rightarrow «\wedge» ; G_1^0 \Rightarrow «\vee».$$

Біт «нуль», який на сигнальному рівні кодується в Галуа одиницею, представляється сигналом +1 і відповідним символом + в кодовому вигляді. Біт «нуль», який кодується в Галуа нулем, на сигнальному рівні, представлено потенціалом -1, а в кодовому вигляді відповідним символом -. Біт «одиниця», яка кодується в Галуа одиницею, представляється фронтом наростання на сигнальному рівні, або символом $\wedge$. Біт «одиниця», яка кодується в Галуа символом нуль, представляється фронтом спаду на сигнальному рівні, або символом $\vee$ в кодовому вигляді.

Формування ПСК коду показано в табл.10.10.

На рис.10.16 представлена символіка розрядно-позиційного сигнального коду Галуа, для півбайтових потоків даних, що відповідає потоку чисел або цифр.



Рис.10.16. Представлення сигнальних ПСК.

При формуванні ПСК даним методом для виключення появи в кодах даних комбінацій Start/Stop – сигналів, останній формується у вигляді трьохбітових послідовностей відповідно:
Start --- Stop +++

Тобто число start/stop –сигналів повинно розраховуватись згідно виразу:

$$S = n+1, \quad (10.15)$$

де n – розрядність кодону Галуа масиву даних, що передаються.

Розглянемо ряд етапів досліджень запропонованого методу ПСК, при зміні позиційно-сигнального кодування даних, що передаються в ІК.

Нехай формат даних, що передаються по ІК рівний 4-м бітам (n=4). Тоді код поля Галуа G_2^2 формується у вигляді послідовності 1100:

$$\begin{array}{ccccccc}
 & G & 1 & 1 & 0 & 0 & \\
 \text{start} & & 0 & 0 & 0 & 0 & \text{stop} \\
 --- & & + & + & - & - & +++
 \end{array} \quad (10.16)$$

Позитивною характеристикою ПСК є можливість виявлення та в окремих випадках виправлення помилок типу «стирання» та «вставок» бітів, які можуть виникати під впливом мультиплікативних завад.

Для 4-х бітового повідомлення, яке передається існує $m=2^4$ комбінацій ПСК, які в сигнальному та кодовому виді наведені в табл.10.10.

Коректуючі властивості ПСК.

N_n	код даних	start		1	1	0	0	stop		Кодове представлення даних
0	0000	–	–	–	–	–	–	–	–	– – – + + – – + + +
1	0001	–	–	–	–	–	–	–	–	– – – + + – √ + + +
2	0010	–	–	–	–	–	–	–	–	– – – + + √ – + + +
3	0011	–	–	–	–	–	–	–	–	– – – + + √ √ + + +
4	0100	–	–	–	–	–	–	–	–	– – – + ∧ – – + + +
5	0101	–	–	–	–	–	–	–	–	– – – + ∧ – √ + + +
6	0110	–	–	–	–	–	–	–	–	– – – + ∧ √ – + + +
7	0111	–	–	–	–	–	–	–	–	– – – + ∧ √ √ + + +
8	1000	–	–	–	–	–	–	–	–	– – – ∧ + – – + + +
9	1001	–	–	–	–	–	–	–	–	– – – ∧ + – √ + + +
10	1010	–	–	–	–	–	–	–	–	– – – ∧ + √ – + + +
11	1011	–	–	–	–	–	–	–	–	– – – ∧ + √ √ + + +
12	1100	–	–	–	–	–	–	–	–	– – – ∧ ∧ – – + + +
13	1101	–	–	–	–	–	–	–	–	– – – ∧ ∧ – √ + + +
14	1110	–	–	–	–	–	–	–	–	– – – ∧ ∧ √ – + + +
15	1111	–	–	–	–	–	–	–	–	– – – ∧ ∧ √ √ + + +

Функціональним обмеженням є відсутність можливості визначення числа нулів та одиниць в блоці даних.

Принцип формування НРСК полягає в тому, що послідовність нулів, які передаються в пакеті даних, нумерується рекурентним кодом Галуа G_2^k . Причому біт Галуа “1” передається фронтом спаду, тобто маніпуляційним сигналом “10”, а нулі бітів Галуа передаються сигналом “00”. Для передавання одиниць використовується фронт наростання. В результаті такого способу формування сигналів виникає можливість виявлення помилок при передаванні даних, на базі рекурентних властивостей коду Галуа (табл.10.11).

В табл.10.11. введені наступні позначення:

Д- дані, які передаються; SK – сигнальний код каналу зв'язку (Манчестерський код); SK1–сигнальний код передавання "1" в каналах зв'язку; G(0) – код Галуа, який нумерує нулі; SK01 – сигнальний код нулів, коли біт- Галуа "1" передає кодом "11", а біт-Галуа "0" передає кодом "00"; SK02 – сигнальний код нулів, коли біт-Галуа "1" передає "10", а "0" → "00"; GK01

– сигнальні коди маніпуляції, в яких нулі нумеруються кодом Галуа; GK02 – сигнальні коди маніпуляції, в яких нулі нумеруються кодом Галуа по фронту спаду.

Таблиця 10.11.

Реалізація методу сигнального кодування даних, з використанням кодової послідовності Галуа G_2^3 методом НРСК

D	0	0	0	1	0	1	1	0	1	1	1	0	1	0	0	1	1
SK	10	10	10	01	10	01	01	10	01	01	01	10	01	10	10	01	01
SK1				↑		↑	↑		↑	↑	↑		↑			↑	↑
G(0)	1	1	1		0			1				0		0	0		
SKO1																	
SKO2	↓	↓	↓					↓									
GK01				↑		↑	↑		↑	↑	↑		↑			↑	↑
GK02	↓	↓	↓	↑		↑	↑	↓		↑	↑	↑	↑			↑	↑

Сигнальний код GK02, внаслідок використання сигналів фронту спаду для одиничних бітів коду Галуа, забезпечує кращу бітову синхронізацію по відношенню до сигнального коду GK01, в якому використовуються тільки потенціальні сигнали.

Функціональним обмеженням такого коду є відсутність можливості визначення загального числа нулів в блоці даних, оскільки модуляційний код Галуа повторюється багато разів. Крім того, викладений принцип сигнального кодування даних в базисі Галуа, як видно з табл.2, характеризується функціональним обмеженням, який полягає в тому, що при повторенні нулів і їх кодуванні в базисі Галуа потенціальними сигналами «-1» відсутня бітова синхронізація.

Модифікацією НРСК є використання коду Галуа для одночасного симетричного сигнального формування нулів і одиниць потоку даних, при чому об'єм коду Галуа відповідає об'єму даних, що передаються (РССК).

В табл.10.12 показані приклади формування сигналів даної модифікації РССК, при G_2^4 , де сигнали +1 та -1 нульових позицій коду Галуа формуються згідно нульових позицій коду Галуа.

Таблиця 10.12.

Формування сигналів модифікованим РССК, при G_2^4

		d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12	d13	d14	d15	d16
	D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	G_2^4	1	1	1	1	0	1	0	1	1	0	0	1	0	0	0	0
	SK	↓	↓	↓	↓	—	↓	—	↓	↓	—	—	↓	—	—	—	—
	CK	∨	∨	∨	∨	+	∨	+	∨	∨	+	+	∨	+	+	+	+
2	D	1	1	1	1	0	0	0	0	1	0	1	1	0	0	1	1
	$1G_2^4$	1	1	1	1					0		1	0			1	1
	SK1	↑	↑	↑	↑							↑	—			↑	↑
	$0G_2^4$					1	1	1	1		0			1	0		
	SK0					↓	↓	↓	↓		—			↓	—		
	CK	∧	∧	∧	∧	∨	∨	∨	∨	—	+	∧	—	∨	+	∧	∧
3	D	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	G_2^4	1	1	1	1	0	1	0	1	1	0	0	1	0	0	0	0
	SK	↑	↑	↑	↑	—	↑	—	↑	↑	—	—	↑	—	—	—	—
	CK	∧	∧	∧	∧	—	∧	—	∧	∧	—	—	∧	—	—	—	—

В табл.10.12. SK і CK – відповідно сигнальні і символні коди.

2. + √ -- ⇒

3. + - --

Оскільки двох таких стартових комбінацій нулів в базисі Галуа не існує, то помилка однозначно не виявляється лише в одному випадку.

Введемо помилку в третьому розряді даних:

1. + + ∧ -

2. + + √ - ⇒ ++ ■ -

3. + + + -

В третьому розряді даним кодом ПСК помилка не виявляється теж в одному випадку.

Введемо помилку в четвертому розряді:

1. + + - ∧

2. + + - √ ⇒ ++ - ■

3. + + - +

Аналогічно, як і в попередньому випадку не виявляється одна помилка.

Результати досліджень можливості виявлення однократних помилок для всіх кодових комбінацій даних від 0000 до 1111 приведені в табл. 10.14.

Таблиця 10.14

Коректуючі властивості ПСК для 4-х біт даних.

N _п	Код даних	Код ПСК	Помилка в 1-му розряді	Помилка в 2-му розряді	Помилка в 3-му розряді	Помилка в 4-му розряді
	0000	++ --	■ + -- √ + -- - + --	+ ■ -- + √ -- + - --	+ + ∧ - + + ■ - + + + -	+ + - ∧ + + - ■ + + - +
	0001	++ -√	■ + -√ √ + -√ - + -√	+ ■ -√ + √ -√ + - -√	+ + ∧ √ + + ■ √ + + + √	+ + - ∧ + + - ■ + + - +
	0010	++ √-	■ + √ - √ + √ - - + √ -	+ ■ √ - + √ √ - + - √ -	+ + ∧ - + + ■ - + + + -	+ + √ ∧ + + √ ■ + + √ +
	0011	++ √√	■ + √ √ √ + √ √ - + √ √	+ ■ √ √ + √ √ √ + - √ √	+ + ∧ √ + + ■ √ + + + √	+ + √ ∧ + + √ ■ + + √ +
	0100	+ ∧ --	■ ∧ -- √ ∧ -- - ∧ --	+ √ -- + ■ -- + - --	+ ∧ ∧ - + ∧ ■ - + ∧ + -	+ ∧ - ∧ + ∧ - ■ + ∧ - +
	0101	+ ∧ -√	■ ∧ -√ √ ∧ -√ - ∧ -√	+ √ -√ + ■ -√ + - -√	+ ∧ ∧ √ + ∧ ■ √ + ∧ + √	+ ∧ - ∧ + ∧ - + + ∧ - ■
	0110	+ ∧ √-	■ ∧ √ - √ ∧ √ - - ∧ √ -	+ √ √ - + ■ √ - + - √ -	+ ∧ ∧ - + ∧ + - + ∧ ■ -	+ ∧ √ ∧ + ∧ √ ■ + ∧ √ +
	0111	+ ∧ √√	■ ∧ √ √ √ ∧ √ √ - ∧ √ √	+ √ √ √ + ■ √ √ + - √ √	+ ∧ ∧ √ + ∧ + √ + ∧ ■ √	+ ∧ √ ∧ + ∧ √ + + ∧ √ ■
	1000	∧ + --	√ + -- ■ + -- - + --	∧ ■ -- ∧ √ -- ∧ - --	∧ + ∧ - ∧ + ■ - ∧ + + -	∧ + - ∧ ∧ + - ■ ∧ + - +

	1001	$\wedge +$ $- \vee$	$\vee + - \vee$ $\blacksquare + - \vee$ $- + - \vee$	$\wedge \blacksquare - \vee$ $\wedge \vee - \vee$ $\wedge - - \vee$	$\wedge + \wedge \vee$ $\wedge + \blacksquare \vee$ $\wedge + + \vee$	$\wedge + - \wedge$ $\wedge + - +$ $\wedge + - \blacksquare$
0	1010	$\wedge +$ $\vee -$	$\vee + \vee -$ $\blacksquare + \vee -$ $- + \vee -$	$\wedge \blacksquare \vee -$ $\wedge \vee \vee -$ $\wedge - \vee -$	$\wedge + \wedge -$ $\wedge + + -$ $\wedge + \blacksquare -$	$\wedge + \vee \wedge$ $\wedge + \vee \blacksquare$ $\wedge + \vee +$
1	1011	$\wedge +$ $\vee \vee$	$\vee + \vee \vee$ $\blacksquare + \vee \vee$ $- + \vee \vee$	$\wedge \blacksquare \vee \vee$ $\wedge \vee \vee \vee$ $\wedge - \vee \vee$	$\wedge + \wedge \vee$ $\wedge + + \vee$ $\wedge + \blacksquare \vee$	$\wedge + \vee \wedge$ $\wedge + \vee \blacksquare$ $\wedge + \vee +$
2	1100	$\wedge \wedge$ $- -$	$\vee \wedge - -$ $\blacksquare \wedge - -$ $- \wedge - -$	$\wedge \vee - -$ $\wedge \blacksquare - -$ $\wedge - - -$	$\wedge \wedge \wedge -$ $\wedge \wedge \blacksquare -$ $\wedge \wedge + -$	$\wedge \wedge - \wedge$ $\wedge \wedge - \blacksquare$ $\wedge \wedge - +$
3	1101	$\wedge \wedge$ $- \vee$	$\vee \wedge - \vee$ $\blacksquare \wedge - \vee$ $- \wedge - \vee$	$\wedge \vee - \vee$ $\wedge \blacksquare - \vee$ $\wedge - - \vee$	$\wedge \wedge \wedge \vee$ $\wedge \wedge \blacksquare \vee$ $\wedge \wedge + \vee$	$\wedge \wedge - \wedge$ $\wedge \wedge - +$ $\wedge \wedge - \blacksquare$
4	1110	$\wedge \wedge$ $\vee -$	$\vee \wedge \vee -$ $\blacksquare \wedge \vee -$ $- \wedge \vee -$	$\wedge \vee \vee -$ $\wedge \blacksquare \vee -$ $\wedge - \vee -$	$\wedge \wedge \wedge -$ $\wedge \wedge + -$ $\wedge \wedge \blacksquare -$	$\wedge \wedge \vee \wedge$ $\wedge \wedge \vee \blacksquare$ $\wedge \wedge \vee +$
5	1111	$\wedge \wedge$ $\vee \vee$	$\vee \wedge \vee \vee$ $\blacksquare \wedge \vee \vee$ $- \wedge \vee \vee$	$\wedge \vee \vee \vee$ $\wedge \blacksquare \vee \vee$ $\wedge - \vee \vee$	$\wedge \wedge \wedge \vee$ $\wedge \wedge + \vee$ $\wedge \wedge \blacksquare \vee$	$\wedge \wedge \vee \wedge$ $\wedge \wedge \vee +$ $\wedge \wedge \vee \blacksquare$

Розглянемо приклад можливості виявлення помилок при формуванні ПСК даних довільної довжини $N = 2^k$, $k = 2, 3, \dots$, при використанні кодів Галуа типу:

$$G_2^2, G_2^3, G_2^4, \dots, G_2^k.$$

$$G_2^2 = 1100\dots$$

$$G_2^3 = 11101000\dots$$

$$G_2^4 = 1111010110010000\dots$$

$$G_2^{10} = 1111111111010101010110011\dots$$

В табл.10.15 показано приклади формування ПСК для різних кодових комбінацій даних.

В табл.10.15 показано приклади невиявлених помилок в ПСК при різних модулюючих кодах Галуа: G_2^3, G_2^4 , з якої видно, що число невиявлених помилок також становить 25%.

Таблиця 10.15

	start	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12	d13	d14	d15	d16	d17	d18	d19	d20	d21	d22	d23	d24	stop
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	...	
G_2^2	$\begin{smallmatrix} 2+1 \\ \hline \end{smallmatrix}$	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-	...	+++
G_2^3	$\begin{smallmatrix} 3+1 \\ \hline \end{smallmatrix}$	+	+	+	-	+	-	-	-	+	+	+	-	+	-	-	-	+	+	+	-	+	-	-	...	++++
G_2^4	$\begin{smallmatrix} 4+1 \\ \hline \end{smallmatrix}$	+	+	+	+	-	+	-	+	+	-	-	+	-	-	-	-	+	+	+	+	-	+	-	...	+++++
G_2^k	$\begin{smallmatrix} k=10 \\ \hline \end{smallmatrix}$	+	+	+	+	+	+	+	+	+	+	-	+	-	+	-	+	-	+	-	-	+	+	+	...	+++++
		1	1	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	0	0	0	1	1	1	...	
G_2^2	$\begin{smallmatrix} 2+1 \\ \hline \end{smallmatrix}$	$\wedge$	$\wedge$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\vee$	$\vee$	+	$\wedge$	-	$\vee$	+	+	$\vee$	$\vee$	$\wedge$	+	-	-	$\wedge$	$\wedge$	$\vee$	...	+++
G_2^3	$\begin{smallmatrix} 3+1 \\ \hline \end{smallmatrix}$	$\wedge$	$\wedge$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\vee$	$\vee$	+	$\wedge$	+	$\vee$	+	-	$\vee$	$\vee$	$\wedge$	+	+	-	$\wedge$	$\vee$	$\vee$	...	++++
G_2^4	$\begin{smallmatrix} 4+1 \\ \hline \end{smallmatrix}$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\wedge$	+	-	-	$\wedge$	-	-	$\vee$	$\vee$	$\wedge$	+	+	+	$\vee$	$\wedge$	$\vee$	...	+++++
G_2^k	$\begin{smallmatrix} k=10 \\ \hline \end{smallmatrix}$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	+	$\wedge$	-	$\wedge$	-	+	$\vee$	$\wedge$	$\vee$	+	-	+	$\wedge$	$\vee$	$\vee$	...	+++++
		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	...	
G_2^2	$\begin{smallmatrix} 2+1 \\ \hline \end{smallmatrix}$	$\wedge$	$\wedge$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\vee$	...	+++
G_2^3	$\begin{smallmatrix} 3+1 \\ \hline \end{smallmatrix}$	$\wedge$	$\wedge$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\vee$	...	++++
G_2^4	$\begin{smallmatrix} 4+1 \\ \hline \end{smallmatrix}$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\wedge$	$\wedge$	$\vee$	$\vee$	$\wedge$	$\vee$	$\vee$	$\vee$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\vee$	...	+++++
G_2^k	$\begin{smallmatrix} k=10 \\ \hline \end{smallmatrix}$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\wedge$	$\vee$	$\wedge$	$\wedge$	$\vee$	$\vee$	...	+++++

Таким чином при формуванні інформаційних повідомлень на основі ПСК незалежно від довжини об'єму масиву даних та різних модулюючих кодів Галуа ймовірність виявлення помилок постійна і число невиявлених помилок становить 25%. Позитивною характеристикою ПСК є можливість виявлення та в окремих випадках виправлення помилок типу «стирання» та «вставок» бітів, які можуть виникати під впливом мультиплікативних завад. Функціональним обмеженням є відсутність можливості визначення числа нулів та одиниць в блоці даних.

10.2.3 Несиметричний рекурентний сигнальний код (НРСК).

Запропонований автором в роботі [13] принцип формування НРСК полягає в тому, що послідовність нулів, які передаються в пакеті даних нумеруються рекурентним кодом Галуа G_2^k . Причому біт Галуа "1" передається фронтом спаду, тобто маніпуляцій ним сигналом "10", а нулі бітів Галуа передаються сигналом "00". Для передачі одиниць використовують фронт наростання. В результаті такого способу формування сигналів виникає можливість виявлення помилки при передаванні даних (рис.10.17).

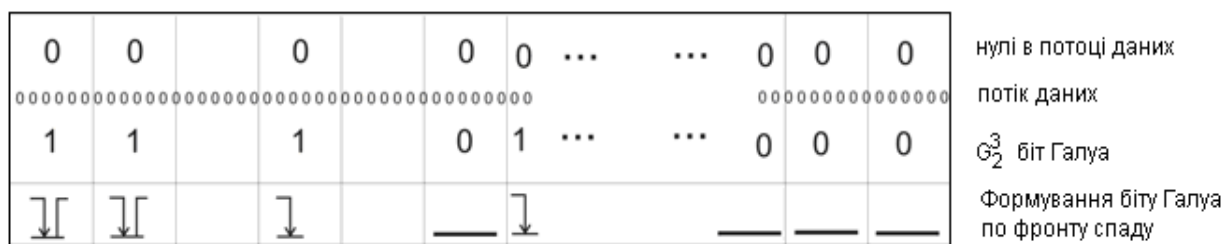


Рис.10.17 Метод формування нулів G_2^3 в потоці даних.

В табл.10.18. введені наступні позначення.:

Д- дані, які передаються;

SK – сигнальний код каналу зв'язку (Манчестерський код);

SK1–сигнальний код передавання "1" в каналах зв'язку;

$G(0)$ – код Галуа, який нумерує нулі;

SK01 – сигнальний код нулів, коли біт- Галуа "1" передає кодом "11", а біт-Галуа "0" передає кодом "00";

SK02 – сигнальний код нулів, коли біт-Галуа "1" передає "10", а "0" —> "00";

GK01 – сигнальні коди маніпуляції, в яких нулі нумеруються кодом

Галуа;

GK02 – сигнальні коди маніпуляції в яких нулі нумеруються кодом Галуа по фронту спаду.

Таблиця 10.18

Метод сигнального кодування двох байт даних, з використанням кодової послідовності Галуа G_2^3 (11101000).

Д	0	0	0	1	0	1	1	0	1	1	1	0	1	0	0	1	1
SK	10	10	10	01	10	01	01	10	01	01	01	10	01	10	10	01	01
SK1				↑		↑	↑		↑	↑	↑		↑			↑	↑
$G(0)$	1	1	1		0			1				0		0	0		
SK01																	
SK02	↓	↓	↓					↓									
GK01				↑		↑	↑		↑	↑	↑		↑			↑	↑
GK02	↓	↓	↓					↓					↓				↓

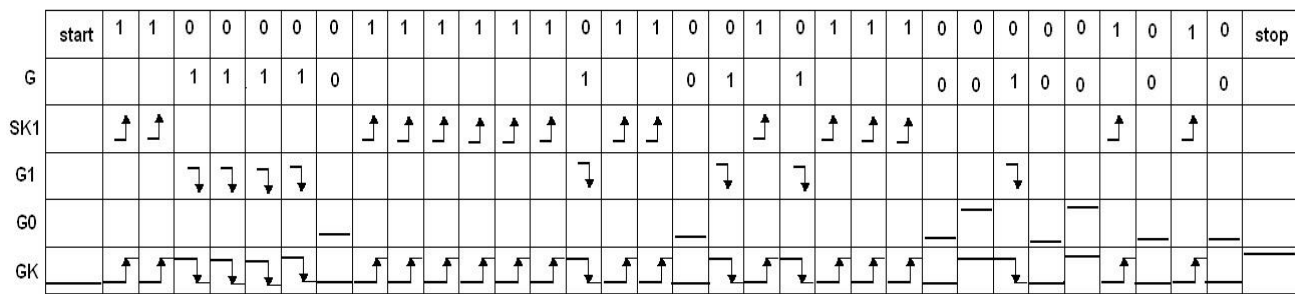
10.2.4. Дослідження надлишковості захисту даних від помилок існуючих протоколів передавання даних.

Сигнальний код GK02, внаслідок використання сигналів фронту спаду для одиничних бітів коду Галуа, забезпечує кращу бітову синхронізацію по відношенню до сигнального коду GK01, в якому використовуються тільки потенціальні сигнали. Крім того, викладений принцип сигнального кодування даних в базисі Галуа, як видно з табл.5.18, характеризується функціональним обмеженням, який полягає в тому, що при повторенні нулів і їх кодуванні в базисі Галуа потенціальними сигналами «-1» відсутня бітова синхронізація. Тому для реалізації ефективної бітової синхронізації запропоновано модифікацію НРСК на основі використання зміни потенціалу «-1» на потенціал «+1» при однократному повторенні Галуа кодованих послідовностей нулів і кодуванню одиниць Галуа фронтом спаду.

В табл.10.19. наведений приклад модифікованого методу безнадлишкового сигнального кодування з бітовою синхронізацією інформації в базисі Галуа, на основі НРСК кодів, який характеризується ефективною блоковою і бітовою синхронізацією, а також можливістю виявлення та виправлення помилок типу зміни значення біту на протилежний, а також стирань та вставок бітів внаслідок кодових завад.

Таблиця 10.19

Метод безнадлишкового кодування 4 байт даних в базисі Галуа, на основі НРСК кодів з бітовою синхронізацією.



При застосуванні безнадлишкового методу сигнального формування даних з захистом від помилок на основі кодів поля Галуа структура фрейму має вигляд, представлений на рис.10.20.

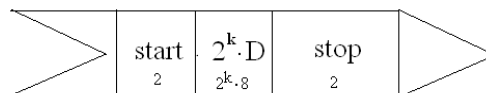


Рис.10.20. Структура фрейму протоколу G1.

Згідно структури фрейму протоколу G1, оцінку надлишковості передавання даних розраховуємо за формулою:

$$K_n (G1) = (4 + 2^k \cdot 8 + V_{kk}) / K_{\text{FC}} \cdot 2^k \cdot 8,$$

де $K_{\text{FC}} (G1) = 0.75$.

Приклад реалізації протоколу G1 в сигнальному просторі ± 1 показано на рис.10.21.

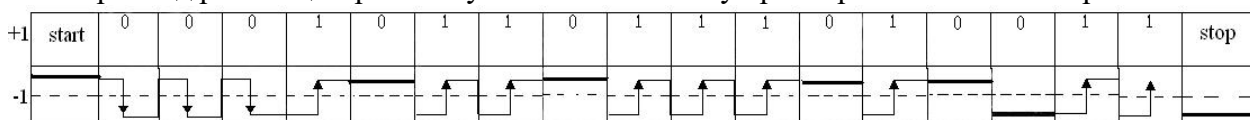


Рис.10.21 Структура протоколу G1 в сигнальному просторі ± 1 .

Для виявлення помилок при передаванні сигналу використовують стандартні методи на основі рекурентних надлишкових коректуючих кодів[118].

Для того, щоб в даних не з'являвся код флага після кожних п'яти одиниць, виконується процедура біт-стаффінга, яка потім вилучається. В результаті код-фрейм має змінну

довжину, що ускладнює процедуру виявлення помилок і перевантажує трафік передачі даних. На рис.10.22 показана структура фрейму стандартного протоколу HDLC, рекомендації МККТ х.25.

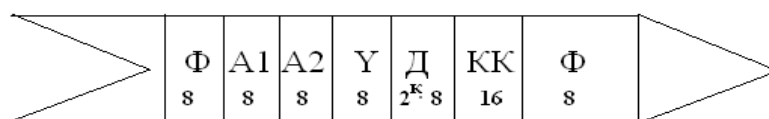


Рис.10.22. Структура фрейму протокола HDLC:

Ф – флаг (байт 01111110); A1,A2 – відповідно адреси передавальної та приймальної станцій; Y – тип кадру; Д – дані ($D=2^k \cdot 8$); КК – коректуючий код БЧХ.

Для дослідження надлишковості, що виникає при передаванні даних, згідно протоколу розглянемо фрагмент вхідних даних (рис.10.23), який внаслідок виникнення операції біт-стафінга передається по інформаційному каналу у вигляді коду:

01111110.001111101.111110

Рис.10.23. Код Галуа

10.2.5. Симетричний рекурентний сигнальний код (СРСК).

Функціональними обмеженнями модифікованого НРСК є відсутність визначення числа нулів в блоці даних в тому випадку, коли довжина модуляційного коду Галуа є менша від довжини коду даних.

Другою модифікацією НРСК є використання коду Галуа для одночасного симетричного сигнального формування нулів і одиниць потоку даних, при чому об'єм коду Галуа відповідає об'єму даних, що передаються (РССК).

В табл.10.20 показані приклади формування сигналів даної модифікації РРСК, при $N=2^4$, де сигнали +1 та -1 нульових позицій коду Галуа формуються таким чином, щоб виключити повторення плюсів і мінусів та забезпечити символну синхронізацію.

Таблица 10.20

Формування сигналів модифікованим РРСК, при $N=2^4$

		d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12	d13	d14	d15	d16
	Д	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	G_2^4	1	1	1	1	0	1	0	1	1	0	1	0	0	0	0	0
	SK	↘	↘	↘	↘	—	↘	—	↘	↘	—	↘	—	—	—	—	—
	CK	∨	∨	∨	∨	+	∨	+	∨	∨	+	∨	+	+	+	+	+
2	Д	1	1	1	1	0	0	0	0	1	0	1	1	0	0	1	1
	G_2^4	1	1	1	1	0	1	0	1	1	0	1	0	0	0	0	0
	SK1	↗	↗	↗	↗	—	—	—	—	—	—	↗	—	—	—	↗	↗
	SK0	—	—	—	—	↘	↘	↘	↘	—	—	—	—	↘	—	—	—
	CK	^	^	^	^	∨	∨	∨	∨	—	+	^	—	∨	+	^	^
3	Д	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	G_2^4	1	1	1	1	0	1	0	1	1	0	1	0	0	0	0	0
	SK	↗	↗	↗	↗	—	↗	—	↗	↗	—	↗	—	—	—	—	—
	CK	^	^	^	^	—	^	—	^	^	—	^	—	—	—	—	—

В табл.10.20 SK і CK – відповідно сигнальні і символні коди.

З табл.10.20 видно, що в блоці даних об'ємом $N=2^4$ завершення послідовності нулів відповідає коду Галуа 1010 і завершується символами ∨+ ∨+, тобто $N=7$, згідно G_2^4 . А

завершення послідовності одиниць в коді Галуа відповідає символам $\wedge - \wedge \wedge$, тобто коду Галуа 1011, $N=9$.

Недоліком даного способу формування сигнального коду є недостатньо ефективна символна (бітова) синхронізація при повторенні потенціальних сигналів +1 або -1.

Таким чином РССК забезпечує ефективне симетричне кодування у вигляді кодів Галуа послідовності нулів і одиниць блоку даних з однозначним визначенням їх числа $N_0 + N_1 = N$, яке може бути використане для виявлення та виправлення помилок після передавання даних в ІК комп'ютерних систем.

10.3. Теоретичні основи оцінки ефективності сигнальних кодів.

Ефективність різних методів маніпуляції можна оцінити на основі виразу.

$$\alpha_c = \frac{k_\phi \cdot d_{ij} \min \sqrt{\log_2 M}}{2\sqrt{E_c}}, \quad (10.17)$$

де d_{ij} - кодова віддаль між маніпульованими сигналами,

M - число ознак маніпуляції,

E_c - середня енергія сигналу,

K_ϕ - коефіцієнт форми сигналу (залежить від базисної функції)

S_i - енергія сигналів в двовірному сигнальному просторі.

E_c розраховується по формулі:

$$E_c = \frac{1}{M} \sum_{i=1}^M S_i. \quad (10.18)$$

В оптичних каналах використовуються виключно двохрівневі сигнали $S_0 = 0$; $S_1 = +1$. Тому $d_{ij} = 1$; $E_c = 0 + 1/2 = 0.5$. Незважаючи на різноманітність даних сигналів, всі вони кодуються в сигнальному вікні з кодовою віддаллю $d_{ij} = 1$, з коефіцієнтом форми $K_f = 0,5$. Використання даного типу сигналів на інтервалі одного сигнального вікна дозволяє збільшити число M (число маніпуляцій) і тим самим підвищити швидкість передавання даних по оптичних каналах зв'язку. Ефективність методів маніпуляції K_{ef} залежить лише від форми сигналів K_f .

Дані аналізу ефективності методів маніпуляції в залежності від форми сигнального вікна представлено у табл. 10.21.

Таблиця 10.21.

Значення коефіцієнта форми сигналу.

Форма сигналу	Коефіцієнт форми сигналу
	$K_\phi = 1$
	$K_\phi = 0,7$
	$K_\phi = 0,5$
	$K_\phi = 0,9$

E_c розраховується за формулою:

$$E_c = \frac{1}{M} \sum_{i=1}^M S_i, \quad (10.19)$$

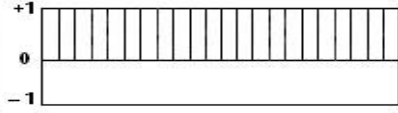
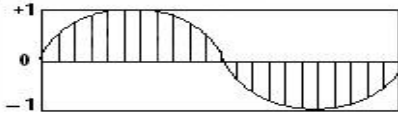
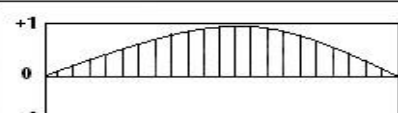
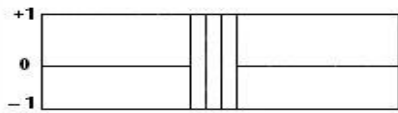
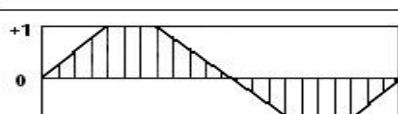
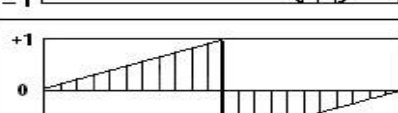
В оптичних каналах використовуються виключно двохрівневі сигнали $S_0 = 0$; $S_1 = +1$. Тому $d_{ij} = 1$; $E_c = 0 + 1/2 = 0.5$.

Ефективність методів маніпуляції K_{ef} залежить лише від форми сигналів K_f .

Дані аналізу ефективності методів маніпуляції в залежності від форми сигнального вікна представлено у таблиці 10.22.

Таблиця 10.22

Сигнальні вікна оптичних сигналів маніпуляції різних форм.

	Тип сигналу	сигнальне вікно	$K_{\text{эф}}$
1	$S=1$		0.5
2	$S=\{1, T < 0.5\}$ $\{0, T \geq 0.5\}$		0.35
3	$S=\sin^2 x$		0.4
4	$\{0, T \leq 0.5\}$ $S=\{1, T < 0.5\}$ $\{0, T > 0.66\}$		0.16
5	$S=\{T < 0.5\}$ $\{T > 0.5\}$		0.33
6	$y = \frac{b}{a_1} x, \quad 0 \leq x \leq a_1$ $y = b, \quad a_1 \leq x \leq a_2$		0.45
7	$y = ax$		0.25

Для підвищення завадозахищеності оптичних каналів зв'язку необхідно використовувати широкосмугові сигнали на основі наступних випадкових послідовностей:

- ШПС (шумоподібні сигнали);
- М-послідовності (послідовності довжини максимуму);
- коди Баркера;
- модифіковані коди Баркера;
- двомірні коди Баркера.

10.4. Теоретичні основи ентропійних методів опрацювання сигналів.

Викладені теоретичні основи та інформаційна теорія визначення ентропії ДІ на основі інформаційних мір, які у світовій практиці для оцінки ентропії інформації найширше застосування отримали інформаційні міри Р. Хартлі, К. Шеннона та „3δ”, та автокореляції на основі інформаційної міри запропонованої д.т.н. проф. Я.М.Николайчуком.

Інформаційні міри Р. Хартлі, К.Шеннона та „3δ”, не враховують автокореляційні характеристики станів ДІ, тому не можуть бути адекватними моделями ДІ, які представлені технологічними системами та технологічними процесами.

Застосування автокореляційної міри ентропії відкрило широкі можливості потужного ентропійного аналізу технологічних процесів, які можуть бути ефективно використані в

задачах підвищення якості продукції, контролю, попередження відхилень від норми та інше.

В той же час на практиці застосовується багато інших оцінок ентропії (табл.10.23), які можуть бути теоретичною основою для методів формування та опрацювання сигналів.

Реалізація ентропійних моделей на основі функцій автокореляції дозволяє зняти існуючі функціональні обмеження, які присутні при розрахунку ентропії на основі інформаційних мір Р. Хартлі та К. Шеннона. Оскільки перша не враховує статистичного розподілу потоку інформаційних даних, а друга враховує тільки ймовірності станів джерел інформації і не враховує ймовірностей їх переходу з одного стану в інший, що реалізується в інформаційній мірі ентропії на основі формули, запропонованої проф. Николайчуком Я.М.

Прикладом кодів, які володіють високими ентропійними характеристиками є сигнальні коректуючі коди в базисі Галуа.

Таблиця 10.23

Формули оцінки ентропії

№	Найменування формул оцінки ентропії	Міра ентропії
1.	$H = \log S^n = n \cdot \log S$, де H – ентропія; S – число незалежних рівноймовірних станів джерел інформації(ДІ); n – довжина вибірки.	Р.Хартлі
2.	$I_x = \log_2 \sqrt{2\pi e \sigma_x^2}$, $I_x = \hat{E}[\log_2 3\sigma_x]$ $\hat{E}[\]$ – цілочисельна функція з округленням до більшого	К.Крампа
3.	$H_\varepsilon \leq T / \Delta t + \log(C / \varepsilon)$, де Δt – крок дискретизації, що забезпечує точність квантування ε ; C – діапазон квантування; T – інтервал часу спостереження ДІ.	Н.Колмого-рова
4.	$H = -k \sum_{i=1}^n p_i \cdot \log p_i$, де k – додатний коефіцієнт, що враховує основу логарифма; p_i – ймовірність i -го стану дискретного ДІ.	К.Шеннона
5.	$h_\Delta = f'_{\text{сер}}(t) / f'_{\text{max}}(t) $, де $f'_{\text{сер}}(t)$, $f'_{\text{max}}(t)$ – відповідно середнє і максимальне значення похідних зміни кількості станів джерела.	В.Боюна
6.	$I_x = n \cdot \hat{E} \left[\frac{1}{2} \log_2 \frac{1}{m} \sum_{j=1}^m (D_x^2 - R_{xx}^2(j)) \right]$, де D_x – дисперсія значень x_i ; $R_{xx}(j)$ – автокореляційна функція; m – число точок функції $R_{xx}(j)$ на інтервалі кореляції.	Я.Николайчука

7.	$H(u, p) = -k \sum_{i=1}^n [u_i p_i \cdot \log p_i]$, де u_i – коефіцієнт корисності; k – стала величина; $p = p_i$ – імовірність i -го стану.	Дж. Лонго
8.	$H(P, W) = - \sum_{i=1}^n \frac{P_i W_i}{\sum_{j=1}^n P_j W_j} \cdot \log \frac{P_i W_i}{\sum_{j=1}^n P_j W_j}$, де $P_i W_j$ – оціночні коефіцієнти.	Г. Шульца
9.	$H = \lim[(\log N)/n] = - \sum p(j) \cdot \log p(j)$, де $N = n! / \prod_j S_j$ – загальна кількість можливих комбінацій станів	Б. Олівера
10.	$H(X) = - \sum_{l_1}^L \dots \sum_{l_n}^L p(X) \log p(X)$, де $(1 \leq l_i \leq L; i = 1, 2, \dots, n)$. X – апіорна невизначеність; X_i, Y_i – статистично залежні стани ДІ	Д. Мідлтона
11.	$H \leq k 2 B T (1 + S / N)$ $H = k \cdot n \log S_{ave}$, де S_{ave} – середнє значення станів ДІ; BT – інформаційна база повідомлень, що формується; N – значення шуму. $1/S$ – інтервал кореляції між відліками	В. Тайллера

10.4.1. Ентропійні характеристики двовимірних та одновимірних ШПС

Розвиток нових інформаційних технологій обумовлює пошук оптимальних каналів їх розподілу. Для організації швидких ліній зв'язку використовують оптичні системи та радіоканали, які є досить насиченими та характеризуються високим рівнем завад. Для передачі інформації в каналах зв'язку використовують шумоподібні сигнали (ШПС), які найкращим чином дозволяють передавати сигнал із великим рівнем захисту від помилок. В якості ШПС використовуються коди Баркера та М-последовності.

Оскільки шумоподібні характеристики сигналу є важливими для аналізу систем зв'язку, то їх слід розглядати з точки зору ентропійного підходу, що дозволить оцінити надмірність одновимірних та двовимірних ШПС.

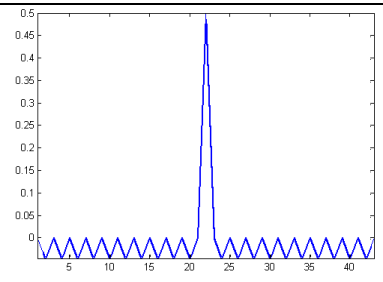
10.4.2. Кореляційні характеристики ШПС.

Для оцінки ШПС використовуються 8 типів кореляційних функцій, які характеризуються різними параметрами точності та алгоритмічної складності.

В табл. 10.24. подані аналітичні вирази та кореляційні функції кодів Баркера.

Таблиця. 10.24

Аналітичні вирази та кореляційні функції кодів Баркера

№	Назва функції	Аналітичний вираз	Кореляційна функція ШПС
1	Знакова	$H_{xx}(j) = \frac{1}{n} \sum_{i=1}^n \text{sign}(x_i) \times \text{sign}(x_{i+j})$	

2	Релейна	$B_{xx}(j) = \frac{1}{n} \sum_{i=1}^n x_i \times \text{sign}(x_{i+j})$	
3	Коваріаційна	$K_{xx}(j) = \frac{1}{n} \sum_{i=1}^n x_i \times x_{i+j}$	
4	Кореляційна	$R_{xx}(j) = \frac{1}{n} \sum_{i=1}^n x_i \times x_{i+j}$	
5	Нормована кореляційна	$\rho_{xx}(j) = \frac{R_{xx}(j)}{D_x}$	
6	Структурна	$C_{xx}(j) = \frac{1}{n} \sum_{i=1}^n (x_i - x_{i+j})^2$	
7	Модульна	$G_{xx}(j) = \frac{1}{n} \sum_{i=1}^n x_i - x_{i+j} $	
8	Еквівалентно-сті	$F_{xx}(j) = \frac{1}{n} \sum_{i=1}^n \tilde{Z}_{xx}$	

10.4.3. Оцінка ентропії одновимірних та двовимірних ШПС.

Очевидно, що використання різних ШПС має різну ефективність їх застосування. Для оцінки ефективності використання різних шумоподібних сигналів необхідно провести розрахунок на основі ентропійного підходу [4].

Розроблена аналітика розрахунку ентропії на основі кореляційних функцій

$$H(x_i, x_{i+j}) = \log_2 2\pi e \sqrt{D_x^2 - D_x^2 \cdot \rho_{xx}^2(j)} = \log_2 2\pi e + \frac{1}{2} \log_2 ([D_x - K_{xx}(j)] \cdot [D_x + K_{xx}(j)])$$

На рис.10. 24 приведений розрахунок кореляційних характеристик та ентропії еталонного 7-бітного коду Баркера з захисним інтервалом

1 1 1 -1 -1 1 -1,

який формується у вигляді восьмирівневої нецентрованої кодової послідовності

8 8 8 0 0 8 0 4 4 4 4 4 4 4

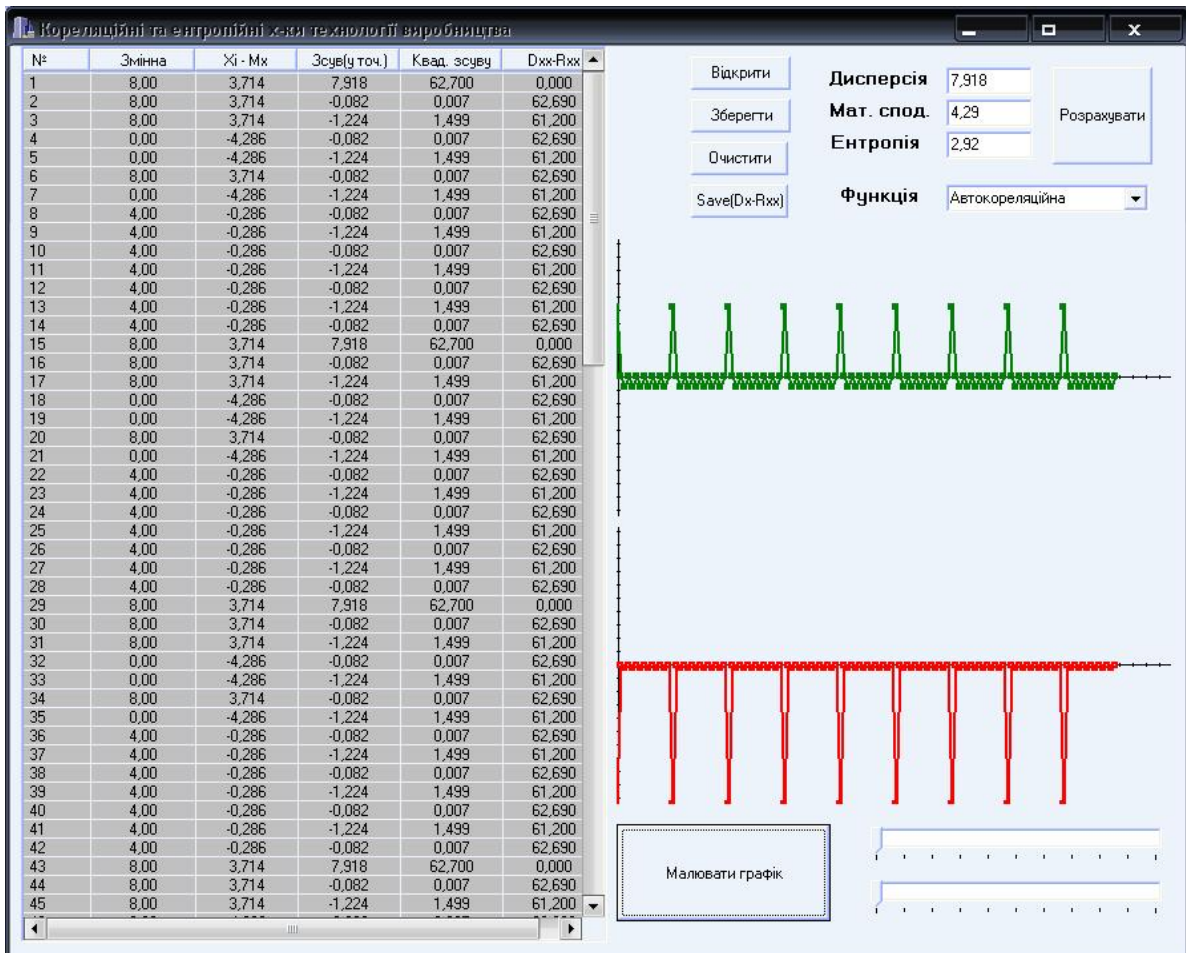


Рис. 10.24. Оцінка ентропії коду Баркера.

На рис.10.25 показаний приклад розрахунку ентропії та кореляційних характеристик 7-бітного коду М-сигналу

1 1 1 0 0 1 0,

Який формується у вигляді восьмирівневої нецентрованої кодової послідовності

8 8 8 0 0 8 0

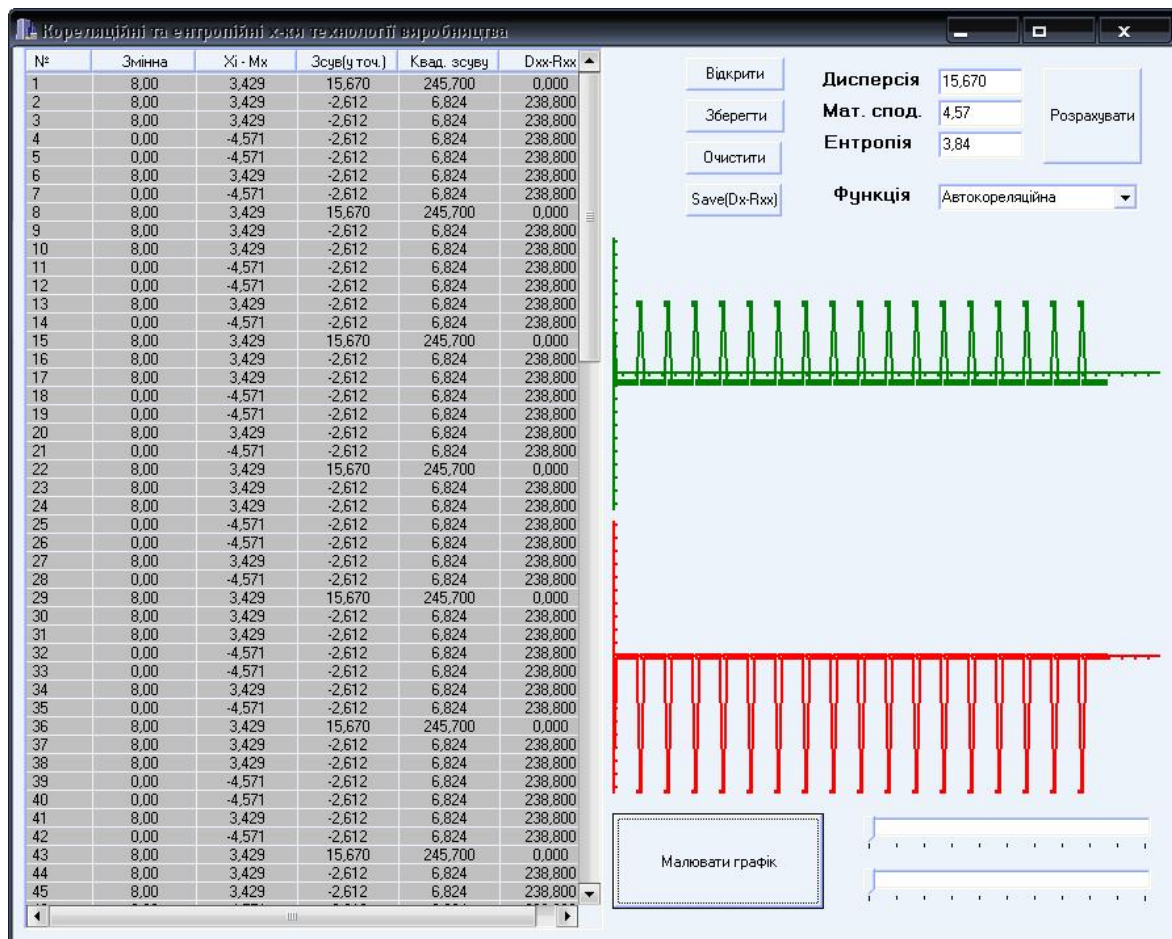


Рис. 10.25. Оцінка ентропії М-сигналу.

Проведено моделювання автокореляційних характеристик ДІ на основі двовимірних кодів Баркера та знайдено найефективніші двовимірні ШПК та їх кореляційні функції, що дозволяє оцінити ефективність двовимірних ШПК в порівнянні з одновимірними з точки застосування ентропійного підходу. На рис.10.26 приведена діаграма ентропій одновимірних та двовимірних ШПС.

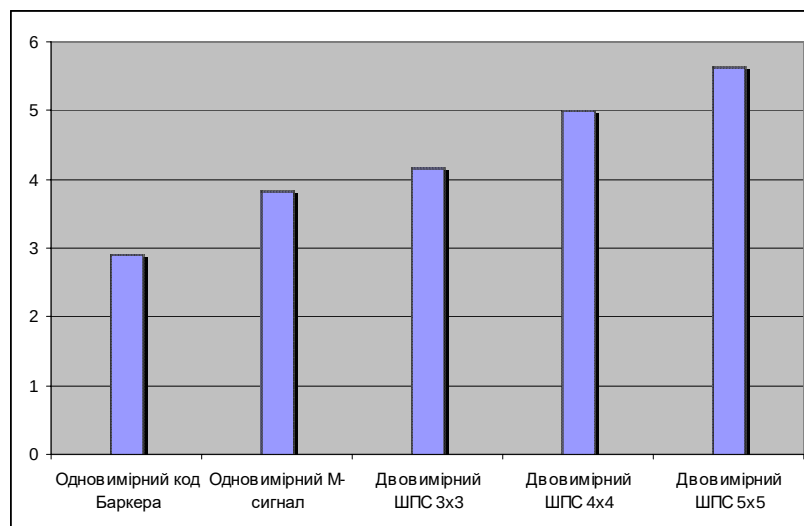


Рис. 10.26. Діаграма ентропій одновимірних та двовимірних ШПС.

З рис.10.26 видно, що двовимірні шумоподібні сигнали мають значно кращі ентропійні характеристики у порівнянні з одновимірними ШПС.

Значний вплив на кореляційні характеристики ШПК обумовлює поява певної кількості помилок, що значно погіршує ентропійні характеристики ШПС.

Для виявлення погіршення взаємкореляційних властивостей вибраних ШПК в досліджувану кодову послідовність, для якої розраховувалась взаємо кореляційна функція, вводились однократні та багатократні помилки, які полягали в інвертуванні одного чи кількох бітів коду. Оскільки досліджувані коди при інвертуванні певних бітів мало чутливі до цих змін, то з отриманих результатів вибирались ті, що приводять до найбільшого зниження ефективності кодової послідовності.

На рис.10.27 показана діаграма зміни ентропійних характеристик двовимірних та одновимірних ШПС в залежності від кількості введених помилок.

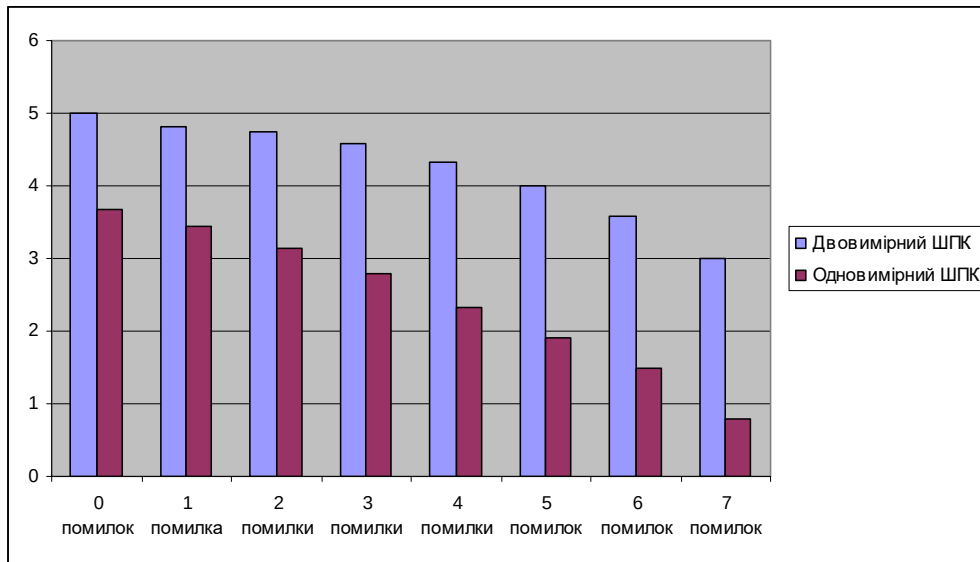


Рис. 10.27 Ентропія кореляційних функцій ШПК з помилками.

Проведені дослідження методики сигнального кодування даних на основі коду Галуа з використанням рекурентних кодів, в оптичних сигналах у відкритих лініях зв'язку, показують, що можливості оптимізації та покращення ефективнішого методу передачі даних можуть бути ефективно реалізовані на основі безнадлишкового кодування даних, при використанні сучасних методів цифрової обробки сигналів. Даний метод дозволяє це реалізувати невикористані біти, для ефективнішого обміну даними.

А також проведені дослідження які дозволяють сумісно оцінити ентропійні характеристики різних типів ШПС. Які можуть зустрічатися в різних каналах зв'язку на основі ентропійних властивостей. Двовимірні ШПС мають більшу ентропійну інформативність в порівнянні з одновимірними сигналами.

РОЗДІЛ 11

КРИТЕРІЇ ЕМЕРДЖЕНТНОСТІ ПРИ ОРГАНІЗАЦІЇ МОДЕЛЕЙ БАЗ ДАНИХ

Сучасні технології БД є одним із визначних факторів вдосконалення процесів зберігання та доступу до великих масивів даних. Потужність сучасних БД базується на результатах досліджень, та технологічних розробках які отримані на протязі кількох останніх десятиріч. Управління БД виконується спеціалізованими програмами (СУБД), які є ефективним інструментом збору. Особливим видом бази даних є розподілені бази даних, важливими характеристиками яких є ступінь паралелізму операцій та швидкого пошуку даних. Слід зауважити, що технології розподіленої обробки інформації ускладнюють всі компоненти і змушують переглядати їх структури аж до базових елементів. При чому в багатьох розподілених БД затрати на підтримку телекомунікацій можуть переважити вартість обчислювальних процесів та програмно-технічного забезпечення.

Тому при дослідженні та проектуванні розподілених БД найважливішу роль отримують задачі компактного кодування даних та швидкість обміну даними, при заданій пропускну здатності каналного зв'язку.

11.1. Оцінка надлишковості бази даних

Розрахунок системних характеристик: дослідження емерджентності в різних типів баз даних, оцінка надлишковості і зменшення надлишковості в різних моделях БД досягається за рахунок зростання числа зв'язків між об'єктами БД. Таким чином оцінку надлишковості моделі БД можна розрахувати на основі критерію емерджентності (інтелектуальності).

$$K_e = \frac{N_3}{N_0}, \quad (11.1)$$

де N_3 - число зв'язків, N_0 - число елементів БД.

Система об'єктів (наприклад: даних) характеризується емерджентністю в тому випадку коли

$$K_e > 2. \quad (11.2)$$

Результатами дослідження емерджентності різних типів БД приведені в таблиці 11.1, звідки видно, що ієрархічні структури ніколи не забезпечують умови емерджентності.

11.2. Критерії та методи підвищення ефективності БД

Важливими характеристиками ефективності БД є:

-емерджентність $K_e = \frac{N_3}{N_0}; \quad (11.3)$

-надлишковість $K_c = \frac{mR}{mG}; \quad (11.4)$

-швидкість пошуку даних $V = \frac{\Delta N_i}{T}; \quad (11.5)$

де: N_3 - число зв'язків, N_0 - число елементів,

mR - ентропія ієрархічних даних в базисі Радемахера, визначається згідно формули Хартлі

$$V_i = \hat{E}[\log_2 m_i]; i \in 1, n \quad (11.6)$$

$$\sum_{i=1}^n V_i = \sum_{j=1}^k \hat{E}[\log_2 m_j] \quad (11.7)$$

де V_i - елемент коду, $\sum_{i=1}^n V_i$ - загальний об'єм кодів,
 mG – ентропія рекурентних даних в базисі Галуа,
 ΔN_i - число тактів ідентифікації даних,
 T - час доступу до даних.

11.3. Фактори які впливають на підвищення ефективності БД

Основними характеристиками оптимізації і підвищення ефективності є:

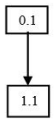
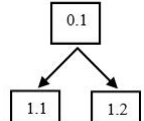
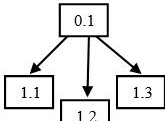
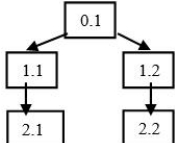
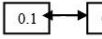
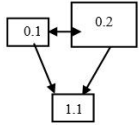
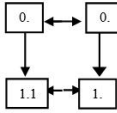
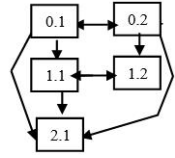
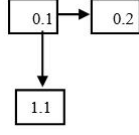
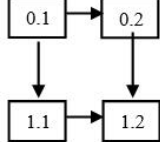
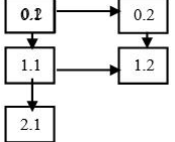
- здешевлення системи СУБД ;
- збільшення швидкості пошуку доступу до даних;
- зменшення або стиснення об'єму інформації ;
- розробка різноманітних типів третинних пристроїв зберігання (tertiary storage devises) даних;
- тенденції росту системи ;
- підвищення продуктивності системи ;
- паралельні обчислення;
- розвиток системи клієнт/сервер і багаторівневої архітектури;
- підвищення надійності захисту від помилок;
- захист від несанкціонованого доступу до БД тощо.

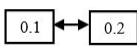
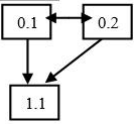
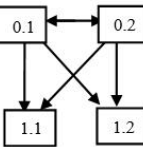
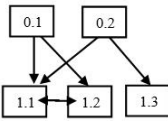
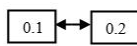
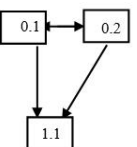
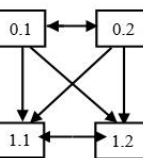
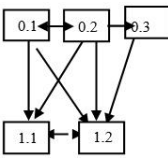
11.4. Результати досліджень емерджентності різних типів моделей баз даних

Проведені дослідження архітектури БД показують, що ієрархічна, мережеві та реляційні архітектури БД не задовільняють умови емерджентності. Тому перспективним є розвиток та удосконалення архітектури БД об'єктно-орієнтованих та об'єктно-реляційних архітектур. Результати досліджень наведені в табл.11.1.

Таблиця 11.1

Результати досліджень емерджентності різних типів моделей баз даних

Тип моделі БД	$N_0=2$	$N_0=3$	$N_0=4$	$N_0=5$	$N_0...$	$N_0=10$
Ієрархічна	$K_e = \frac{1}{2} = 0.5$ 	$K_e = \frac{2}{3} = 0.6$ 	$K_e = \frac{3}{4} = 0.75$ 	$K_e = \frac{4}{5} = 0.8$ 	...	$e < 1$
Мережева	$K_e = \frac{2}{2} = 1.0$ 	$K_e = \frac{4}{3} = 1.3$ 	$K_e = \frac{6}{4} = 1.5$ 	$K_e = \frac{9}{5} = 1.8$ 	...	$e < 2$
Реляційна	$K_e = \frac{1}{2} = 0.5$ 	$K_e = \frac{2}{3} = 0.6$ 	$K_e = \frac{4}{4} = 1.0$ 	$K_e = \frac{5}{5} = 1.0$ 	...	$e < 2$

Об'єктно-орієнтована	$K_e = \frac{2}{2} = 1.0$ 	$K_e = \frac{4}{3} = 1.3$ 	$K_e = \frac{5}{4} = 1.5$ 	$K_e = \frac{8}{5} = 1.6$ 	...	$e < 4$
Об'єктно-реляційна	$K_e = \frac{2}{2} = 1.0$ 	$K_e = \frac{4}{3} = 1.3$ 	$K_e = \frac{8}{4} = 2.0$ 	$K_e = \frac{11}{5} = 2.2$ 	...	$e < 4$

Проведені дослідження архітектур БД показують, що ієрархічна, мережеві та реляційні архітектури БД не задовільняють умови емерджентності. Тому перспективним є розвиток та вдосконалення об'єктно-орієнтованих та об'єктно-реляційних архітектур БД.

РОЗДІЛ 12

СПОСОБИ ДОСТУПУ ДО БАЗ ДАНИХ ГЛОБАЛЬНИХ РОЗПОДІЛЕНИХ КОМП'ЮТЕРНИХ СИСТЕМ

12.1. Керування правами доступу

Визначити права доступу користувачів –це означає зафіксувати інформацію стосовно:

- осіб, яким надається права доступу;
- умов надання прав доступу;
- об'єктів, на які поширюються права доступу;
- операцій, щодо яких специфікуються права доступу;
- можливості передавання прав доступу іншими особами.

Схематично аспекти керування правами доступу зображені на рис.12.1.

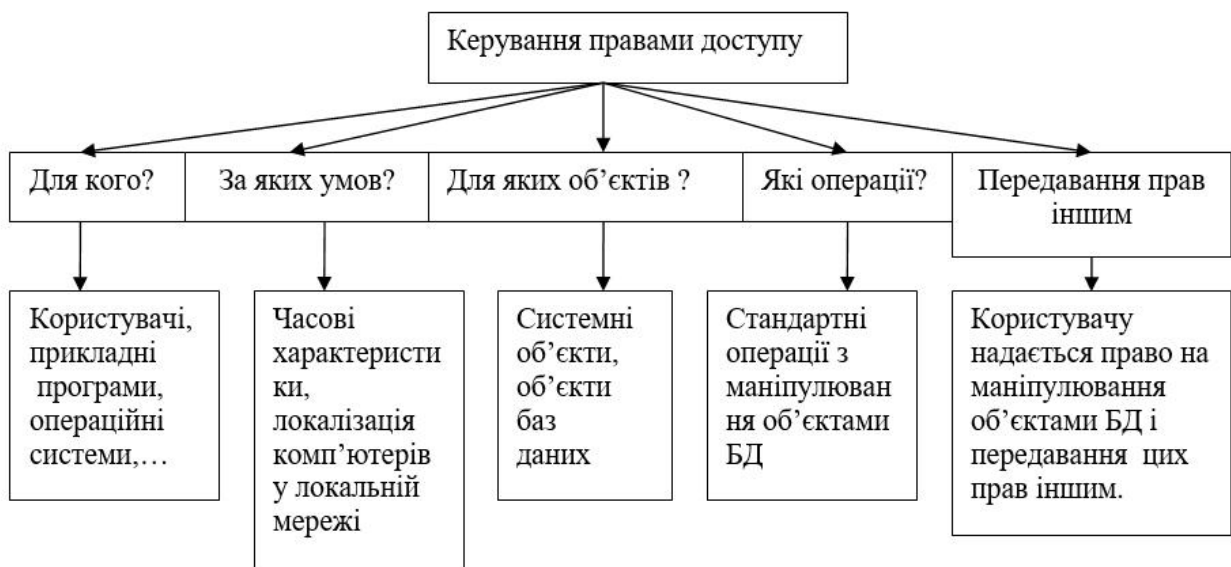


Рис.12.1. Складові прав доступу користувачів

12.2. Середовища доступу до БД глобальних розподілених комп'ютерних систем в базисі Галуа

Існують три основні середовища доступу до БД ГКС:

Web-interface доступ до архівів БД. Підключення баз даних MySQL до сайтів, а також управління даними через web -інтерфейс (тобто через браузер).

Archie-доступ до архівів БД. Швидкий ріст об'єму інформаційних ресурсів привів до необхідності створення спеціальних пошукових засобів: перша пошукова система Archie для пошуку файлів в базах даних ARCHIE (допомагає знаходити файли, які зберігаються на анонімних FTP-серверах).

Gopher – доступ до архівів БД. Система Gopher, система WAIS в базі даних, в яких зберігається індексована інформація про ресурси мережі Інтернет, пошукові системи або пошукові машини, які виконують пошук WWW-сторінок на Web-серверах.

12.3. Пошукові системи в Інтернет

Системи для пошуку ресурсів можуть бути розподілені на наступні групи:– системи Web-пошуку; – системи пошуку FTP-файлів (наприклад, (<http://www.filesearch.ru>); – системи

пошуку в архівах Gopher; – системи пошуку в Usenet; – каталоги інформаційних ресурсів; – портали інформаційних ресурсів.

Послідовний доступ (*sequential access memory, SAM*) — комірки пам'яті вибираються (зчитуються) послідовно, одна за другою, в черзі їх розташування. Варіант такої пам'яті — стекова пам'ять. Довільний доступ (*random access memory, RAM*) — обчислювальний пристрій може звернутися до довільної комірки пам'яті з довільним адресом.

Використовують три основні варіанти Web-доступу до баз даних.

Одноразове або періодичне перетворення вмісту БД у статичні документи.

За цього варіанту зміст БД переглядає спеціальна програма-перетворювач, яка створює множину файлів — низку HTML-документів. Отримані файли копіюються на WWW-сервер. Доступ до них здійснюється як до статичних гіпертекстових документів сервера. Цей варіант характеризується мінімальними початковими витратами. Він ефективний у разі використання невеликих масивів даних простої структури з нечастим оновленням, а також у разі занижених вимог до актуальності даних, що доставляються за допомогою WWW. Крім того, для нього характерна цілкова відсутність механізму пошуку, але можливе використання індексування. Як перетворювач може використовуватися програмне забезпечення, яке автоматично або напівавтоматично генерує статичні документи. Програма-перетворювач може бути або самостійно розробленою, або ж інтерпретованим засобом з-поміж існуючих на ринку різноманітних програм типу генераторів звітів.

Динамічне створення гіпертекстових документів на основі вмісту БД.

У цьому варіанті доступ до БД здійснюється за допомогою спеціальної CGI (Common Gateway Interface) — програми, що запускається Web-сервером у відповідь на запит Web-клієнта. Ця програма, обробляючи запит, переглядає зміст БД і створює вихідний HTML-документ, який повертається клієнту (рис. 12.2).

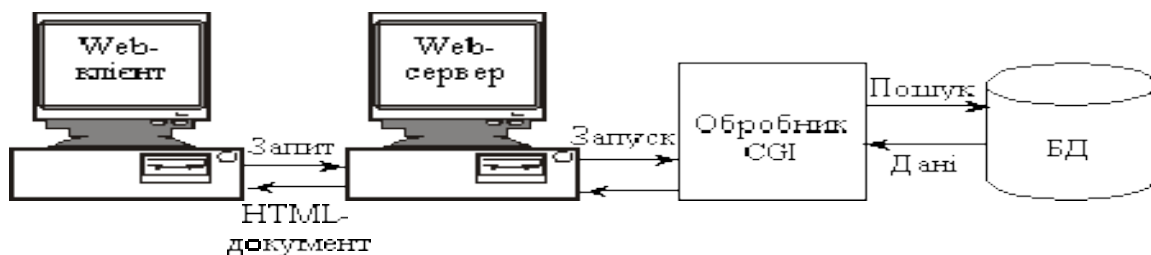


Рис. 12.2. Схема динамічного створення гіпертекстових документів

Використання цього варіанта ефективне для великих баз даних зі складною структурою та в разі необхідності підтримки операцій пошуку, а також за частого оновлення і неможливості синхронізації перетворення БД у статичні документи з оновленням вмісту БД. За цього варіанта можливо вносити зміни до БД за допомогою WWW-інтерфейсів. Для реалізації такої технології необхідно використати взаємодію Web-сервера з програмами CGI, що запускаються. Вибір програмних засобів для реалізації цього варіанта достатньо широкий. Це мови програмування (C++, Perl, Ada) та інтегровані засоби типу генераторів звітів. Під час використання сучасних реляційних

СУБД із внутрішніми мовами програмування можливе використання цих мов для генерації документів. До вад варіанта Web-доступу до БД за допомогою CGI можна віднести значний час оброблення запитів, необхідність постійного доступу до основної бази даних, додаткове завантаження засобів підтримки БД, пов'язане з обробленням запитів від Web-сервера.

Створення інформаційного сховища на базі високопродуктивної СУБД з мовою запитів SQL.

Цей варіант є найбільш вдалим з погляду розвитку сучасних концепцій сховищ даних. В основі його лежить періодичне завантаження даних у сховище з традиційних СУБД. Для оброблення різноманітних запитів, зокрема й від Web-сервера, використовується проміжна БД

(інформаційне сховище) з високою продуктивністю. Інформаційне наповнення сховища даних здійснюється спеціалізованим програмним забезпеченням на основі вмісту основних баз даних і охоплює два етапи — перевантаження даних з основних БД до сховища даних (СД) і оброблення запитів (рис. 12.3. і 12.4. відповідно).

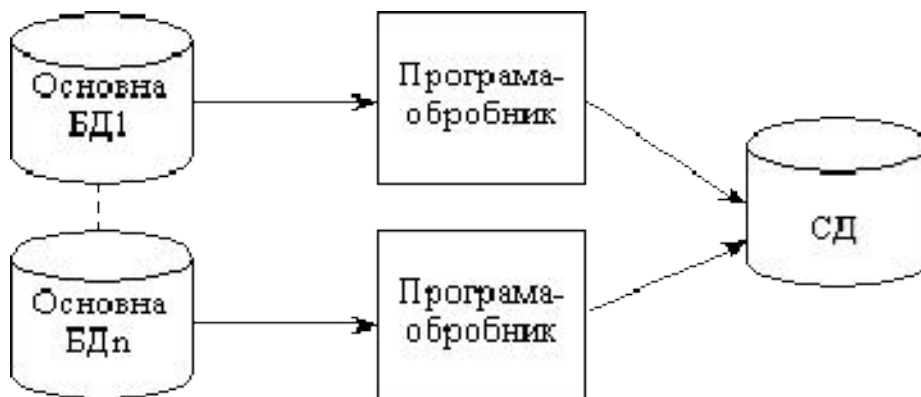


Рис. 12.3. Перевантаження даних до сховища

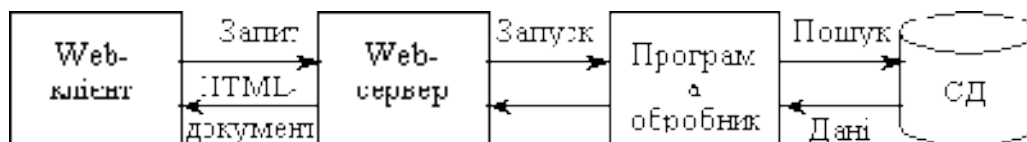


Рис. 12.4. Опрацювання запитів

Цей варіант позбавлений усіх вад попередньої схеми. Крім того, після встановлення синхронізації даних інформаційного сховища з основними БД можливе перенесення користувацьких інтерфейсів на інформаційне сховище, що істотно підвищить надійність і продуктивність доступу, дозволить організувати розподілені атоматизовані робочі місця. Основою підвищення продуктивності оброблення Web-запитів і різкого підвищення швидкості розроблення Web-інтерфейсів є використання внутрішніх мов СУБД інформаційного сховища для створення гіпертекстових документів. Для перевантаження даних з основної бази до сховища можуть використовуватися мови програмування, інтегровані засоби, а також спеціалізовані засоби перевантаження, що постачаються із SQL-сервером, і продукти підтримки інформаційних сховищ. Провідні розробники СУБД інтенсивно шукають постреляційні рішення (POSTGRES), оскільки, що можливості широкопоширеної реляційної ідеології майже вичерпалися. Найімовірніше, у найближчі роки відбудеться відхід від реляційної побудови баз даних і перехід до об'єктно-орієнтованих СУБД. Напрямок POSTGRES належить до числа так званих постреляційних систем, тобто до наступного етапу в розвитку реляційних СУБД. Як відомо, реляційним СУБД притаманна деяка обмеженість під час використання в нетрадиційних галузях, в яких потрібні гранично складні структури даних. Іншим недоліком реляційних баз даних є неможливість адекватного відображення семантики предметної галузі. Крім того, суто реляційна модель і реалізовані на ній SQL-сервери потребують потужного устаткування й далеко не завжди дають змогу провести оптимізацію запитів, а тим більше, забезпечити роботу з постійно оновлюваними даними в реальному масштабі часу. Тому сучасні дослідження в галузі постреляційних систем присвячені переважно усуненню зазначених вад і наданню їм якостей, яких немає в більшості реляційних СУБД. Такими якостями є, наприклад, повнота системи типів, підтримка ієрархії і успадкування типів, можливість управління складними об'єктами.

Доступ до баз даних зі сторони сервера.

Механізм реалізується за рахунок наявності двох більш або менш стандартизованих засобів: можливості включення форм в документ, складений з використанням мови

гіпермедійної розмітки HTML, і можливості використання зовнішніх відносно до серверу Web програм, взаємодія яких відбувається через специфікований протокол CGI (Common Gateway Interface) або упроваджений пізніше API (Application Program Interface). (Хоч CGI називається "загальним інтерфейсом шлюзування", насправді, це одночасно деяка підмножина протокола HTTP, і спосіб його дотримання при взаємодії сервера з зовнішньою програмою.) При реалізації на основі CGI загальна схема реалізації доступу до бази даних на стороні Web-сервера виглядає наступним чином: при перегляді документа клієнт зустрічає посилання на сторінку, яка містить одну або декілька форм, призначених для запиту даних з бази даних; клієнт запитує цю сторінку; крім того, незаповнених форм сторінок може містити загальну інформацію про бази даних та призначення пропонованих форм; якщо клієнта дійсно цікавить інформація з баз даних, яку можна отримати на підставі запропонованих форм, то він заповнює одну з форм і відправляє заповнену форму на сервері; отримавши заповнену форму, сервер запускає відповідну зовнішню програму, передаючи їй параметри отримуючи результати на підставі протокола CGI; зовнішня програма перетворює запит, виражений з допомогою заповненої форми, в запит на мові, зрозумілій серверу баз даних (звично це мова SQL) [92,157,172].

Доступ до баз даних з сторони клієнта.

Очевидно, найбільш потужні засоби забезпечення доступу до баз даних зі сторони Web-клієнта забезпечує мова Java. Java - це об'єктно -орієнтована мова програмування, яка є, насправді, "безпечним" підмножиною мови C++. У випадку, Java не містить засобів адресної арифметики, не підтримує механізм чисельного успадкування і т. п. Тому стверджується, що коректність Java-програми можна перевірити до її реального виконання (це абсолютно недоведене твердження). Розрізняють:—мова Java, як такої, для якої існують компілятори в так званий "мобільний код" (машинно-незалежний код, який може інтерпретуватися або з якого можуть генеруватися машинні коди на різних платформах);—мова JavaScript, яка звичайно використовується для розширення можливостей мови HTML за рахунок додавання процедурною складовою; програмний продукт HotJava, є, насправді, інтерпретатором мобільних кодів Java. Для забезпечення доступу до баз даних на стороні Web-клієнта найбільш суттєвою є мова Java. Технологія розробки HTML-документа дозволяє написати довільну кількість додаткових Java-програм, відкомпілювати їх в мобільні коди і поставити посилання на відповідні коди в тілі HTML-документа. Такі додаткові Java-програми називаються апплетами (Java-applets). Отримавши доступ до 37 документів, яка містить посилання на апплети, клієнтська програма перегляду запитує у Web-сервера всі мобільні коди. Коди можуть розпочати виконуватись відразу після розміщення в комп'ютері клієнта або бути активізованими з допомогою спеціальних команд. При виборі СКБД необхідно визначити, з допомогою яких засобів розробки буде створюватися інформаційна система на основі даної СКБД, а також те, яким чином розроблені застосування будуть маніпулювати даними. Від того, чи вірно вибраний механізм доступу до даних, залежить ефективність системи доступу до даних. Існують декілька способів доступу до баз даних. Переважна більшість СКБД містить в своєму складі бібліотеки, надає спеціальною прикладною програмою інтерфейс (application programming interface, api) для доступу до даних цієї СКБД. Використання клієнтського api (або клієнтських СОМ-об'єктів) є найбільш ймовірним способом маніпулювання даними в застосуваннях. Однак в цьому випадку створене застосування може використовувати дані

тільки СКБД цього виробника, і заміна її на іншу потягне за собою переписування значної частини кода клієнтського застосування - клієнтські api, а об'єктні моделі не підпорядковуються ніяким стандартам і різноманітні для різноманітних СКБД.

Інший спосіб маніпуляції даними в застосуванні базується на використанні універсальних механізмів доступу до даних.

Універсальний механізм доступу до даних звичайно реалізований у вигляді бібліотек і додаткових модулів, які називаються драйверами або провайдерами. Бібліотеки містять деякий стандартний набір функцій або класів, нерідко підпорядкований цій або іншій специфікації.

Додаткові моделі, специфічні для тої чи іншої СКБД, реалізує безпосереднє звернення до функцій клієнтського арі конкретних СКБД. Відмітимо, що достоїнством універсальних механізмів є можливість застосування одного і того ж абстрактного арі, а в багатьох випадках - СОМ-серверів, компонентів, класів для доступу до різноманітних типів СКБД. Тому застосування, використовуючі універсальні механізми доступу до даних, легко модифікувати, якщо необхідна зміна СКБД. Найбільш поширеними серед універсальних механізмів доступу до даних можна назвати наступні: open database connectivity (odbc). ole db. activex data objects (ado). borland database engine (bde). Універсальні механізми odbc, ole db и ado фірми microsoft представляють собою насправді промислові стандарти. [92,157,172]. Модель мобільного доступу. В загальному виді модель, яка описує організацію мобільного доступу до віддалених баз даних, подана на рис.12.5.

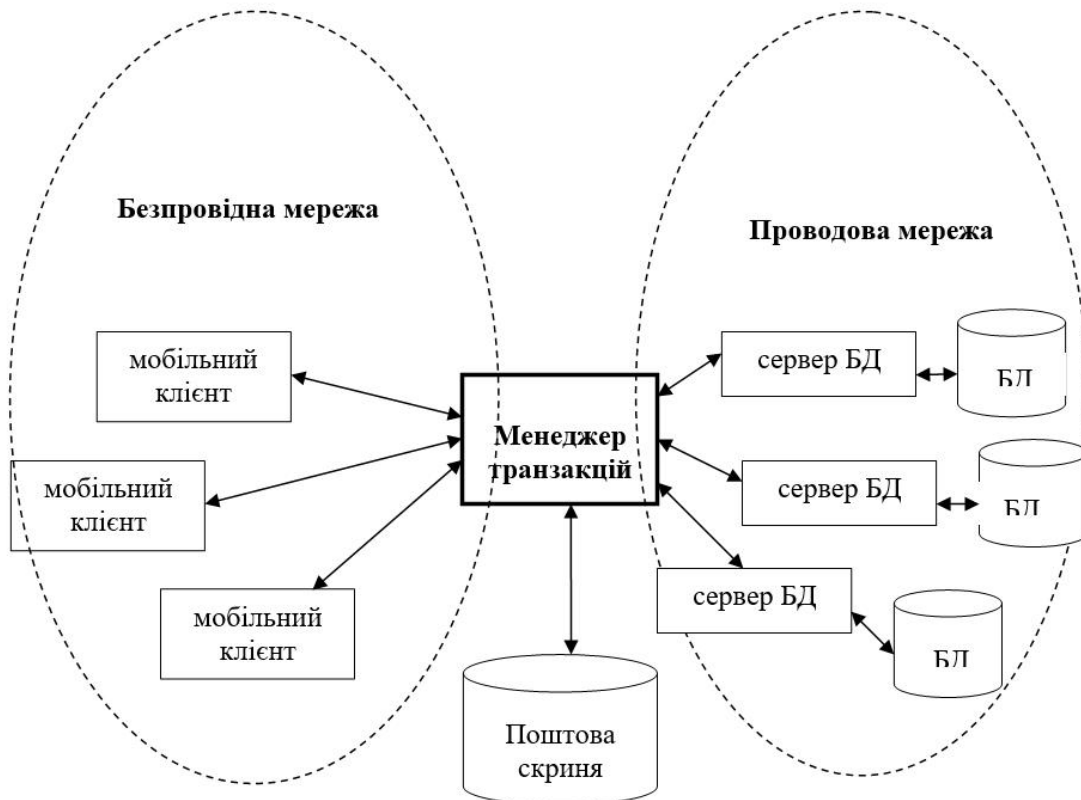


Рис.12.5.. Схема мобільного доступу до віддалених баз даних

Оскільки в якості мобільного пристрою найчастіше використовується сотовий телефон, тому у цій архітектурі системи функціонують наступним чином:

- ✓ запити користувачів надходять від мобільних клієнтів (мікробраузерів) через безпроводну сотову мережу, на сервер доступу до баз даних (Менеджер мобільних транзакцій або просто менеджер транзакцій);
- ✓ менеджер транзакцій виконує запит, звертаючись при необхідності до того або іншого віддаленого серверу бази даних;
- ✓ виконавши запит, менеджер транзакцій формує відповідь клієнту у вигляді набору WML-сторінок і передає їх через безпроводну мережу мобільному клієнту, який пред'являє їх користувачу.

Алгоритм доступу до даних –M-LWDF/EXP.

Алгоритм M-LWDF/EXP (modified largest weighted delay first scheduling) підтримує як звичайні сервіси так і сервіси реального часу. Метою є збереження затримки для кожного трафіку менше заздалегідь визначеного значення з певною ймовірністю. Вимога до затримки може бути задоволена шляхом призначення певних значень параметру. Алгоритм M-LWDF дозволяє отримати оптимальний значення пропускну здатності. Основною характеристикою

даного алгоритму в тому, що рішення при диспетчеризації приймається на підставі як поточного стану каналу зв'язку так і від черг. Хоча важко знайти оптимальне значення параметра? І для кожного типу трафіку, алгоритм M-LWDF може задовольняти вимоги до затримки. Алгоритм M-LWDF заснований на експоненційному законі для забезпечення найкращого обслуговування при використанні звичайних і чутливих до затримки сервісів різними користувачами. Різниця полягає в тому, щоб використовувати різні ваги для різних видів послуг. Функцію пріоритетів можна виразити таким чином:

$$U_i(n) = a_i \exp\left(\frac{a_i D_i(n) - \overline{aD(n)}}{1 + \sqrt{\overline{aD(n)}}}\right) \frac{T_i(n)}{\bar{T}_i} \quad (12.1)$$

де $U_i(n)$ – значення пріоритету для i -го користувача; Кожне значення $U_i(n)$ розраховується за експоненціальним законом, включаючи затримку на обслуговування в черзі. $D_i(n)$ - затримка а обслуговування в черзі для i -го користувача у n -й момент часу; $T_i(n)$ - комфортне значення швидкості для i -го користувача у n -й момент часу; $\bar{T}_i$ - середовищі значення $T_i(n)$; a_i - необхідна ймовірність втрати пакету для користувача i ; - максимальна затримка обслуговування в черзі. $\overline{aD(n)}$ можна розрахувати за формулою:

$$\overline{aD(n)} = \frac{1}{N} \sum a_i D_i(n) \quad (12.2)$$

де N - кількість користувачів використовують потокові програми.

Пропускна здатність вважається найбільш важливим критерієм для сервісів негарантованої доставки (best effort service), а затримка-для поточкових сервісів. При розрахунку з використанням експоненціального закону задовольняються вимоги QoS для сервісів чутливих до затримки.

12.4. Об'єм та швидкість доступу до баз даних

Об'єм бази даних вимірюється : бітами, байтами, кілобайтами, мегабайтами, гігабайтами, терабайтами та петабайтами, табл.12.1.

Таблица.12.1

Об'єм бази даних		
1біт = 0, 1;	1 Мб = 1024 Кб;	1 Пб = 1024 Тб;
1байт = 8 біт;	1 Гб = 1024 Мб;	...
1 Кб = 1024 байт;	1 Тб = 1024 Гб;	...

Об'єм бази даних - відносне число записів описаних об'єктів або документів, доступних для зберігання і обробки, з порівнянням з повним числом реальних об'єктів у зовнішньому середовищі. Об'єм БД ВБД складається з обсягу системних даних ВОСД, обсягу мета даних ВОМД, обсягу даних (записів) ВОД.

$$ВБД = ВОСД + ВОМД + ВОД \quad (12.3)$$

Так як практично визначити обсяг системних даних та обсяг даних (записів)практично неможливо, можливо тільки визначити обсяг мета даних.

Метадані - це обсяг структур, що є в базі даних.

$$V_{meta} = \sum_{table} \sum_{column(type,length)} \quad (12.4)$$

Метадані складаються із суми таблиць та суми колонок(типів, довжин).

Швидкість доступу до БД. Швидкодія запам'ятовуючого пристрою ЗП вимірюється часом записування і зчитування та тривалістю відповідних їм циклів. Час записування t_{WR} -це інтервал між моментами появи керуючого сигналу записування і установленням КП в стан, який задають вхідні сигнали. Час зчитування –це інтервал між моментами появи керуючого

сигналу читання t_{RD} і даних на виході пам'яті. Мінімально доступний інтервал між послідовними читаннями t_{CYR} і записуваннями t_{CYW} створює відповідний цикл. Тривалість циклів може перевищувати час читання чи записування, оскільки після цих операцій необхідна додаткова затримка для встановлення початкового стану пам'яті. Як тривалість циклу звернення до пам'яті, беруть величину $t_{CY} = (t_{CYW}, t_{CYR})$. Середній час доступу до даних Величина P_i , представлена є ймовірності того, що необхідний об'єкт даних знаходиться на рівні i . Для багаторівневої системи пам'яті дотримується умова:

$$P_1 + P_2 + P_3 + \dots + P_n \quad (12.5)$$

Позначимо через t_i загальний доступ до даних на рівні t_i . В цьому випадку середній час доступу:

$$E(t_a) = P_1 E(t_1) + P_2 E(t_2) + P_3 E(t_3) + \dots + P_n E(t_n). \quad (12.6)$$

Або в загальному випадку для системи, пам'яті, який складається із L рівнів:

$$E(t_a) = \sum_{i=1}^L P_i E(t_i) \quad (12.7)$$

У бездротових системах зв'язку різні користувачі x -широкосмугових використовують різні сервіси і потребують різних швидкостей передавання даних. Алгоритм пропорційного обмеження швидкості (PRC) дозволяє збільшити загальну пропускну здатність за рахунок додаткових обмежень на швидкість передачі даних кожного користувача згідно з попередніми налаштуванням системних параметрів $(R_k)_{k=1}^K$. Пропорційне обмеження швидкості може бути виражене за допомогою формули:

$$\frac{R_1}{\beta_1} = \frac{R_2}{\beta_2} = \dots = \frac{R_k}{\beta_k}, \quad (12.8)$$

Швидкість передачі даних R_k може бути виражена за формулою:

$$R_k = \sum_{l=1}^L \frac{\rho_{k,l}}{L} \log_2 \left(1 + \frac{P_{k,l} h_{k,l}^2}{\sigma^2 \frac{B}{L}} \right), \quad (12.9)$$

де k - кількість користувачів; L - кількість піднесучих; $h_{k,l}$ - посилення каналу для k -го користувача піднесучої l ; $P_{k,l}$ - потужність передачі, користувача k в піднесучій l ; σ^2 - СКО спектральної щільності потужності; B - загальна пропускну здатність передатчика; $\rho_{k,l}$ - індикатор, чи використовується піднесуча l k -м користувачем.

12.5. Перспективні принципи зберігання та доступу до інформації в глобальних комп'ютерних системах в базисі Галуа

Проведені дослідження теоретико-числових базисів та їх застосування при організації БД, показують що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадоне захищеним, крім того традиційний метод є незахищеним від несанкціонованого доступу, що достатньо ефективно реалізується в базисі Галуа. Очевидно, що глибокі дослідження застосування теорії і методів кодування даних та організації БД в базисі Галуа є актуальними і перспективними. В той же час дані характеристики БД достатньо ефективно реалізується в базисі Галуа, що визначає перспективу його застосування при організації великооб'ємних багаторівневих БД та баз знань. Проведені дослідження моделей БД показують, що перспективним є розвиток та досконалення нової лінійно-рекурентної моделі БД в базисі Галуа. Запропоновані багаторівневі лінійно-рекурентні моделі БД в базисі Галуа можуть бути ефективно застосовані для компактного завадозахищеного (таємного) кодування даних. Проведений аналіз оцінки ефективності кодування ID та OD на прикладі корпорації „ТЯНЬ ШІ“ показує, що застосування ієрархічно-реляційної організації БД в базисі Галуа є найбільш ефективними та може забезпечити зменшення надлишковості кодування даних, які формуються, зберігаються і передаються каналами зв'язку більше, ніж на один-два порядки. При цьому, враховуючи відносно низьку

вартість засобів зберігання даних на фізичних носіях, які в діапазоні 10^6 - 10^{20} Гбайт, основний ефект зниження собівартості руху даних в транснаціональних міжнародних корпораціях можуть досягти в каналах зв'язку локальні та глобальні комп'ютерні мережі та Інтернет, що при інтенсивному трафіку обміну даними 0 – 1 Гбайт/с може скласти сотні тисяч у.о. на рік. Перевага вертикально-рекурентної моделі БД в базисі Галуа ефективніша в 14,42 рази ніж кодування ідентифікаційних даних унітарного дерева СЕС корпорації „Pallada Travell” USA. Запропоновані вертикально-рекурентні моделі БД в базисі Галуа соціально-економічного структурування. можуть бути ефективно застосовані для компактного завадозахищеного (таємного) кодування даних, наприклад, в міжнародних транснаціональних корпораціях глобальних розподілених комп'ютерних систем та каналах космічному зв'язку. Проведені дослідження розроблених моделей показують, що перспективним є розвиток та досконалення нового синтезу конгломератів епохальних моделей сховищ даних в базисі Галуа можуть бути застосовані у глобальних розподілених комп'ютерних системах

Визначені систематичні характеристики даних, які включають час зберігання даних та їх об'єм, швидкість запиту та зчитування, а також захист від помилок і несанкціонованого доступу, що дозволяє конкретизувати місце моделей теоретико-графових та теоретико-множинних баз даних серед інших моделей.

Описані існуючі типи баз даних та рівні їх проектування, що дозволяє визначити перспективи удосконалення та підвищення ефективності баз даних глобальних комп'ютерних систем.

Виконана класифікація баз даних, яка дозволила встановити взаємозв'язок між розподіленими базами даних і надвеликими комп'ютерними базами даних.

Досліджені архітектурні характеристики моделей баз даних які підтверджують перспективний розвиток об'єктно-орієнтованих та об'єктно-реляційних баз даних. виконана оцінка коефіцієнта ефективності розподілених баз даних яка враховує пропускну здатність інформаційних каналів та швидкість модемів комунікаційних центрів та абонентів.

Обґрунтовані перспективи зберігання даних у конгломератів епохальних моделей сховищ даних в базисі Галуа, які можуть бути застосовані у глобальних розподілених комп'ютерних системах.

РОЗДІЛ 13

БАЗИ ДАНИХ В КОДАХ ПОЛЯ ГАЛУА

13.1.Методи кодування елементів ієрархічних моделей БД в ТЧБ

В даний час в теорії та техніці цифрової обробки даних спостерігається ефективно використання теоретико-числових базисів (ТЧБ). Аналіз використання теоретико-числових базисів в інформаційних технологіях показує виключно широке застосування базису Радемахера, який породжує двійкові кодові системи числення. В той же час світовий досвід застосування ТЧБ для швидкодіючих теоретико-числових перетворень, які реалізують числові згортки та цифрову обробку сигналів, показує що значне прискорення швидкості процесорів та упрощення програмних алгоритмів забезпечують базиси Крестенсона та Галуа.

ТЧБ, які використовуються для кодування даних в БД

Теоретико-числові базиси – є фундаментальними теоретичними основами систем числення та методів кодування даних. (ТЧБ) – породжується системами ортогональних функцій (наприклад: базис, функцій Фур'є, поліноми Лагранжа, Чебишева, Беселя , функції Якобі, та інші) .Класифікація охоплює наступні вісім ТЧБ : Унітарний , Хаара , Крейга, Уолша , Радемахера , Грея , Крестенсона, Галуа.

Застосування теоретико–числових базисів при організації БД

Очевидно, що застосування базисів з високою надлишковістю кодових матриць (наприклад: унітарної, Хаара, Крейга,) є неефективними при кодування інформації в базах даних. Тому аналіз ефективності застосування ТЧБ в базах даних виконаємо на основі базисів Радемахера, Крестенсона і Галуа.

Аналіз застосування ТЧБ Радемахера в ієрархічних базах даних.

При використанні базису Радемахера для кодування даних БД ієрархічної архітектури число ідентифікованих елементів у загальному випадку описуються формулою:

$$m = n \cdot \hat{E}[\log_2 P] = \hat{E}[\log_2 N] , \quad (13.1)$$

де :P - основа (модуль відповідної позиційної системи числення);

$\hat{E}[\cdot]$ - цілочисельна функція з округленням до більшого цілого; N – число ідентифікованих елементів баз даних; n - число розрядів коду у системі числення з модулем P; m – число двійкових розрядів ідентифікатора і-того елементу БД з діапазону $1 \leq i \leq N$.

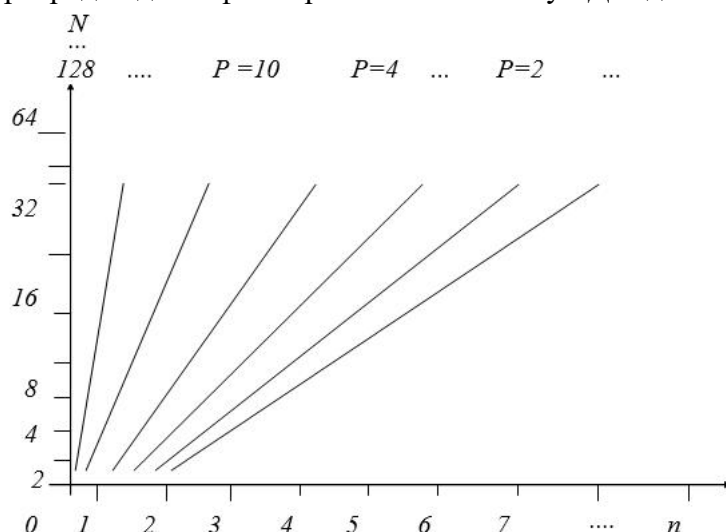


Рис.13.1. Графік числа елементів БД в залежності від параметрів n та P-позиційних систем числення

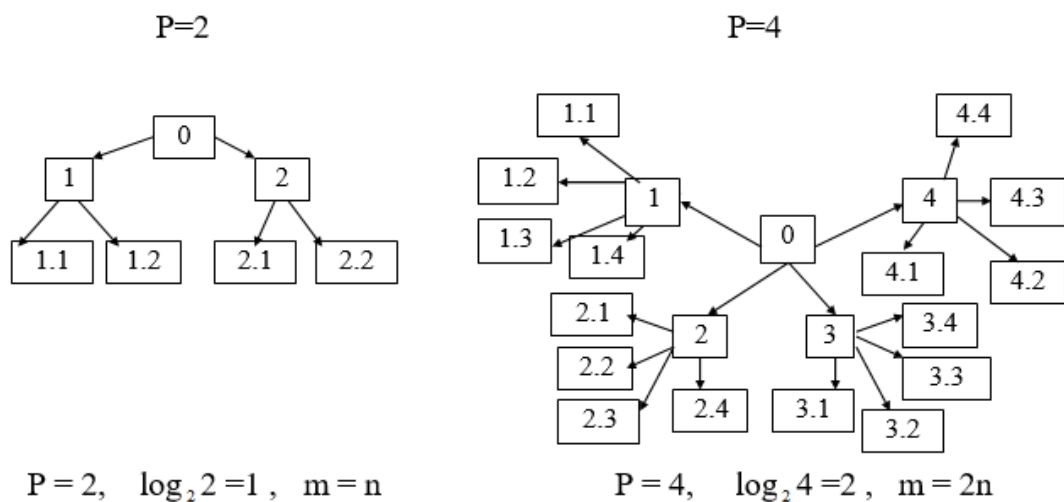
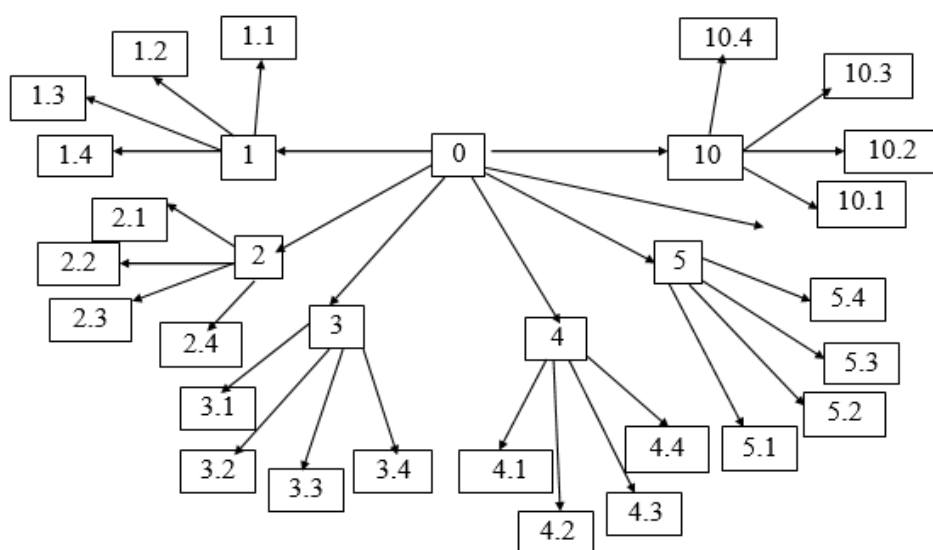


Рис.13.2. Архітектура ієрархічної БД при модулях розширення $P=2, 4$.



$P = 10, \log_2 10 = 4, m = 4n$

якщо $N = 100$, то $m = 7n$

Рис. 13.3. Архітектура ієрархічної БД при модулі розширення $P=10$.

Аналіз кодування даних БД в різних традиційних системах числення та застосування базису Радемахера ефективно при бінарному кодуванні ($P=2$). Для інших систем числення коли $P \neq 2^k$, $k = 0, 1, 2, \dots$, кодування даних БД в базисі Радемахера є надлишковим.

ТЧБ Крестенсона

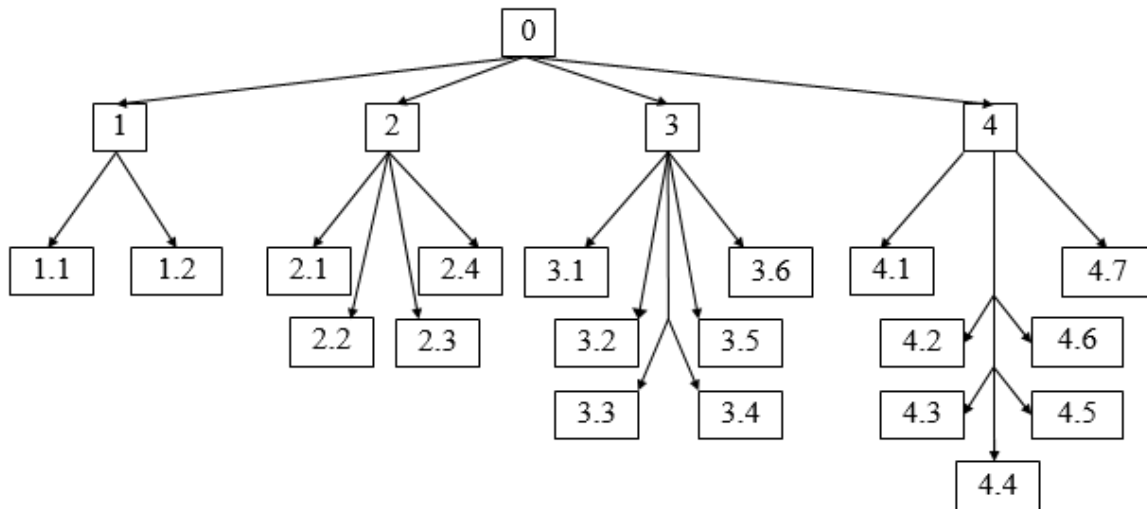
При використанні базису Крестенсона реалізується оптимальне кодування ієрархічної БД автономно на кожному рівні в системі взаємнопростих модулів:

$$P^1, P^2, P^3, \dots, P^i, \dots, P^k, \dots, \text{ тобто } P^i \neq P^j. \quad (13.2)$$

Наприклад в системі базису Крестенсона виберемо модулі

$$P^1 = 3, P^2 = 5, P^3 = 7, P^4 = 8, \quad (13.3)$$

що відповідають ієрархічній архітектурі БД представлений на рис.13.4.



$P_1=2$, 2 біти; $P_2=4$, біти; $P_3=6$, 3 біти; $P_4=7$, 3 біти.

Рис.13.4. Ієрархічна архітектура БД в базисі Крестенсона

При застосуванні СЗК досягається оптимальне кодування елементів БД у вигляді коду N^k , який описується формулою:

$$N^k = \text{res} \sum_{i=1}^k b^i V^i \pmod{P}, \quad (13.4)$$

де: res – символ операції обчислення найменшого невід’ємного залишку;

b^i - код елементу БД ($0 \leq b^i \leq P^{i-1}$);

V^i - базисні числа СЗК;

$P = \prod_{i=1}^k P_i$ - узагальнений модуль СЗК;

При цьому код елементів БД в СЗК для заданої оптимальної системи модулів P^i записується у вигляді послідовності залишків.

$$N^k = (b^1, b^2, b^3, \dots, b^i, \dots, b^k). \quad (13.5)$$

Кодування елементів БД в базисі Крестенсона за рахунок оптимального вибору системи модулів P^i забезпечує більш компактний запис кодів:

$$0 \leq N^k \leq P-1,$$

які мають розрядність

$$m = \hat{E}[\log_2(P-1)], \quad (13.6)$$

Визначаємо $P = 2 \cdot 4 \cdot 6 \cdot 7 = 336$, звідки

$$m = \hat{E}[\log_2(336-1)] = 11 \text{ біт} \quad (13.7)$$

В даному випадку кодування елементів в базисі Крестенсона дає оцінку 11 біт, що на 12% менше порівняно з базисом Радемахера.

Базис Крестенсона характеризується можливостями захисту кодів залишкових класів від помилок шляхом розширення системи модулів P^i додатковим модулем P^0 по якому $b^0 = 0$, що забезпечує виявлення помилок в кодових елементів БД. Розширення системи модулів двома

модулями P^k, P^0, P^{k+1} , з відповідними залишками $b^0 = 0$, b^{k+1} , дозволяє реалізувати алгоритми виправлення помилок в кодах СЗК.

Але недоліком викладеного методу кодування елементів БД є нелінійність кодів N^k , які без декодування не дозволяють визначити значення b^i . В той же час дана властивість може бути ефективно використана для шифрування елементів БД шляхом вибору невідомого набору модулів P^i .

ТЧБ Галуа.

При застосуванні базису Галуа реалізується рекурентний метод кодування елементів ієрархічних БД, що представлено архітектурою на рис.13.5.

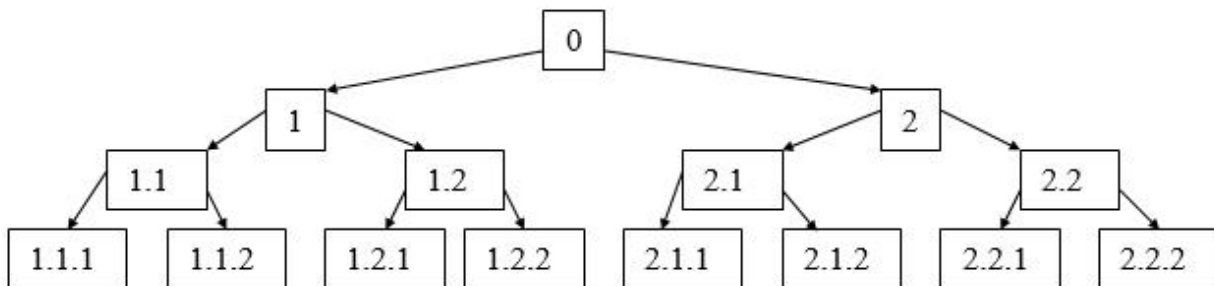


Рис.13.5. Ієрархічна БД в базисі Радемахера

При цьому для кодування елементів ієрархічної БД в базисі Радемахера або Крестенсона необхідно дані кодувати згідно з кодовою матрицею.

Очевидно що в даній кодовій матриці базису Радемахера однозначно відповідає матриця-вектор базису Галуа.

Таким чином ієрархічна архітектура БД в базисі Галуа може бути представлена новою лінійно - рекурентною архітектурою (рис.13.6).

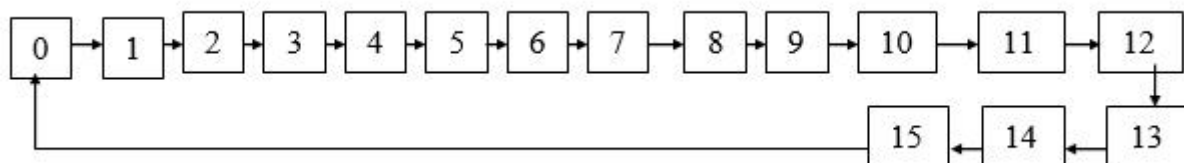


Рис.13.6 Лінійно - рекурентна архітектура БД в базисі Галуа.

Властивість рекурентності зв'язків ієрархічної БД при кодуванні елементів її в базисі Галуа дозволяє по новому розглянути характеристики організації таких БД.

Порівняння ефективності кодування елементів баз даних в різних базисах

Нехай маємо ієрархічну БД з бінарним кодуванням елементів на різних рівнях: $N = 5000$, $n = 4$. Необхідно виконати адресацію 50 елементів згрупованих в три групи : 30, 16, 4.

Виходячи з максимального числа елементів БД визначимо розрядність коду індифікатора елементів в базисі Галуа.

$$m = \hat{E}[\log^2 5000] = 13 \text{ біт.} \quad (13.8)$$

Таким чином об'єм кодової інформації для індексації всіх груп елементів Д описується наступним чином:

$$mG = (3 \cdot 13) + 30 + 16 + 4 = 89 \text{ біт} \quad (13.9)$$

Аналогічно кодова розрядність ідентифікатора в базисі Радемахера визначається наступним чином :

$$mR = 13 \cdot 50 = 650 \text{ біт,} \quad (13.10)$$

звідки коефіцієнт стиснення ідентифікатора елементів БД в базисі Галуа

$$K_c = \frac{mR}{mG} = \frac{650}{89} = 7,303. \quad (13.11)$$

Якщо в БД елементи тісно пов'язані між собою і не перевищує $n = 100$, то ефективність кодування БД в базисі Галуа можна підвищити наступним чином, шляхом кодування числа елементів в групі після коду ідентифікатора, тобто

$$mGn = 3(13 + \hat{E}[\log^2 100]) = 3(13 + 7) = 60 \text{ біт}. \quad (13.12)$$

Звідки коефіцієнт стиснення об'єму даних при переході в базис Галуа по відношенні до базису Радемахера дорівнює:

$$K_c = \frac{mR}{mRn} = \frac{650}{60} = 10,833. \quad (13.13)$$

13.2. Ієрархічно–рекурентна модель баз даних у теоретико-числовому базисі Галуа

В даний час в теорії та техніці цифрової обробки даних спостерігається ефективне використання теоретико-числових базисів (ТЧБ) для швидкодіючих теоретико-числових перетворень, які реалізують числові згортки та цифрову обробку сигналів. Аналіз використання технологій має виключно широке застосування ТЧБ Радемахера, який породжує двійкові кодові системи числення. Значне прискорення швидкості процесорів та спрощення програмних алгоритмів забезпечують ТЧБ Крестенсона та ТЧБ Галуа.

Проведені дослідження і порівняння матриці ТЧБ Радемахера з матрицею–вектором ТЧБ Галуа та їх застосування при організації баз даних (БД) та знань, показують що традиційний спосіб ідентифікації та кодування елементів БД в ТЧБ Радемахера є неефективним, надлишковим і завадонебезпечним, крім того традиційний метод є незахищеним від несанкціонованого доступу, що достатньо ефективно реалізується в ТЧБ Галуа. Очевидно, що глибокі дослідження застосування теорії і методів кодування даних та організації БД в ТЧБ Галуа є актуальними і перспективними.

Логічна структура ієрархічно–рекурентної моделі БД у ТЧБ Галуа.

При застосуванні ТЧБ Галуа реалізується рекурентний метод кодування елементів ієрархічних БД, що представлено архітектурою на рисунку 13.7. ТЧБ Галуа має кодову матрицю, яка перетворюється у вектор, кодування даних в ТЧБ Галуа є рекурентним, і описується теорією полів Галуа. Таким чином ТЧБ Галуа є найбільш компактним з відомих ТЧБ. Кодова матриця-вектор ТЧБ Галуа $-V^{Gal}$, де $m=1$ – число стовпців; N – однобітових рядків; об'єм матриці $V=N$; число активних елементів: $n=N/2$.

При цьому для кодування елементів ієрархічної моделі БД в ТЧБ Радемахера або Крестенсона необхідно дані кодувати згідно з кодовою матрицею.

Очевидно що в даній кодовій матриці ТЧБ Радемахера однозначно відповідає матриця-вектор ТЧБ Галуа. Тому, на основі відповідності можна зробити порівняльну характеристику ТЧБ Радемахера з ТЧБ Галуа.

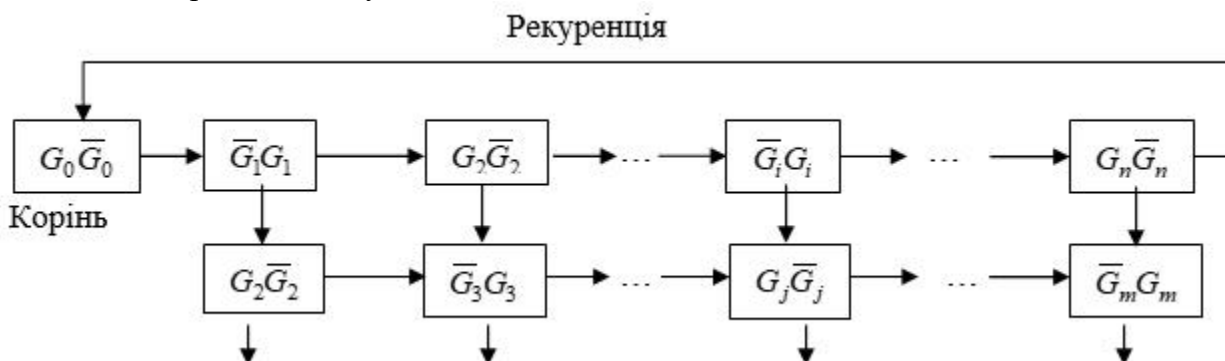


Рис.13.7. Ієрархічно-рекурентна модель БД у базисі Галуа

Порівняльна характеристика матриці Радемахера з вектором Галуа

ТЧБ Галуа породжує систему числення Галуа. Наприклад при $m=4$; $N=2^4=16$ кодовий вектор базису Галуа описується двійковою послідовністю 11110101110010000. Для кодування чисел m в ТЧБ Радемахера в діапазоні 0- N необхідно $\log^2 N$ розрядів двійкових чисел. При кодуванні ТЧБ Галуа на кожен N^i число припадає 1біт Галуа. А для кодування чисел N з вектору необхідні рекурентно вибрати m -біт даних що засвідчує максимальну упаковку інформації в ТЧБ Галуа. Властивість рекурентності зв'язків ієрархічної моделі БД при кодуванні елементів її в ТЧБ Галуа дозволяє по новому розглянути характеристики організації таких БД.

Порівняння ефективності кодування елементів БД в різних ТЧБ. Визначимо розрядність коду ідентифікатора елементів в ТЧБ Галуа, де:

$m = \sum_{i=1}^N [\log^2 5000]$, об'єм кодової інформації $mG = 122$ біт, кодова розрядність ідентифікатора в базисі Радемахера $mR = 1079$ біт, коефіцієнт стиснення ідентифікатора

елементів БД в базисі Галуа $K_c = \frac{mR}{mG} = \frac{1079}{122} = 8,84$, кодування числа елементів в групі після коду ідентифікатора, тобто $mGn = 60$ біт. Звідки коефіцієнт стиснення об'єму даних при переході

в базис Галуа по відношенні до базису Радемахера дорівнює: $K_s = \frac{mR}{mGn} = \frac{1079}{60} = 17,98$ [1,2,3].

Результати досліджень показали на основі проведених досліджень ієрархічної бази даних, перевагу ТЧБ Галуа ефективнішого у 17,98 рази над кодуванням ідентифікаційних даних ТЧБ Радемахера. Побудована логічна структура нової ієрархічно-рекурентної моделі БД у ТЧБ Галуа.

13.3. Ієрархічно-реляційна модель баз даних у ТЧБ Галуа

Ієрархічні системи, в яких базова структура даних мала деревоподібний вигляд, досягли найвищої ефективності функціонування, але виразові можливості цих систем лишилися відносно низькими. Системам зі структурами даних типу мережі, навпаки. Вдалося надати значно кращих виразових можливостей, але вони програють в ефективності функціонування, тобто для плідної експлуатації таких систем від користувача вимагається значно вищий рівень кваліфікації. Останніми з'явилися СКБД реляційного типу, які характеризуються найпростішою структурою даних, таблицею, або плоским файлом з одного боку та високим рівнем мов маніпулювання даними – з іншого боку. Це зробило їхні виразові можливості максимально потужними, але знизило ефективність функціонування.

Тому об'єднання ієрархічної моделі БД та реляційної моделі БД в одну гібридну ієрархічно-реляційну модель БД в ТЧБ Галуа, дає можливості подолати недоліки та підвищити виразові можливості максимально потужними, досягти найвищої ефективності функціонування з високим рівнем мов маніпулювання даними.

Логічна структура ієрархічно-реляційної моделі БД у ТЧБ Галуа

Схема бази даних визначає спосіб зберігання даних. У нашому випадку необхідно створити гібридну схему деревоподібного вигляду і таблиці або плоскі файли для зберігання даних, що використовуються в ієрархічно-реляційних БД в ТЧБ Галуа, показаних у табл. 13.1. та на рис.13.8.



Рисунок.13.8. Ієрархічно-реляційна база даних в базисі Галуа

Таблиця 13.1.

Середньостатистичні параметри даних ієрархічно –реляційної БД п'ятирівневої системи БРМ

НП	КД	ИОД	НСД	ИО СД	ТЕ	ОН	З	ОМЗ	СО	ЛНО	НОС
6·X	8·X	24·X	8·X	24·X	1·X	2·X	3·X	5·X	5·X	9·X	12·X
...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...
6·X	8·X	24·X	8·X	24·X	1·X	2·X	3·X	5·X	5·X	9·X	12·X

Де: X – байт, ІД – ідентифікаційні дані дистриб'ютора , ОД – організаційні дані дистриб'ютора

В табл.13.1. символ $n \cdot X$ - означає число алфавітно-цифрових даних в кожній позиції двомірного файлу ($n = 1, 2, \dots$). Таким чином при заданій кількості користувачів згідно табл.2 можна розрахувати об'єм алфавітно-цифрових даних одного кортежа БД наприклад, дистриб'юторських організацій транснаціональних корпорацій.

Для зберігання даних кортежа реляційної БД, виходячи з умови байт-орієнтованого кодування ,загальний об'єм даних може бути обчислений згідно формули :

$$I_j = \sum_{j=1}^m \sum_{i=1}^k d_{ij} , \quad (13.14)$$

де d^{ij} - об'єм даних ij – го атрибута кортежа ($i=1, 2, \dots, k$) – число атрибутів, ($j = 1, 2, \dots, m$) - число кортежів.

Аналіз табл.2 показує, що об'єми оперативних даних (ОД) діяльності дистриб'юторів визначаються доменами : ЛЗ, ОМЗ, СО, ЛНО та ОНОС. , а ідентифікаційні дані описуються доменами: НП, НКД, ФИОД, НСД, ФИОСД, СТЕ та ПОН. Таким чином ефективність кодування БД ієрархічно-реляційної архітектури, можна розрахувати за формулами відносної та приведеної оцінки:

$$K_{e1} = \frac{I_{ID}}{I_{OD}} ; K_{e2} = \frac{I_{ID} - I_{OD}}{I_{OD} + I_{ID}} \cdot 100\%. \quad (13.15)$$

Ефективність кодування БД ієрархічно-реляційної архітектури, яка відповідає корпорації , можна розрахувати за формулою (13.16):

$$I^{OD} = 34 \cdot X;$$

$$I^{ID} = 73 \cdot X. \quad (13.16)$$

Підставляючи дані у формулу (13.15), отримуємо значення коефіцієнта ефективності у вигляді:

$$K^{e_1} = 2,147; K^{e_2} = 36,4 \% \quad (13.17)$$

що відповідає більш ніж двохкратній надлишковості кодування (ІД) по відношенню до (ОД) або 36,4 % відносній надлишковості кодування (ІД).

Практично у корпорації використовується універсальна система кодування ІД у вигляді двох 8 – розрядних десяткових кодів, які ідентифікують код наставника і дистриб'ютора в будь-якій структурі і підрозділі в будь-якій країні світу на ринку яких працює корпорація. Це складає 64 біт двійково-десяткового коду, або символно-байтового стандартного коду. При цьому об'єм бази ідентифікаційних даних менеджерів нижчого рівня корпорації складає $11200 \times 128 = 1433600$, а загальна кількість можливих ідентифікованих членів складає 2^{64} осіб дистриб'юторів і споживачів населення планети Земля. Для визначення певної ланки (наприклад дистриб'юторської організації №1) структури від лідера до дистриб'ютора і заявника відслідковуємо шлях: 0 – 1 – 2 – 3 – 1 – 2 – ... тобто лідер – дистриб'ютор підписує в свою групу 1 – 4 – .. нових заявників, які дублюють своїх наставників. Таким чином, наприклад, відслідковуючи шлях присвоюємо номер дистриб'ютору 1 – рівня – код 01, 2 – рівня – код 012, 3 – рівня – код 0123, 4 – рівня – код 01231, 5 – рівня – код 012312, новому заявнику – код 0123124. Відповідно з десяткової системи числення переходимо на кодування даних в двійковій системі числення в базисі Галуа. Якщо взяти окрему структуру дистриб'юторської організації (вузла k_1) і починаючи з лідера і $1^* 0$ -рівня та пройти шлях до $8^* 5$ - рівня зафіксувавши шлях та згідно порядку підписання контракту та розташуванні на певному рівні у групах буде присвоєний відповідний номер, який можна закодувати в базисі Галуа. Відкриваючи код ієрархічної структури по його номеру, відкривається реляційна БД всієї структури кожного дистриб'ютора.

Проведені дослідження теоретико-числових базисів та їх застосування при організації БД, показують що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним надлишковим і завадоне захищеним, крім того традиційний метод є незахищеним від несанкціонованого доступу, що достатньо ефективно реалізується в базисі Галуа.

На основі ієрархічно-рекурентної моделі БД у ТЧБ Галуа можна побудувати нову лінійно-рекурентну модель БД, нову вертикально-рекурентну модель БД та нову ієрархічно-реляційну модель БД, вони будуть основою для побудови інших баз даних у ТЧБ Галуа і які можуть широко застосовуватися для освіти і науки України.

Очевидно, що глибокі дослідження застосування теорії і методів кодування даних та організації БД в базисі Галуа є актуальними і перспективними.

В 21 столітті корпорацією планується реалізація 80% роздрібною торгівлі в мережевому маркетингу і 20% в класичній роздрібній торгівлі. Завдяки мережевому п'ятирівневому маркетингу тетраїдних груп дистриб'юторських організацій компанії вперше пропонувано ієрархічно-рекурентну організацію БД в базисі Галуа.

Проведений аналіз оцінки ефективності кодування ІД та ОД на прикладі корпорації „ТЯНЬ ШІ“ показує що застосування ієрархічно-рекурентної організації БД в базисі Галуа є найбільш ефективними та може забезпечити зменшення надлишковості кодування даних, які формуються, зберігаються і передаються по міжнародних каналах зв'язку більше на один два порядки. При цьому враховуючи відносно низьку вартість засобів зберігання даних на фізичних носіях, які в діапазоні $10^6 - 10^{20}$ Гбайт основний ефект зниження собівартості руху даних в транснаціональних міжнародних корпораціях можуть досягти в каналах зв'язку локальні та глобальні комп'ютерні мережі та Інтернет, що при інтенсивному трафіку обміну даними 0 – 1 Гбайт/с може скласти сотні тисяч у.о. на рік.

РОЗДІЛ 14

ОРГАНІЗАЦІЯ ЛІНІЙНО-РЕКУРЕНТНИХ МОДЕЛЕЙ БАЗ ДАНИХ НА ОСНОВІ ТЧБ ГАЛУА

14.1. Однорівнева лінійно-рекурентна модель БД у ТЧБ Галуа.

В роботі викладені принципи побудови лінійно-рекурентної моделі бази даних в базисі Галуа і проведений аналіз ефективності кодування ідентифікаційних даних на прикладі ієрархічно-реляційної бази даних міжнародних транснаціональних корпорацій в глобальних розподілених комп'ютерних системах

Одним із вирішальних факторів забезпечення компактного та ефективного зберігання корпоративної інформації є вдосконалення сучасних технологій побудова баз даних. Одним із перспективних напрямків теорії організації БД є реалізація доступу до даних на основі нової лінійно-рекурентної моделі бази даних в базисі Галуа.

Лінійно-рекурентна модель БД є новим типом структурної організації баз даних, яка формується на основі рекурентних властивостей теоретико-числового базису Галуа. Важливою характеристикою таких БД є максимальна компактність адресації даних, в тому числі при багаторівневій архітектурі. Позитивною характеристикою лінійно-рекурентних БД є можливість ефективного захисту від помилок та несанкціонованого доступу.

При цьому особливістю використання базису Галуа при організації БД є можливість не тільки компактного кодування даних але і організований захист даних від помилок та несанкціонованого доступу. Тому дослідження ефективності лінійно-рекурентної організації БД є актуальною науково-технічною задачею.

При проектуванні бази даних, насамперед, вирішується дві задачі: організація структури всієї бази даних, що визначає типи зв'язків між даними, і організація структури самих даних. Логічна структура бази даних також називається її логічним представленням і може бути таблицею, деревом чи мережею. Крім логічного існує також фізичне представлення бази даних – це спосіб розміщення її пам'яті комп'ютера, як правило, на твердому магнітному диску. Головне при цьому – забезпечити ті зв'язки, що відбиті в логічній структурі бази даних. Програми що дозволяють забезпечити зв'язок між фізичними даними і їхнім логічним описом, а також обробку даних, називається системою керування базою даних[1,2].

Логічна структура ієрархічної моделі бази даних в базисі Радемахера

При застосуванні базису Радемахера реалізується метод кодування елементів ієрархічних моделей БД, що логічно представлена на рис.14.1

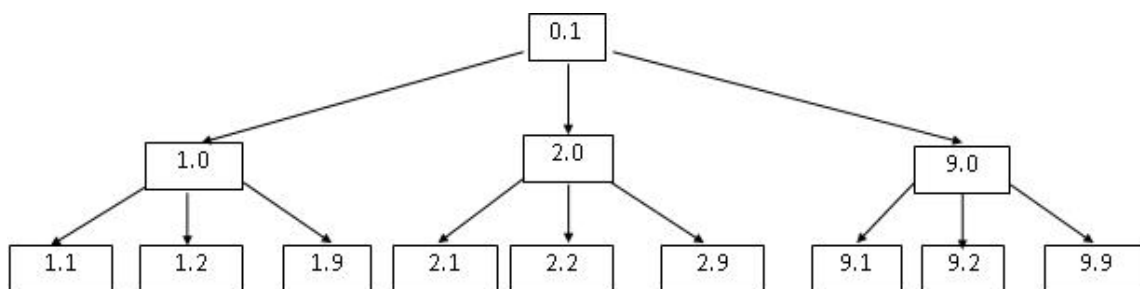


Рис.14.1. Ієрархічна модель БД в базисі Радемахера.

При цьому для кодування елементів ієрархічної моделі БД в базисі Радемахера або Крестенсона необхідно дані кодувати згідно з кодовою матрицею між базисного переходу. Очевидно, що даній кодовій матриці базису Радемахера однозначно відповідає матриця-вектор базису Галуа. Ієрархічна архітектура БД в базисі Галуа може бути представлена новою лінійно - рекурентною архітектурою БД (Рис. 10.2).

Логічна структура циклів однорівневої лінійно-рекурентної моделі БД в ТЧБ Галуа

Новим типом структурної організації баз даних, яка формується на основі рекурентних властивостей теоретико-числового базису Галуа. Важливою характеристикою таких БД є максимальна компактність адресації даних, в тому числі при багаторівневій архітектурі.

На основі петлі, однорівневої, дворівневої та багаторівневої схеми з циклами - категорії схем [1] та горизонтального вектора базису Галуа [2] можна побудувати лінійні цикли з ланцюгами нової лінійно-рекурентної моделі БД в базисі Галуа.

Позитивною характеристикою лінійно-рекурентних моделей БД, є можливість ефективного захисту від помилок та несанкціонованого доступу. При застосуванні базису Галуа реалізується рекурентний метод кодування елементів ієрархічних БД. Ієрархічна модель БД в базисі Галуа може бути логічно представлена новою лінійно - рекурентною моделлю БД в базисі Галуа без відвітвлень (рис. 14.2).

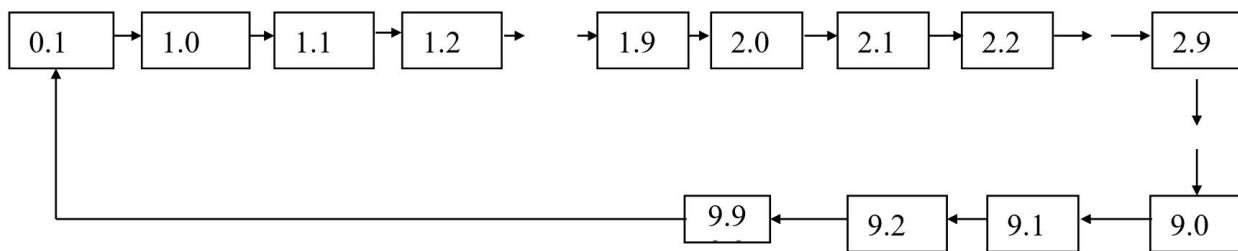


Рис.14.2. Цикли однорівневої лінійно-рекурентної моделі БД в базисі Галуа.

Властивість рекурентності зв'язків ієрархічної БД при кодуванні елементів її в базисі Галуа дозволяє по новому розглянути характеристики організації таких БД.

Для БД, показаних на рис.14.1, рис.14.2, визначаємо коефіцієнт стиснення:

$$K_c = \frac{mR}{mG} = \frac{800}{100} = 8, \quad (14.1)$$

де $mR = 100 \cdot 8 \text{ біт} = 800 \text{ біт}$,

$mG = 100 \text{ біт}$.

14.2. Дворівнева лінійно-рекурентна модель БД у ТЧБ Галуа

Логічна структура циклів дворівневої лінійно-рекурентної моделі бази даних в базисі Галуа. На основі циклів однорівневої лінійно-рекурентної моделі БД у базисі Галуа може бути побудована дворівнева лінійно-рекурентна модель БД, яка відповідає ієрархічній структурі з десятковими елементами розширення та розгалуження (рис.14.3).

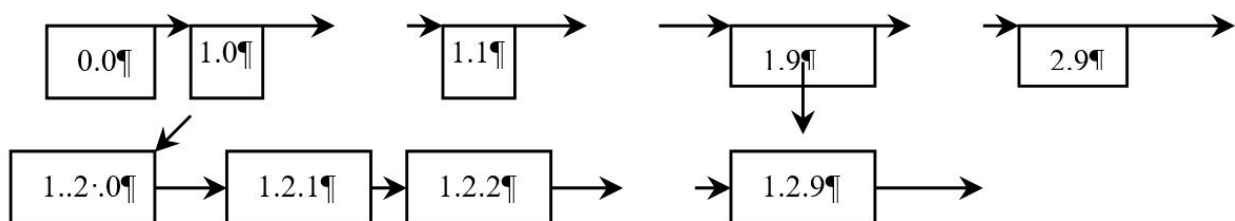


Рис.14.3. Цикли дворівневої лінійно-рекурентна модель БД в базисі Галуа.

14.3. Багаторівнева лінійно-рекурентна модель БД у ТЧБ Галуа

Логічна структура багаторівневої лінійно-рекурентної моделі БД в ТЧБ Галуа

На основі лінійно-рекурентної моделі БД у базисі Галуа може бути побудована багаторівнева лінійно-рекурентна модель БД, яка відповідає ієрархічній структурі з десятковими елементами розширення та розгалуження (рис.13.4).

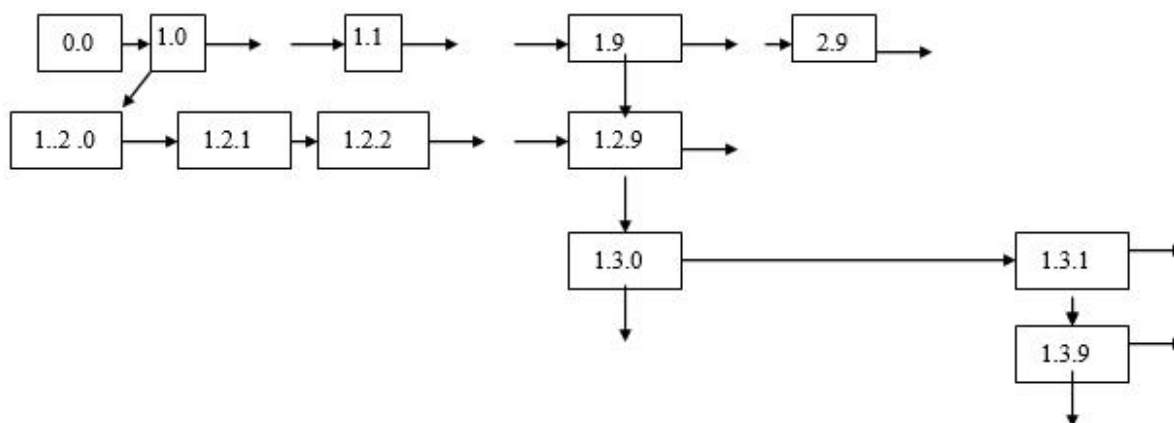


Рис.14.4.Цикли багаторівневої лінійно-рекурентної моделі БД в базисі Галуа

Фізична структура багаторівневої лінійно-рекурентної моделі БД в ТЧБ Галуа

Проведемо розрахунок порівняння ефективності кодування елементів БД у різних базисах. Нехай маємо ієрархічну БД з бінарним кодуванням елементів на різних рівнях: $N = 5000$, $n = 4$. Необхідно виконати адресацію 50 елементів згрупованих в три групи: $p_1 = 30$, $p_2 = 16$, $p_3 = 4$. Виходячи з максимального числа елементів БД визначимо розрядність коду індифікатора елементів в базисі Галуа.

$$m = n \cdot \hat{E}[\log_2 P] = \hat{E}[\log_2 N] = \hat{E}[\log_2 5000] = 13 \text{ біт.} \quad (14.2)$$

Таким чином об'єм кодової інформації для індифікації всіх груп елементів Д описується наступним чином:

$$mG = 3 \cdot m + (p_1 + p_2 + p_3) = (3 \cdot 13) + 30 + 16 + 4 = 89 \text{ біт} \quad (14.3)$$

Аналогічно кодова розрядність індифікатора в базисі Радемахера визначається наступним чином:

$$mR = mP = 13 \cdot 50 = 650 \text{ біт,} \quad (14.4)$$

звідки коефіцієнт стиснення ідентифікатора елементів БД в базисі Галуа

$$K_c = \frac{mR}{mG} = \frac{650}{89} = 7,303. \quad (14.5)$$

Якщо в БД елементи тісно пов'язані між собою і не перевищує $n = 100$, то ефективність кодування БД в базисі Галуа можна підвищити наступним чином, шляхом кодування числа елементів в групі після коду ідентифікатора, тобто

$$mGn = 3(13 + \hat{E}[\log_2 100]) = 3(13 + 7) = 60 \text{ біт.} \quad (14.6)$$

Звідки коефіцієнт стиснення об'єму даних при переході в базис Галуа по відношенні до базису Радемахера дорівнює:

$$K_c = \frac{mR}{mGn} = \frac{650}{60} = 10,84. \quad (14.7)$$

Результати моделювання

Методи оптимізації та підвищення ефективності БД

Основними характеристиками оптимізації і підвищення ефективності є: -здешевлення системи СУБД; -збільшення швидкості пошуку та доступу до даних; -зменшення тастиснення об'єму інформації; -розробка різноманітних типів пристроїв зберігання даних; -тенденції росту системи; -підвищення продуктивності системи; -паралельні обчислення; -розвиток лінійно-рекурентної моделі БД; -підвищення надійності захисту від помилок; -захист від несанкціонованого доступу.

Критерії ефективності БД.

Важливими характеристиками ефективності БД є: емерджентність, надлишковість, швидкість пошуку даних.

Фактори які впливають на підвищення ефективності БД

Основними характеристиками оптимізації і підвищення ефективності є:

- здешевлення системи СУБД ;
- збільшення швидкості пошуку доступу до даних;
- зменшення або стиснення об'єму інформації ;
- розробка різноманітних типів третинних пристроїв зберігання (tertiary storage devices)

даних;

- тенденції росту системи ;
- підвищення продуктивності системи ;
- паралельні обчислення;
- розвиток системи клієнт/сервер і багаторівневої архітектури;
- підвищення надійності захисту від помилок;
- захист від несанкціонованог доступу до БД тощо.

Категорії нових схем, однорівневі, дворівневі та багаторівневі лінійно–рекурентні моделі БД в базисі Галуа, які запропоновані для мереж структур

БД,здатні моделювати в повному обсязі ієрархічну структуру та власне лінійно–рекурентні схеми. Окрім того, у загальному випадку вони дають можливість описувати петлі та цикли, зображені у табл.14.1.

Таблиця.14.1.

Категорії схем лінійно–рекурентних моделей баз даних в базисі Галуа

№ п/п	Схеми Структури	Петля з циклом	Однорівнева схема з циклами,ланцюгами і рекурентією	Дворівнева схема з циклами, ланцюгами і рекурентією		Багаторівнева схема з циклами, ланцюгами і рекурентією	
			без відвітлення	без відвітлення	з відвітленням	без відвітлення	з відвітленням
1	Лінійно-рекурентні моделі бази даних в базисі Галуа						

Проведені дослідження моделей БД показують, що перспективним є розвиток та досконалення нової лінійно-рекурентної моделі БД в базисі Галуа.

Дослідження теоретико–числових базисів та їх застосування при організації БД, показують що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадоне захищеним. Крім того, традиційний метод є незахищеним від несанкціонованого доступу. В той же час дані характеристики БД достатньо ефективно реалізується в базисі Галуа.

Очевидно, що глибокі дослідження застосування теорії і методів кодування даних та організації БД в базисі Галуа є актуальними і перспективними.

Запропоновані багаторівневі лінійно-рекурентні моделі БД в базисі Галуа можуть бути ефективно застосовані для компактного завадозахищеного (таємного) кодування даних, наприклад, в глобальних комп'ютерних системах міжнародних транснаціональних корпорацій.

РОЗДІЛ 15

ОРГАНІЗАЦІЯ ВЕРТИКАЛЬНО-РЕКУРЕНТНИХ МОДЕЛЕЙ БД НА ОСНОВІ ТЧБ ГАЛУА

15.1. Однорівнева вертикально-рекурентна модель БД в ТЧБ Галуа.

В даній статті досліджуються принципи побудови нової вертикально-рекурентної та багаторівневої вертикально-рекурентної моделі бази даних у базисі Галуа на основі унітарного дерева. Проведений аналіз ефективності двобітового кодування ідентифікаторів даних з інверсіями. Це дозволяє збільшити ефективність кодування ідентифікаторів, значно скоротити об'єм даних, які зберігаються і передаються в міжнародних транснаціональних корпораціях та глобальних розподілених комп'ютерних системах, на прикладі корпорації Pallada Travel.USA та українське ВАТ „Укртелеком”

Ключові слова: вертикально - рекурентна, модель, база даних, базис Галуа, кодування, ідентифікатори. Потужний розвиток сучасних інформаційних систем потребує відповідного розвитку теоретичних засад реалізації системи керування БД та БЗ. Особливо великі масиви даних формуються, підлягають зберіганню та оперативно циркулюють у каналах зв'язку міжнародних корпорацій, до яких належить міжнародна американська корпорація „Pallada Travel.USA”, українське ВАТ „Укртелеком”.

Застосування вертикально-рекурентної моделі БД в базисі Галуа є перспективним для ефективного вирішення задач кодування ідентифікаційних даних в глобальних розподілених комп'ютерних системах міжнародних транснаціональних корпораціях та каналах космічному зв'язку. Тому, що об'єм ідентифікаційних даних в цих системах може значно перевищувати об'єм даних що передаються і реєструються центральними серверами корпорацій, які знаходяться на значних відстанях великої кількості джерел інформації. Наприклад, при підготовці і проведенні реорганізації об'єднання у єдине підприємство, шляхом створення в його складі вертикальних організаційних спеціалізованих структур акціонування та перетворення „Укртелекому”-лідера в сфері телекомунікацій України у відкрите акціонерне товариство. В сучасних структурах руху даних глобальних комп'ютерних систем в міжнародних транснаціональних корпораціях використовуються дві архітектури, які включають ієрархічну модель БД або унітарне дерево. Прикладом використання ієрархічної моделі БД є тетраїдна модель БД яка використовується МТН китайською корпорацією „Тянь Ші”. А більш сучасна модель на основі СЕС, використовується унітарне дерево в корпорації „Pallada Travel USA” та українському ВАТ „Укртелеком”. Лінійно-рекурентна модель БД в базисі Галуа дозволяє замінити метод ідентифікації даних у базисі Радемахера на основі паралельних двійкових кодів рекурентними кодами Галуа, що дозволяє у 8 разів зменшити об'єм ідентифікаційних даних при однаковому числі абонентів корпорації. В цей же час організація руху даних на основі вертикальних дерев є більш ефективним, що потребує відповідної розробки вертикально-рекурентної моделі БД в базисі Галуа. При застосуванні базису Галуа реалізується метод кодування елементів нової ієрархічної вертикально-рекурентної моделі БД в базисі Галуа, що логічно представлена на рис.15.1. Кожен вектор структури є циклічний і рекурентний. Кожен вертикальний вектор базису Галуа виконує вертикальну упаковку інформації найбільш компактно в двовимірних кодах поля Галуа. Метою даної роботи є моделювання вертикально-рекурентної та багаторівневої вертикально-рекурентної моделі бази даних у базисі Галуа на основі унітарного дерева, проведення аналізу ефективності двобітового кодування ідентифікаторів даних з інверсіями, для збільшення ефективності кодування ідентифікаторів і значного скорочення об'єм інформаційних потоків у глобальних комп'ютерних систем. Моделювання логічної структури вертикально-рекурентної моделі БД в базисі Галуа. Порівняння базисів Радемахера та Галуа показує, що коди матриці базису Радемахера, які забезпечують горизонтально-паралельне кодування даних відповідно відображається у вигляді вектора рекурентних кодів базису Галуа. При цьому вертикальна

Figure 1 consists of three diagrams illustrating the construction of the matrix N . Each diagram shows a vertical column of elements with various annotations.

- Diagram 1 (Left):** A vertical column of elements: $0, 0, 0, 1, 0, 1, 1, 1, N-1$. To the left of the column are labels $a_0, a_1, a_2, \dots, a_{i-n}, \dots, a_i$. To the right of the column is a label N . A horizontal line connects the first 0 to the label 1 on the left. A bracket groups the last three 1 's, with an arrow pointing to the label $N-1$ on the right.
- Diagram 2 (Middle):** A vertical column of elements: $0, 0, 0, 1, 0, 1, 1, 1, N-1$. To the left of the column are labels $0, 1, 2, \dots, a_{i-n}, \dots, a_i$. To the right of the column is a label N . A horizontal line connects the first 0 to the label 1 on the left. A bracket groups the last three 1 's, with an arrow pointing to the label $N-1$ on the right.
- Diagram 3 (Right):** A vertical column of elements: $0, 0, 0, 1, 0, 1, 1, 1, N-1$. To the left of the column are labels $0, 1, 2, \dots, a_{i-n}, \dots, a_i$. To the right of the column is a label N . A horizontal line connects the first 0 to the label 1 on the left. A bracket groups the last three 1 's, with an arrow pointing to the label $N-1$ on the right.

де N - об'єм кодового вектора базису Галуа, N_i - кодон ідентифікатора.
Для генерування кодів Галуа потрібно знати ключ:

де n – довжина кодона .

111110101011101100111...0000

↑

КОЛОН

←

Даний алгоритм включає наступні операції:

- Формування послідовності Галуа відбувається рекурентно, поки вона не буде дорівнювати: $2^n - 1$, де n - розмір ключа.

$$N_i = (a_{i-n}, a, \dots, a_i), \quad (15.2)$$

15.2. Дворівнева вертикально-рекурентна модель БД у ТЧБ Галуа

218

нової вертикально-рекурентної моделі БД в базисі Галуа. Вертикально-рекурентна модель бази даних в базисі Галуа складається з вертикальних відвітвлень послідовності ключових вузлів $0, 1, 2, \dots, n$, які дублюються по принципу „Ханойської башти”. При цьому для кодування елементів ієрархічної моделі БД в базисі Радемахера необхідно дані кодувати згідно з кодовою матрицею міжбазисного переходу. Очевидно, що даній кодовій матриці базису Радемахера однозначно відповідає матриця-вектор базису Галуа. При застосуванні базису Галуа реалізується рекурентний метод кодування елементів унітарного дерева. Ієрархічна модель БД в базисі Галуа може бути логічно представленою новою вертикально - рекурентною моделлю БД в базисі Галуа-вектор (рис.15.3) та вертикально - рекурентною моделлю БД в базисі Галуа з відвітвлень (рис.15.4).

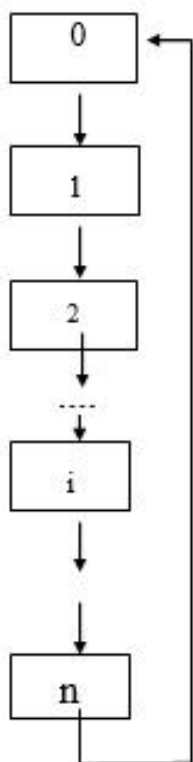


Рис. 15.3– Цикли однорівневої вертикально-рекурентної моделі БД в базисі Галуа-вектор

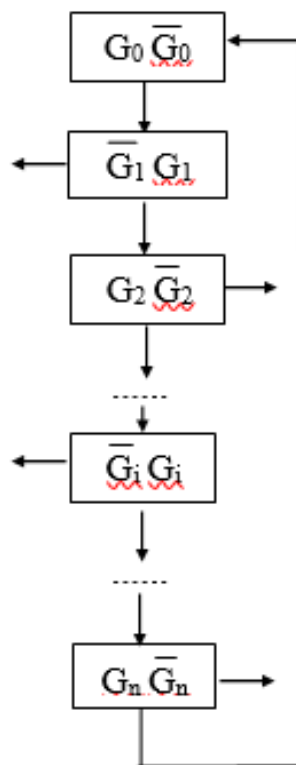


Рис.15.4 –Цикли дворівневої вертикально-рекурентної моделі БД в базисі Галуа з відвітвлень

Особливістю кодування ідентифікаційних даних вертикально-рекурентної моделі БД в базисі Галуа є використання двобітового кодування ідентифікаторів даних з інверсіями. При цьому, якщо інвертується перший біт кодової послідовності Галуа, то розгалуження унітарного дерева відбувається схематично вліво або вниз, а при інвертуванні другого біта Галуа розгалуження відбувається вправо або вгору.

15.3.Багаторівнева вертикально-рекурентна модель БД у ТЧБ Галуа

На основі вертикально-рекурентної моделі БД побудованої на основі вектора базису Галуа можна побудувати нову багаторівневу вертикально-рекурентну модель БД в базисі Галуа.

Логічна структура унітарного дерева СЕС у базисі Галуа на прикладі МТН корпорації „Pallada Travell”, USA показана на рис.15.5.

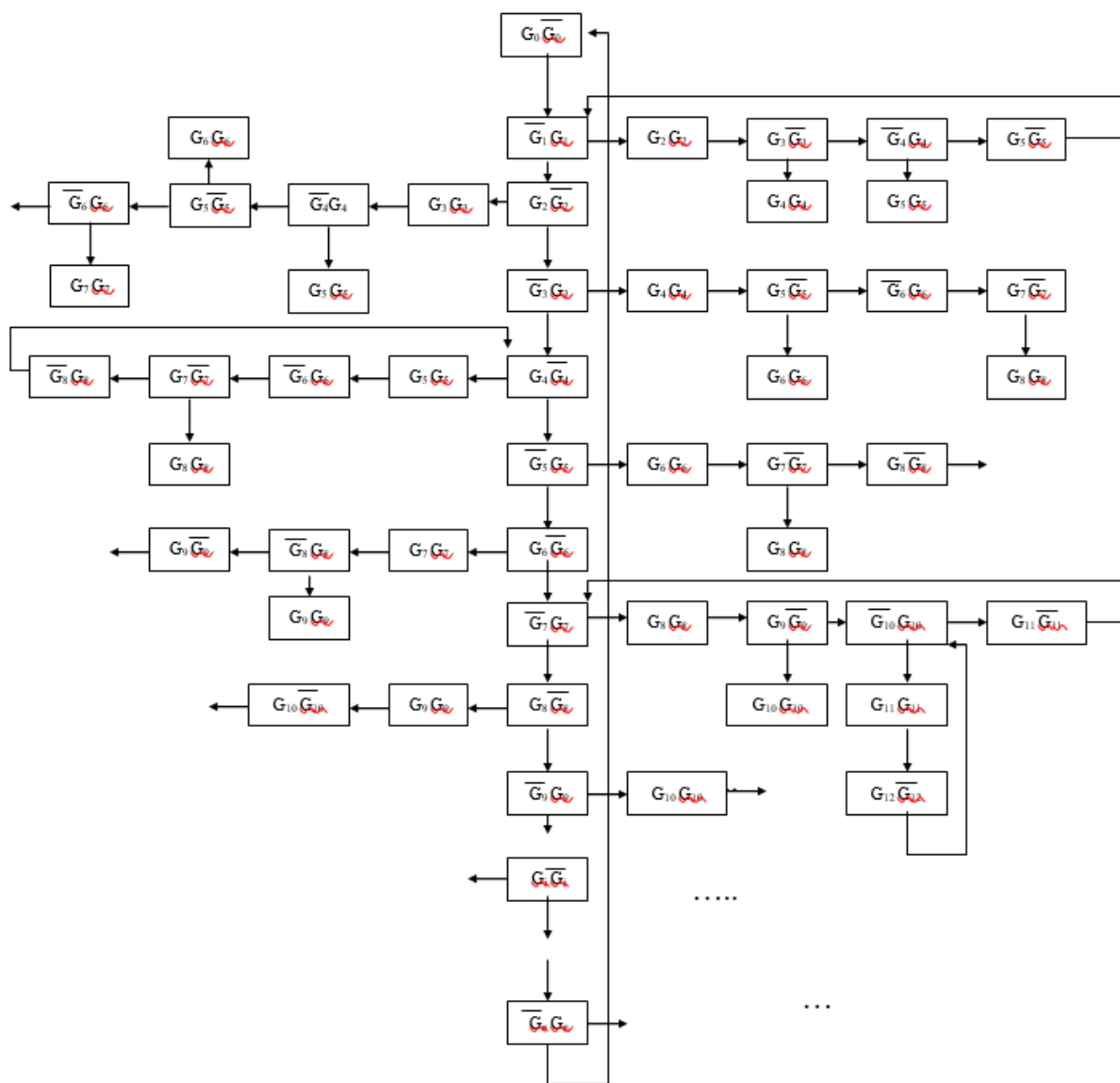


Рисунок 15.5 Ациклічний граф логічної структури циклів багаторівневої вертикально-рекурентної моделі БД базисі Галуа на базі унітарного дерева базису Галуа з відвітвленням руху даних СЕС корпорації „Pallada Travell”, USA.

Критерії ефективності БД.

Важливими характеристиками ефективності БД є:

$$\text{-емерджентність } K_e = \frac{N_3}{N_0}; \quad (15.3)$$

$$\text{-надлишковість } K_c = \frac{mR}{mG}; \quad (15.4)$$

$$\text{-швидкість пошуку } V = \frac{\Delta N_i}{T}; \quad (15.5)$$

де: N_3 - число зв'язків, N_0 - число елементів,

mR - ентропія ієрархічних даних в базисі Радемахера,

mG – ентропія рекурентних даних в базисі Галуа,

ΔN_i - число тактів ідентифікації даних,

T - час доступу до даних.

Фактори які впливають на підвищення ефективності БД

Основними характеристиками оптимізації і підвищення ефективності є:

- здешевлення системи СУБД ;
- збільшення швидкості пошуку доступу до даних;
- зменшення або стиснення об'єму інформації ;
- розробка різноманітних типів третинних пристроїв зберігання (tertiary storage devices)

даних;

- тенденції росту системи ;
- підвищення продуктивності системи ;
- паралельні обчислення;
- розвиток системи клієнт/сервер і багаторівневої архітектури;
- підвищення надійності захисту від помилок;
- захист від несанкціонованого доступу до БД тощо.

Розрахунок ефективності кодування елементів в базисі Галуа.

Для розрахунку ефективності фізичної структури задамо початкові умови. Нехай число елементів унітарного дерева дорівнює $m=1$, число схематично показано у вигляді горизонтальних відгалужень рівних $m=2$, число вертикальних відгалужень рівне $m=3$. Відповідно позначимо m_i та m_n відгалужень. Розрахунок об'єму даних відомої моделі унітарного дерева виконано на основі формули Хартлі.

$$V_i = \hat{E}[\log_2 m_i]; i \in 1, n \quad (15.6)$$

$$\sum_{i=1}^n V_i = \sum_{j=1}^k \hat{E}[\log_2 m_j] \quad (15.7)$$

де V_i - елемент коду, $\sum_{i=1}^n V_i$ - загальний об'єм кодів.

Аналіз ефективності кодування ідентифікаційних даних вертикально-рекурентної та багаторівневої вертикально-рекурентної моделі БД в базисі Галуа. Переваги і реалізація

Особливості кодування ідентифікаційних даних у базисі Галуа. Для цього прикладу об'єм ідентифікаційних даних нижнього абонента складає 28 *bit* при загальному числі 38100 абонентів.

Виходячи з максимального числа елементів БД визначимо розрядність коду індифікатора елементів в базисі Галуа. Якщо в БД число елементів тісно пов'язані між собою і не перевищує $n = 100$, то ефективність кодування БД в базисі Галуа можна підвищити наступним чином, шляхом кодування числа елементів в групі після коду ідентифікатора.

Звідки визначається коефіцієнт стиснення об'єму даних вертикально-рекурентної моделі БД при переході в базис Галуа по відношенні до базису Радемахера.

Наприклад: $m_1=1000; m_2=2000; m_3=500; m_4=200; \text{modi } v_1=10; v_2=11; v_3=9; v_4=8$

$$\text{bit}; \sum_{i=1}^n V_i = 28\,000 \text{ bit} \quad K_e = 28,000 \cdot 3,8100 = 106,68 / 7,4 = 14,42.$$

Дослідження теоретико-числових базисів та їх застосування при організації вертикально-рекурентної та багаторівневої вертикально-рекурентної моделі БД в базисі Галуа показують, що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадоне захищеним крім того, традиційний метод є незахищеним від несанкціонованого доступу. В той же час дані характеристики БД достатньо ефективно реалізується в базисі Галуа, що визначає перспективу його застосування при організації БД та БЗ.

Перевага вертикально-рекурентної та багаторівневої вертикально-рекурентної моделі БД в базисі Галуа ефективніша в 14,42 рази ніж кодування ідентифікаційних даних унітарного дерева СЕС корпорації „Pallada Travell”USA. Очевидно, що глибокі дослідження застосування теорії і методів кодування даних та організації БД в базисі Галуа є актуальними і перспективними. Запропоновані вертикально-рекурентні та багаторівневі вертикально-рекурентні моделі БД в базисі Галуа соціально-економічного структурування, можуть бути

ефективно застосовані для компактного завадозахищеного (таємного) кодування даних, наприклад, в міжнародних транснаціональних корпораціях глобальних розподілених комп'ютерних систем та каналах космічному зв'язку. Основною вимогою клієнтів комп'ютерних мереж є забезпечення найвищої швидкості передачі і приймання інформації при мінімальних спотвореннях, а основною вимогою мережених компаній є забезпечення найвищої ефективності використання мережі при достатній якості передачі. Задовольнити ці вимоги на сучасному рівні можуть мережі технології АТМ (Asynchronous Transfer Mode)-режим асинхронного передавання даних-це новітні мережі з інтеграцією послуг, у яких комутація здійснюється не на рівні каналів, а на рівні пакетів даних. Ця технологія на протязі 5-10 років замінить існуючі локальні та глобальну мережі.

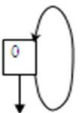
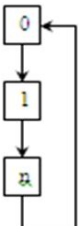
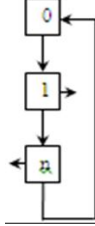
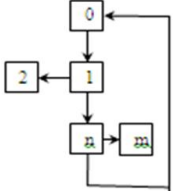
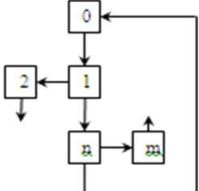
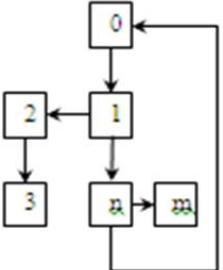
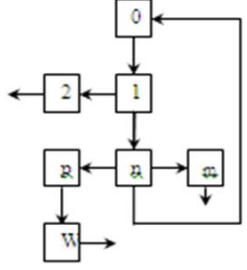
Результати моделювання

Дослідження теоретико-числових базисів та їх застосування при організації БД показують, що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадонебезпечним крім того, традиційний метод є незахищеним від несанкціонованого доступу. В той же час дані характеристики БД достатньо ефективно реалізується в базисі Галуа, що визначає перспективу його застосування при організації БД та БЗ.

Категорії нових схем, однорівневі, дворівневі та багаторівневі вертикально-рекурентні моделі БД в базисі Галуа, які запропоновані для мереж структур БД, здатні моделювати в повному обсязі ієрархічну структуру та, власне, вертикально-рекурентні схеми. Окрім того, у загальному випадку вони дають можливість описувати петлі та цикли, зображені у табл.15.1.

Таблиця.15.1

Категорії схем вертикально-рекурентних моделей баз даних в базисі Галуа

Схема	Структура вертикально-рекурентних баз даних в базисі Галуа	
	Без відвітвлення	З відвітвленням
Петля з циклом		
Однорівнева схема з циклами, ланцюгами і рекуренцією		
Дворівнева схема з циклами, ланцюгами і рекуренцією		
Багаторівнева схема з циклами, ланцюгами і рекуренцією		

Перевага вертикально-рекурентної моделі БД в базисі Галуа буде більш ефективна в 14,42 рази ніж кодування ідентифікаційних даних унітарного дерева СЕС корпорації „ Pallada Travell ” USA. Проведені дослідження моделей БД показують, що перспективним є розвиток та досконалення нової вертикально-рекурентної моделі БД в базисі Галуа соціально-економічного структурування. Дослідження теоретико-числових базисів та їх застосування при організації БД, показують що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадоне захищеним, крім того, традиційний метод є незахищеним від несанкціонованого доступу. В той же час дані характеристики БД достатньо ефективно реалізується в базисі Галуа. Очевидно, що глибокі дослідження застосування теорії і методів кодування даних та організації БД в базисі Галуа є актуальними і перспективними. Запропоновані вертикально-рекурентні моделі БД в базисі Галуа соціально-економічного структурування, можуть бути ефективно застосовані для компактного завадоне захищеного (таємного) кодування даних, наприклад, в міжнародних транснаціональних корпораціях глобальних розподілених комп'ютерних систем. Основою вимогою клієнтів комп'ютерних мереж є забезпечення найвищої швидкості передачі і приймання інформації при мінімальних спотвореннях, а основною вимогою мережевих компаній - є забезпечення найвищої ефективності використання мережі при достатній якості передачі. Задовольнити ці вимоги на сучасному рівні можуть мережі технології АТМ (Asynchronous Transfer Mode)- режим асинхронного передавання даних-це новітні мережі з інтеграцією послуг, у яких комутація здійснюється не на рівні каналів, а на рівні пакетів даних. Ця технологія на протязі 5-10 років замінить існуючі локальні та глобальну мережі.

РОЗДІЛ 16

ОРГАНІЗАЦІЯ РЕЛЯЦІЙНИХ МОДЕЛЕЙ БАЗ ДАНИХ НА ОСНОВІ ДВОМІРНИХ КОДІВ ПОЛЯ ГАЛУА

16.1. Двомірні коди поля Галуа

Теоретичні основи кодів поля Галуа є теорія чисел, теорія структур, груп та полів Галуа. Одночасно теоретичні засади кодів поля Галуа базуються на системі ортогональних дискретних функцій базису Уолша, у середовищі яких існує система квазістаціонарних рекурентних функцій Галуа.

Принципи кодування методом залишків базується на теоретико-числовому підході і представляють собою клас векторних багаторівневих кодів поля Галуа. Застосування цього класу КПП дозволяє зменшити надлишковість вимірювальної інформації за рахунок виключення з інформаційного потоку кодів рангів Діофантового рівняння.

Двовимірні дворівневі КПП можуть бути задані у вигляді матриць з числом елементів $V = 2k \cdot 2m$, які приймають пари бінарних значень $by \in \{00, 01, 10, 11\}$, $i \in \{1, 2, k\}$; $j \in \{1, 2, m\}$.

16.1.1. Матричні двомірні коди поля Галуа

Оптимізація сучасних технологій побудови БД є одним із вирішальних факторів забезпечення ефективного зберігання корпоративної інформації та використання даних користувачем. Реляційна модель БД в даний час отримала широке застосування і ефективно використовується при створенні потужних баз даних і баз знань. Перевагою реляційних БД по відношенню до ієрархічних є забезпечення можливості формалізації перетворення даних на основі алгоритмів реляційної алгебри представлених в табл.16.1.

Таблиця.16.1

Формалізація перетворення даних

Об'єднання	$R(L) = R_1(L) \cup R_2(L) = \{r \mid r \in R_1 \vee r \in R_2\}$,	Обмеження	$S = R[L \theta M] = \{r \mid r \in R \& r[L] \theta r[M]\}$
Перетин	$R(L) = R_1(L) \cap R_2(L) = \{r \mid r \in R_1 \& r \in R_2\}$,	Декартів добуток	$Q = R \times S = \{(r, s) \mid r \in R \& s \in S\}$
Різниця	$R(L) = R_1(L) - R_2(L) = \{r \mid r \in R_1 \& r \notin R_2\}$,	З'єднання	$Q = R[L \theta M] \cup S = \{(r, s) \mid r \in R \& s \in S \& r[M] \theta s[N]\}$,
Проекція	$S = R[A_{s1}, \dots, A_{sn}] = \{r[A_{s1}, \dots, A_{sn}] \mid r \in R\}$,	Ділення	$R[N \div K]S = R[M] (R[M] \times S[K]) - R[M]$,

Структура двомірного файлу - відношення реляційної бази даних показана на рис.16.1.

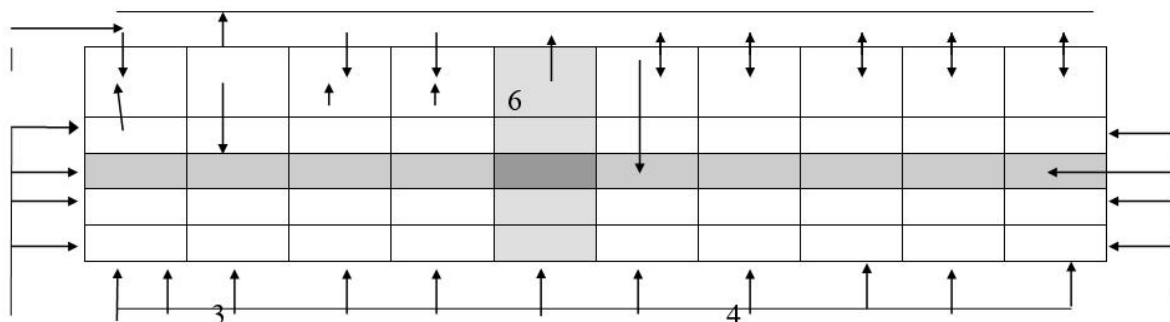


Рис.16.1 - Структура відношення реляційної БД

На рис.16.1: 1- заголовок; 2- власна змістовна частина; 3- первинний ключ; 4- атрибути; 5- кортежі; 6 – домени.

Даний клас БД не в повній мірі забезпечує компактність кодування ідентифікаторів і адресацію даних. Кодування реляційних даних згідно структури відношення реляційної БД поданої на рис.16.1 виконується в двійковій системі числення в базисі Радемахера.

16.1.2. Критерії оцінки ефективності кодування даних в реляційній БД.

У загальному випадку для зберігання даних кортежа реляційної БД, виходячи з умови байт-орієнтованого кодування, загальний об'єм даних кортежа обчислюється згідно формули :

$$I_j = \sum_{i=1}^n d_{ij} \text{ (байт) }, \quad (16.1)$$

де d_{ij} - об'єм даних ij – го атрибута кортежу, (1б, 1Кб, 1Мб, 1Тб., 1Пб.,...);

$n = 1, 2, \dots$, - число атрибутів.

Об'єм даних: відношення реляційної БД складає:

$$I_v = m I_j, \quad (16.2)$$

де $m = 1, 2, \dots$, - число кортежів.

Кількість доменів: відношення реляційної БД складає:

$$D = n m. \quad (16.3)$$

Кожен домен реляційної БД має свій ідентифікаційний код, який буде мати наступні параметри :

$$C_{ij} = \hat{E} [\log_2 n] + \hat{E} [\log_2 m]; \quad (16.4)$$

де, $\hat{E} [\cdot]$ - цілочисельна функція з округленням до більшого цілого.

Таким чином повний об'єм матриці ідентифікаційних даних складає:

$$V_c = C_{ij} n m. \quad (16.5)$$

Отже оцінка співвідношення об'єму матриці ідентифікаційних даних, до об'єму даних відношення реляційної БД складає:

$$K = \frac{V_c}{I_v} \cdot 100\%. \quad (16.6)$$

Проводимо аналіз співвідношення об'ємів матриць ідентифікаційних кодів та атрибутів реляційної БД.

$$M = \hat{E} [\log_2 V_c] 2^4. \quad (16.7)$$

16.2. Рекурентні двомірні коди поля Галуа

16.2.1. Теоретичні основи одновірних кодів поля Галуа

Для одновірного кодування застосовується формула генерування одновірного коду поля Галуа

$$G_{i+1} = G_i \oplus G_{i-n} \quad (16.8)$$

де G_i - текущий біт Галуа, $\oplus$ - додавання по mod2, G_{i-n} - зміщення розрядів біт Галуа.

16.2.2. Теоретичні основи двовірних кодів поля Галуа

На основі базису Галуа застосовується двовірне кодування.

$$(D \underline{X} \underline{X}_i) \quad G \quad D \text{ 1.1} \quad (16.9)$$

Для двовірних кодів поля Галуа аналітичний вираз генератора має вид:

$$G_{i+1, j+1} = G_{i+1}^{\circ} \wedge G_{Gj+1}^{\circ} \quad (16.10)$$

$$\text{де } G_{i+1}^{\circ} = G_i \oplus \bar{G}_{i-n}; G_{j+1}^{\circ} = G_j \oplus \bar{G}_{j-m}.$$

16.3. Дослідження і обробки даних результатів аналізу об'ємів матриці ідентифікаторів атрибутів і кортежів відношення реляційних БД

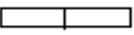
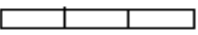
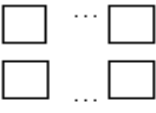
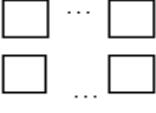
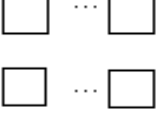
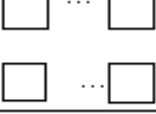
Результати аналізу об'ємів матриці ідентифікаторів атрибутів і кортежів відношення реляційних БД показані в табл.16.2.

Кортежі відношення РБД.

На основі аналітичних виразів (16.1-16.7) та (табл. 16.2) будемо графік функції коефіцієнта К ефективності кодування між параметрами відношення реляційної БД в базисі Радемахера (рис.16.2).

Таблиця 16.2

Результати аналізу об'ємів матриці ідентифікаторів атрибутів

i	j	d _{ij}	Архітектура РБД	c _{ij}	v _c	K	M
1	1	1		0	0	0	0
2	1	2		1	2	0,78	16
3	1	3		2	6	2,34	48
2	2	4		3	16	3,13	64
4	4	16		4	64	25	96
6	8	48		6	286	112,5	130
8	16	128		16	896	350	157
12	50	600		50	600	2314	201

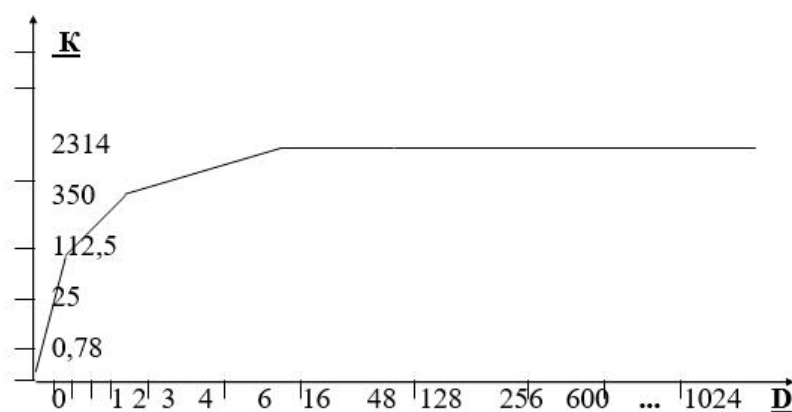


Рис.16.2.Залежності між параметрами відношення реляційних БД та об'ємом матриці ідентифікаційних кодів атрибутів відношення.

16.3.1. Алгоритм визначення коефіцієнта надлишковості реляційної бази даних в базисі Радемахера при багатьох атрибутів і кортежі реляційної БД

Коефіцієнт надлишковості реляційної БД можна обчислити згідно виразу

$$K_1 = \frac{Cij}{Iv} 100\% \quad (16.11)$$

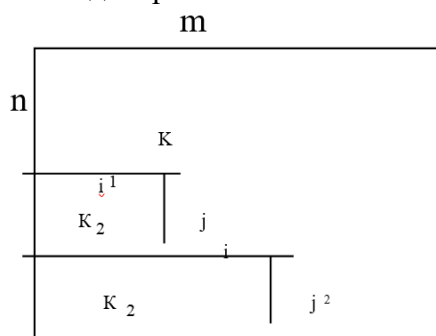
Результати дослідження коефіцієнтів надлишковості реляційної БД показані в табл.16.3
Таблиця.16.3

Результати дослідження коефіцієнтів надлишковості РБД

i	j	n	m	K ₁ %
1	1	1	1	0
1	2	1	2	50
1	3	1	3	66
1	4	1	4	50
1	8	1	8	36
1	128	1	128	30
1	256	1	256	37
1	2	1	2	50
2	2	2	2	44
2	3	2	3	83
4	4	4	4	50
8	16	8	16	30
12	50	12	50	13

16.3.2.Алгоритми визначення коефіцієнта надлишковості при багатьох масивах доменів

У випадку коли потрібно вибрати масив даних з БД ,коли вони суміжні тоді використовуємо формулу визначення коефіцієнта ефективності кодування БД (10), фрагмент матиме вигляд на рис.16.3



$$K_3 = \sum_1^p K_2 \quad (16.12)$$

Рис.16.3 Фрагмент масиву БД при визначенні коефіцієнта K₃ (K₃ - сума коефіцієнтів ефективності кодування БД суміжних фрагментів).

У випадку коли потрібно вибрати масиви з БД , які у вигляді окремих фрагментів тоді використовуємо формулу визначення коефіцієнта ефективності кодування БД (16.13), фрагмент матиме вигляд на рис.16.4.

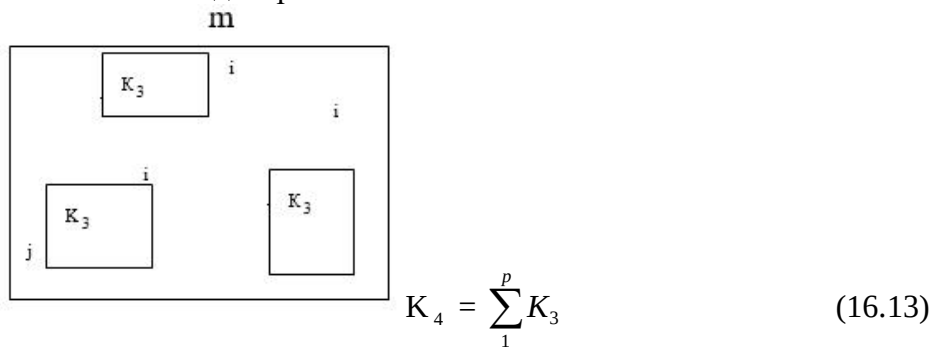


Рис.16.4. Фрагмент масиву БД при визначенні коефіцієнта K_4 (K_4 -сума коефіцієнтів ефективності кодування БД окремих фрагментів).

У випадку, коли масиви в БД представлені у вигляді окремих фрагментів, тоді використовуємо формулу визначення коефіцієнта ефективності кодування БД (9.14). Фрагмент матиме вигляд, показаний на рис.16.5.

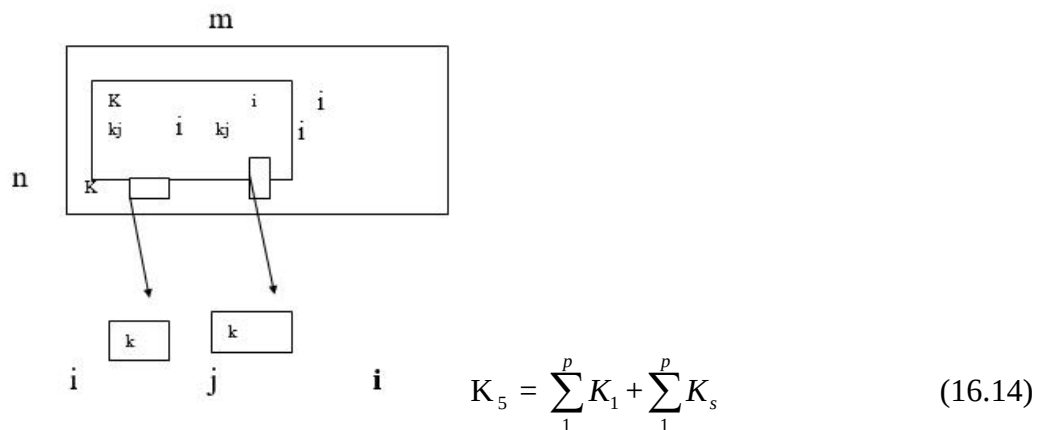


Рис.16.5. Фрагмент масиву БД при визначенні коефіцієнта K_5 (K_5 -сума коефіцієнтів ефективності кодування БД всіх окремих деталей фрагментів).

З допомогою досліджених результатів M та визначених значень коефіцієнта K побудуємо графік функції коефіцієнта ефективності кодування БД.

Графік залежності функції коефіцієнта ефективності кодування БД в базисі Радемахера при великих масивах даних (рис 16.7).

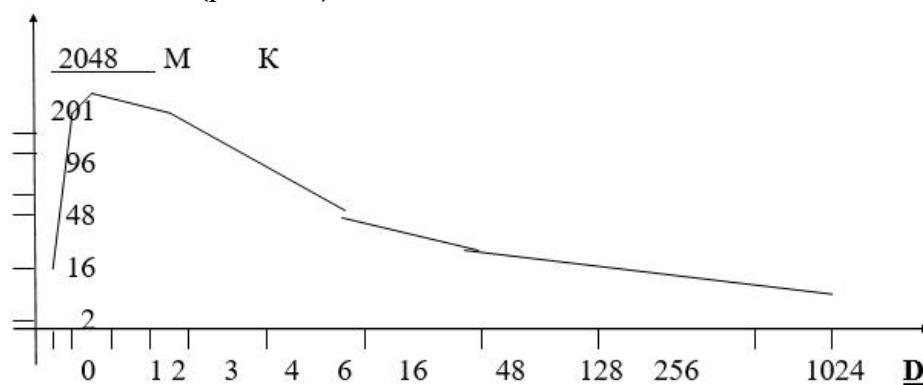


Рис. 16.7. Залежності функції коефіцієнта ефективності кодування БД в базисі Радемахера при великих масивах даних.

Базис Радемахера породжує двійкову систему числення. В базисі Радемахера працюють всі сучасні комп'ютерні системи. На основі базисів Радемахера застосовується одномірне кодування:

$$(D \text{ XXX.XXX}) R \quad D 10.10 \quad (16.15)$$

Теоретичні основи одномірних кодів поля Галуа

Базис Галуа породжує систему числення Галуа. Наприклад при $m=4$; $N=2^4=16$ кодовий вектор базису Галуа описується двійковою послідовністю - кодон Галуа 11110101110010000.

При кодуванні базису Галуа на кожен N_i число припадає 1 біт Галуа .А для кодування чисел N з вектору необхідні рекурентно вибрати m -біт даних що засвідчує максимальну упаковку інформації в базисі Галуа.

Для одномірного кодування застосовується формула генерування одномірного коду поля Галуа

$$G_{i+1} = G_i \oplus G_{i-n} \quad (16.16)$$

де G_i - текучий біт Галуа, $\oplus$ -додавання по mod2, G_{i-n} -зміщення розрядів біт Галуа.

Теоретичні основи двомірних кодів поля Галуа

На основі базису Галуа застосовується двомірне кодування.

$$(D \text{ X. X.}) \quad G \quad D 1.1. \quad (16.17)$$

Для двомірних кодів поля Галуа аналітичний вираз генератора має вид

$$G_{i+1,j+1} = G_{i+1}^{\circ} \wedge G_{Gj+1}^{\circ} \quad (16.18)$$

$$\text{де } G_{i+1}^{\circ} = G_i \oplus G_{i-n}; G_{j+1}^{\circ} = G_j \oplus G_{j-m}.$$

Дослідження теоретико–числових базисів та їх застосування при організації БД, показують що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадонебезпечним, крім того традиційний метод є незахищеним від несанкціонованого доступу, що достатньо ефективно реалізується в базисі Галуа.

РОЗДІЛ 17

РЕЛЯЦІЙНО-РЕКУРЕНТНА МОДЕЛЬ БД В ТЧБ ГАЛУА

17.1. Побудова відношення реляційно-рекурентної моделі бази даних в ТЧБ Галуа

Реляційно-рекурентна модель бази даних двомірних кодів в ТЧБ Галуа у вигляді відношення реляційних баз даних в базисі Галуа показано в табл.17.1.

Таблица.17.1

Відношення реляційно-рекурентної моделі баз даних в ТЧБ Галуа.

d_H	1	1	1	1	0	1	0	1	0	0	1	0	0	0	0
1	11	11	11	11	01	11	01	11	01	01	11	01	01	01	01
1	11	11	11	11	01	11	01	11	01	00	11	01	01	01	01
1	11	11	11	11	01	11	01	11	01	00	11	01	01	01	01
1	11	11	11	11	01	11	01	11	01	00	11	01	01	01	01
0	10	10	10	10	00	10	00	10	00	00	10	00	00	00	00
1	11	11	11	11	01	11	01	11	01	01	11	01	01	01	01
0	10	10	10	10	00	10	01	10	00	00	10	00	00	00	00
1	11	11	11	11	01	11	01	11	01	01	11	01	01	01	01
0	10	10	10	10	00	10	00	10	00	00	10	00	00	00	00
0	10	10	10	10	00	01	00	10	00	00	10	00	00	00	00
1	11	11	11	11	01	11	01	11	01	01	11	01	01	01	01
0	10	10	10	10	00	10	00	10	00	00	10	00	00	00	00
0	10	10	10	10	00	10	00	10	00	00	10	00	00	00	00
0	10	10	10	10	00	10	00	10	00	00	10	00	00	00	00
0	10	10	10	10	00	10	00	10	00	00	10	00	00	00	00

17.2. Розробка рекурентних двовірних кодів поля Галуа

На основі порівняння базисів Радемахера Галуа побудуємо відношення реляційних БД двовірних кодів (табл.17.2).

Таблица 17.2.

Порівняльна характеристика одномірного коду Радемахера з двомірним Галуа

Одномірний код ідентифікатора атрибутів даних в базисі Радемахера		Двомірний код ідентифікатора атрибутів даних в базисі Галуа
i	j	i j
0 0 0	0 0 0	0 0
0 0 1	0 0 1	0 0
...	...	...
1 1 1	1 1 1	1 1

З отриманих результатів дослідження можна зробити порівняльну характеристику одномірного коду ідентифікатора атрибутів даних в базисі Радемахера з двомірним кодом ідентифікатора атрибутів даних в базисі Галуа в табл.17.2 для $i = 1, 2, \dots, n$; $j = 1, 2, \dots, m$; подані двомірний код Галуа.

17.3. Аналіз ефективності кодування ідентифікаторів в реляційній БД та рекурентній БД.

Оцінку аналізу ефективності кодування ідентифікаторів РБД двомірними кодами Галуа, виконуємо на основі виразу.

$$K_e = \log_2 \left[\frac{E \cdot [\log_2 n] + E \cdot [\log_2 m]}{2} \right], \quad (17.1)$$

де, n, m – число атрибутів та кортежів реляційної БД на рис.17.1 показані графіки ефективності кодування ідентифікаторів реляційної БД на основі рекурентних двомірних послідовностей Галуа.

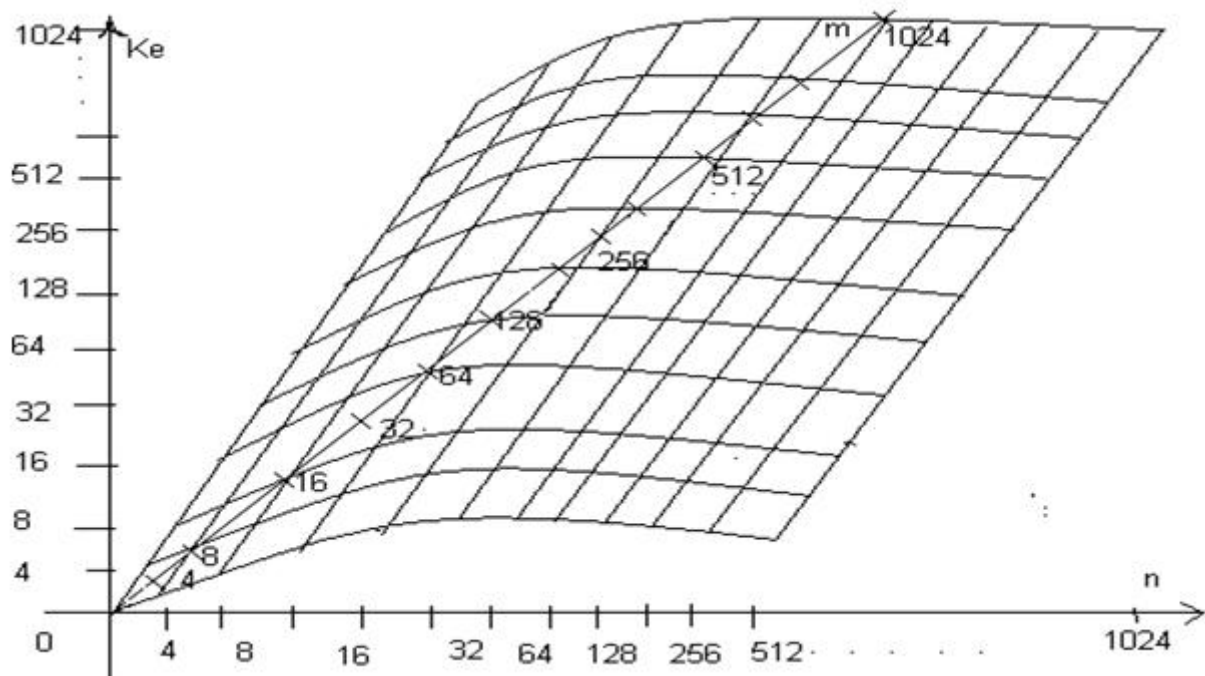


Рис.17.1. Графіки ефективності кодування ідентифікаторів реляційної БД на основі рекурентних двомірних послідовностей Галуа.

Дослідження теоретико-числових базисів та їх застосування при організації БД, показують що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадоне захищеним, крім того традиційний метод є незахищеним від несанкціонованого доступу, що достатньо ефективно реалізується в базисі Галуа.

РОЗДІЛ 18

ОБ'ЄКТНО-РЕЛЯЦІЙНО РЕКУРЕНТНА МОДЕЛЬ СХОВИЩ ДАНИХ В ТЧБ ГАЛУА

18.1. Об'єктно-реляційно-рекурентна модель сховищ даних глобальних розподілених комп'ютерних систем в ТЧБ Галуа

На основі реляційно-рекурентної моделі БД в ТЧБ Галуа та з використанням розподіленої, багатовимірної, кубів даних, просторів даних, надвеликих БД, СД та комп'ютерної БД можна побудувати Об'єктно-реляційно-рекурентну модель сховищ даних глобальних розподілених комп'ютерних систем в базисі Галуа.

На основі моделі глобальної комп'ютерної системи, розроблена об'єктно-реляційно-рекурентна модель сховищ даних глобальних розподілених комп'ютерних систем в теоретико-числовому базисі Галуа, для побудови моделі та системи управління конгломератами епохальних сховищ даних в базисі Галуа.

Широкому використанню моделей руху даних сприяє швидкий розвиток комп'ютерних систем найрізноманітнішого призначення. До важливого класу таких КС слід віднести системи реального часу, проблемно-орієнтовані комп'ютерні системи, низові комп'ютерні та малі локальні мережі, а також спеціалізовані комп'ютерні системи.

Особливо актуальною проблемою є розробка практично значимої теорії руху даних для розподілених комп'ютерних систем, які обслуговують технологічні об'єкти та промислові комплекси і традиційно включають технологічні, цехові та адміністративні рівні контролю станів та управління об'єктами. При цьому в якості об'єктів управління з кібернетичних позицій можуть розглядатися як технологічні, так і природничі та соціальні об'єкти.

В ГКС виникає потреба збору, обробки, зберігання, передачі даних і створення сучасних велико-об'ємних баз даних глобальних комп'ютерних систем. Одним із перспективних напрямків теорії організації БД є реалізація доступу до БД ГКС можна запропонувати нові об'єктно-орієнтовані-рекурентні, об'єктно-реляційні-рекурентні та напівструктуровані-рекурентні моделі бази даних глобальних комп'ютерних систем в базисі Галуа.

Нові запропоновані моделі БД ГКС є новим типом структурної організації баз даних, яка формується на основі рекурентних властивостей теоретико-числового базису Галуа. Важливою характеристикою таких БД є максимальна компактність адресації даних, в тому числі при багаторівневій архітектурі. Позитивною характеристикою БД ГКС є можливість ефективного захисту від помилок та несанкціонованого доступу. Одним із вирішальних факторів забезпечення компактного та ефективного зберігання корпоративної інформації є вдосконалення сучасних технологій побудови баз даних.

Тому дослідження ефективності БД ГКС є актуальною науково-технічною задачею. При проектуванні бази даних, насамперед, вирішується дві задачі: організація структури всієї бази даних, що визначає типи зв'язків між даними, і організація структури самих даних.

Постановка задачі. Метою роботи є дослідження та комп'ютерне моделювання нової об'єктно-реляційно-рекурентної моделі сховищ даних глобальних розподілених комп'ютерних систем в ТЧБ Галуа на основі розробленої моделі глобальної комп'ютерної системи проведення аналізу ефективності кодування ідентифікаційних даних.

На основі розробленої глобальної моделі комп'ютерної системи, логічно представленої на рис.18.1, можна побудувати нову об'єктно-реляційно-рекурентну модель сховищ даних ГРКС в ТЧБ Галуа, що реалізується у вигляді взаємодії п'яťох типів системних об'єктів .

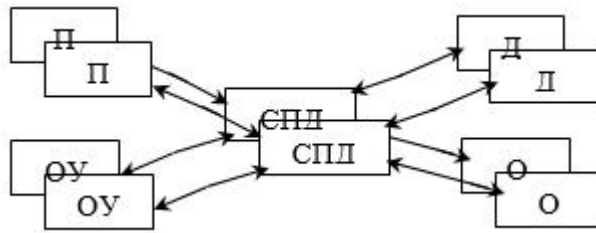


Рис.18.1 . Глобальна модель комп'ютерної системи.

На рис.18.1: П-процесор Д-дані, СПД- система передавання даних, ОУ-об'єкт управління, О-оператор.

Кожен з наведених системних об'єктів може виконувати в середовищі КС чотири системні функції: – формування даних;– передавання даних;– цифрова обробка даних;– приймання та зберігання даних.Тобто, кожен з системних об'єктів (СО) може бути одним з функціональних об'єктів наступного типу:джерело інформації (ДІ);середовище передавання інформації (СПІ);середовище цифрової обробки інформації(СОІ); приймач інформації(ПІ). Формалізація опису характеристик системних об'єктів КС.

У загальному випадку ресурсні характеристики (СО):

Проектованої КС можуть бути достатньо повно описані функціоналом, який заданий коефіцієнтом E_{co} та четвіркою параметрів:

$$E_{co} = F(T, V, M, S) \quad (18.1)$$

де T – час використання ресурсів, V – швидкість виконання системних операцій (формування, передавання, цифрова обробка та зберігання даних), M –об'єм використаного ресурсу пам'яті, S – системні функції.

Системні характеристики процесора (П):

Процесори (П) містять в своєму складі інтелектуальні оснащені одно кристальними спец процесорами сенсори, сигнальні процесори, мікро– та міні-контролери, контролери низових мереж, комутаційні процесори, сервери та комп'ютерні кластери[].Системні характеристики (П) достатньо повно описуються функціоналом:

$$E_p = F(T, V, M, S) \quad (18.2)$$

де T – час використання ресурсів, V – швидкість виконання системних операцій(формування,передавання,цифрова обробка та зберігання даних), VR/Vw , M –об'єм використаного ресурсу пам'яті, S – системні функції операційної системи та прикладного програмного забезпечення.

Системні характеристики даних (Д):

Сучасна систематизація даних охоплює три їх основні класи.

- фізичні дані;
- логічні дані;
- віртуальні дані.

Ресурсні характеристики СО „Дані” – представлені часом зберігання, швидкістю запису, швидкістю зчитування,об'ємом та захистом від помилок:

$$E_d = F(T, VR, Vw, S) \quad (18.3)$$

де T – час зберігання, VR – швидкість , Vw – швидкість зчитування, M –об'єм використаного ресурсу пам'яті, S – захист від помилок та несанкціонованого доступу.

Характеристика моделей об'єктів управління (ОУ):

Об'єкт управління адекватно може бути описаний характеристичними параметрами – часом, ентропією, моделлю об'єкта та системними функціями:

$$E_{ou} = F(T, M, I, S) \quad (18.4)$$

де T – час, M –модель об'єкта, I – ентропія, S –системні функції.

Системні характеристики (СПД):

СПД – представлені швидкістю приймання, швидкістю передавання даних, ймовірності помилок, часом затримки в каналі:

$$ЕСПД = F(VR, Vw, Pi, T) \quad (18.5)$$

де VR– швидкість приймання даних, Vw– швидкість передавання даних, Pi – ймовірність помилок, T– час затримки в каналі,

Системні характеристики операторів (О):

Оператор представлений часом праці, швидкістю запису, швидкістю зчитування, потенційною пам'яттю та знаннями

$$EO = F(T, VR, VwS, M) \quad (18.6)$$

де T– час праці, VR– швидкість запису, Vw– швидкість зчитування, S– знання, M– пам'ять.

18.2. Логічна структура об'єктно-реляційно-рекурентної моделі сховищ даних глобальних розподілених комп'ютерних систем в ТЧБ Галуа.

Об'єктно-реляційна модель БД є частиною стандарту SQL, названого SQL-99 (або SQL-1999, також SQL3) і є варіант розширення звичайної реляційної моделі за рахунок формалізації багатьох загальноприйнятих концепцій об'єктно-орієнтованого проектування. Вказаний стандарт служить основою для побудови об'єктно-реляційних систем баз даних, які сьогодні випускаються всіма основними постачальниками комерційних СУБД. Ці системи, однак, значно відрізняються в деталях практичної реалізації початкових концепцій.

На основі розробленої реляційно-рекурентної моделі БД в ТЧБ Галуа [], розподіленої БД, над великих БД, комп'ютерної моделей БД і сховища даних можна розробити об'єктно-реляційно-рекурентну модель сховища даних глобальних розподілених комп'ютерних систем в ТЧБ Галуа. Яка в свою чергу буде „цеглинкою“ в багаторівневих лінійно-рекурентних та вертикально-рекурентних моделях БД в ТЧБ Галуа при побудові конгломератів епохальних сховищ даних в ТЧБ Галуа. На рис.18.2 показана структура об'єктно-реляційної –рекурентної моделі БД.

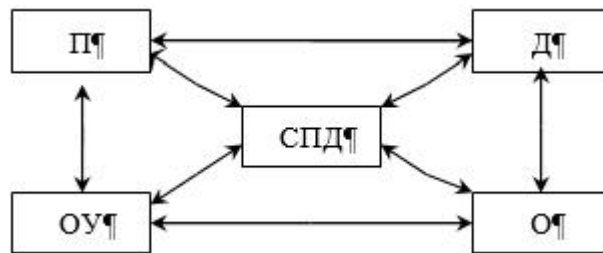


Рис.18.2. Об'єктно–реляційна–рекурентна модель СД ГРКС в ТЧБ Галуа

Система об'єктів (наприклад: даних) характеризується емерджентністю в тому випадку коли

$$K_e = \frac{N_3}{N_0}, \quad K_e > 2 \quad (18.7)$$

де N_3 -число зв'язків БД, N_0 -число елементів БД, K_e – критерій емерджентності БД

18.3. Фізична структура об'єктно-реляційно-рекурентної моделі сховищ даних глобальних розподілених комп'ютерних систем в ТЧБ Галуа

Проведені дослідження архітектури БД показують що ієрархічна, мережеві та реляційні моделі БД не задовільняють умови емерджентності $K_e > 2$.

Тому перспективним є розвиток та удосконалення моделей БД об'єктно-орієнтованих, об'єктно-реляційних $K_e < 4$.

На основі багаторівневих схем - категорії схем та вектора базису Галуа можна побудувати нові об'єктно-орієнтовані-рекурентні, об'єктно-реляційні, напівструктуровані моделі БД в базисі Галуа.

Оцінка надлишковості бази даних.

Зменшення надлишковості в різних моделях БД досягається за рахунок зростання числа зв'язків між об'єктами БД. Таким чином оцінку надлишковості моделі БД можна розрахувати на основі критерію емерджентності (інтелектуальності) K_e . Тому перспективним є розвиток та удосконалення моделей БД об'єктно-орієнтованих, об'єктно-реляційних та напівструктурованих $K_e < 4$.

Критерії ефективності БД.

Важливими характеристиками ефективності БД є:

$$\text{— емерджентність } K_e = \frac{N_3}{N_0}; \quad (18.8)$$

$$\text{— надлишковість } K_c = \frac{mR}{mG}; \quad (18.9)$$

$$\text{— швидкість пошуку даних } V = \frac{\Delta N_i}{T}; \quad (18.10)$$

де: N_3 – число зв'язків, N_0 – число елементів, mR – ентропія ієрархічних даних в базисах:

- Радемахера, mG – ентропія рекурентних даних;
- Галуа, ΔN_i – число тактів ідентифікації даних, T – час доступу до даних.

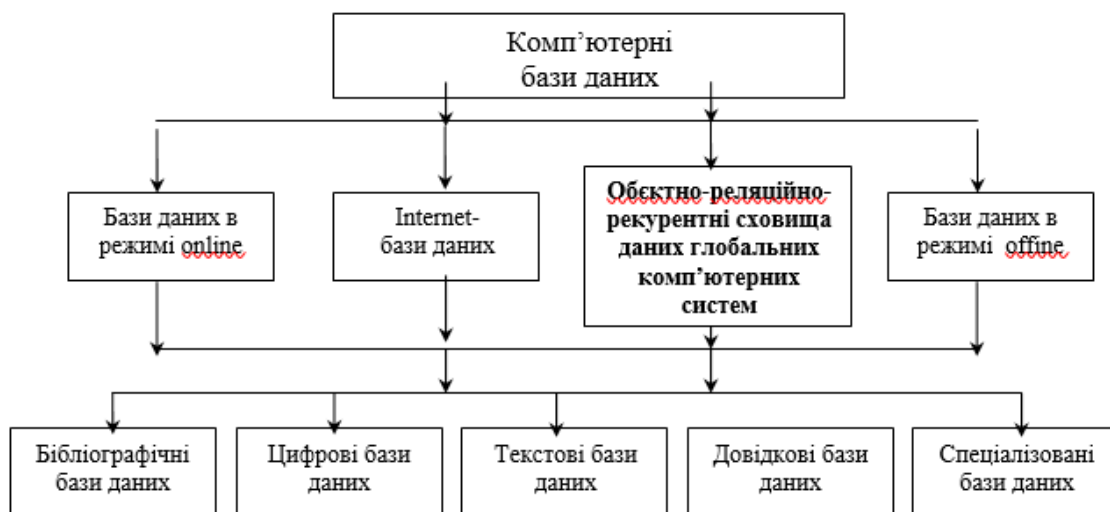


Рис.18.3. Класифікація існуючих та запропонованих комп'ютерних баз даних.

Комп'ютерні бази даних (Computers data base), діляться на бази даних з доступом в режимі online, Internet-бази даних, бази даних в режимі offline та запропоновані нові об'єктно-реляційно-рекурентні сховища даних глобальних розподілених комп'ютерних систем (рис.18.3).

Бази даних з доступом в режимі online (online databases) зберігаються в центральному банку даних. Доступ до них здійснюється посередністю комп'ютера (або іншого термінала) через телекомунікаційну мережу. Доступ, пошук і аналіз Internet-баз даних (internet databases) здійснюється посередністю Internet. А також доступ, пошук і аналіз баз даних глобальних комп'ютерних систем здійснюється посередністю глобальних комп'ютерних систем. Відомості із них можна завантажувати і зберігати на комп'ютері або допоміжним запам'ятовуючий пристрій.

Об'єктно–реляційно–рекурентні сховища даних глобальних розподілених комп'ютерних систем в теоретико–числовому базисі Галуа, є „цеглиною” в побудові лінійно–рекурентних, дворівневих, багаторівневих лінійно–рекурентних баз даних, які є в структурі вертикально–рекурентних, дворівневих, багаторівневих вертикально–рекурентних баз даних які будуть основою конгломератів епохальних сховищ даних глобальних розподілених комп'ютерних систем в теоретико–числовому базисі Галуа.

18.4. Аналіз ефективності кодування ідентифікаційних даних об'єктно-реляційно-рекурентної моделі сховищ даних глобальних розподілених комп'ютерних систем в ТЧБ Галуа .

Для розрахунку ефективності фізичної структури задамо початкові умови. Нехай число елементів об'єктно–реляційно–рекурентної моделі СД ГРКС в ТЧБ Галуа дорівнює $m=5$, число схематично показано у вигляді горизонтальних відгалужень рівних $m=4$, число вертикальних відгалужень рівне $m=4$. Відповідно позначимо m_i та m_n відгалужень. Розрахунок об'єму даних відомої моделі унітарного дерева виконано на основі формули Хартлі.

$$V_i = \hat{E}[\log_2 m_i]; i \in 1, n; \quad \sum_{i=1}^n V_i = \sum_{j=1}^k \hat{E}[\log_2 m_j] \quad (18.11)$$

де V_i - елемент коду, $\sum_{i=1}^n V_i$ - загальний об'єм кодів.

Визначаємо забезпечення умови емерджентності об'єктно-реляційно-рекурентної моделі сховищ даних глобальних розподілених комп'ютерних систем в ТЧБ Галуа:

$K_e = \frac{16}{5} = 3, 2$ відповідає умові $K_e < 4$, де K_e – коефіцієнт емерджентності, N_3 -число зв'язків-16, N_0 - кількість елементів БД -5.

Основними характеристиками оптимізації і підвищення ефективності БД є:

- здешевлення системи СУБД ;
- збільшення швидкості пошуку та доступу до даних;
- зменшення та стиснення об'єму інформації;
- розробка різноманітних типів пристроїв зберігання даних;
- тенденції росту системи;
- підвищення продуктивності системи;
- паралельні обчислення;
- розвиток моделей БД ГКС;
- підвищення надійності захисту від помилок;
- захист від несанкціонованого доступу.

Проведені дослідження показують, що перспективним є розвиток та вдосконалення нових моделей БД ГКС в базисі Галуа.

Дослідження теоретико–числових базисів та їх застосування при організації БД, показують, що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадоне захищеним ,крім того, традиційний метод є незахищеним від несанкціонованого доступу. В той же час дані характеристики БД ГКС достатньо ефективно реалізується в базисі Галуа.

Очевидно, що глибокі дослідження застосування теорії і методів кодування даних та організації БД в базисі Галуа є актуальними і перспективними.

Запропоновані моделі БД ГКС в базисі Галуа можуть бути ефективно застосовані для компактного завадозахищеного (таємного) кодування даних, наприклад, в міжнародних транснаціональних корпораціях глобальних комп'ютерних системах.

РОЗДІЛ 19

КОНГЛОМЕРАТИ ЕПОХАЛЬНИХ СХОВИЩ ДАНИХ В ТЕОРЕТИКО-ЧИСЛОВОМУ БАЗИСІ ГАЛУА

Конгломерати епохальних сховищ даних в базисі Галуа 2012 рік – на порозі „Золотого століття“

Під впливом сонячної активності (СА), яка є циклічною, відбуваються зміни історичних циклів. Починаючи з 2012 року ми вже проживаємо у "Золотому столітті". Початок у "Золотому столітті" – не період розквіту, а час відновлення: після багатьох катаклізмів минулого "Залізного століття".

Але, озираючись назад, дивуєшся точним прогнозам артилерійського генерала професора В.А.Мошкова і знаходиш їх підтвердження у сучасних вчених. Багато хто з них намагалися поєднати циклічність в історії з активністю сонця. Цим займався і наш співвітчизник А.Л.Чижевський. Він зауважив, що цикли сонячної активності змінюють всі процеси життєдіяльності, починаючи від урожаїв і закінчуючи хворобами, як фізичними, так і психічними - у всього людства.

Сонце змінює стан магнітосфери та атмосфери Землі. Магнітні поля та потоки часток, які йдуть від сонячних плям, досягають Землі та впливають перш за все на мозок, серцево-судинну та кровоносну системи людини, на її фізичний, нервовий та психологічний стан. Високий рівень сонячної активності, його швидкі зміни збуджують кожну людину, а тому й колектив, клас, суспільство, особливо, коли є спільні інтереси та зрозуміла і прийнятна ідея. У великому 11-річному циклі йдуть швидкі коливання рівня СА з періодом, близьким до 4 тижнів, пов'язані з власним обертанням Сонця навколо своєї осі та впливом Меркурія і Венери. Земля та "зовнішні" планети обумовлюють довгоперіодичні зміни СА. Враховуючи зміни полярності магнітних плям в кінці 11-річного циклу, вважають справжнім періодом СА 22-річний", котрий було названо за прізвищами їх першовідкривачів циклом Гнєвишева-Оля.

В результаті, ці цикли відображаються на історичних подіях –війнах, політико-економічних кризах, повстаннях, революціях і т. п.. Розглядаючи цикли сонячної активності, Сила тяжіння планет, тих, що зібралися по одну сторону від Сонця, помітно впливає на наше світило. На притягання планет Сонце реагує більшою кількістю викидів — протуберанців. А.Л. Чижевський, до речі, думав, що самі серйозні соціальні потрясіння, у тому числі війни і революції, доводяться саме на ті роки, коли кількість протуберанців на Сонце виявлялося збільшеним. цікаве підтвердження теорії генерала-професора Валентина Мошкова можна знайти в Інтернеті. Так виглядають графік сонячної активності по століттях, розрахований сучасними вченими (рис.19.1). III-тисячоліття – епоха, що складає суму 2,5 циклів на основі якої можна побудувати Конгломерати епохальних сховищ даних.

Знайдений новий спосіб заглянути в середину Сонця в 18.04.2014 р 20:18. Астрофізики з США і Канади запропонували техніку визначення гранулярної структури фотосфери Сонця, використовуючи данні про його магнітне поле.

Робота вчених опублікована в The Astrophysical Journal Letters, коротко з нею можна ознайомитися на сайті NASA.

Дані про магнітне поле світила було отримано обсерваторією сонячної динаміки. Астрофізики звернули увагу на відповідність між структурою магнітного поля на поверхні Сонця і розмірами і кількістю гранул в тій же області. Було виділено чотири типи масштабів процесів, перші два із яких зв'язані з грануляцією в фотосфері зірки. Астрофізики привели аналогію для свого методу: якщо нема можливості безпосередньо спостерігати жилі будинки і квартали, то їх розміри можна оцінити по кількості проживаючих в них пар чоловіків та жінок. Гранули — короткоживущі (до 20 хвилин) утворення у фотосфері Сонця, які утворюються в результаті конвекції плазми. Діаметр гранул досягає тисяч кілометрів, супергранул — десятків тисяч.

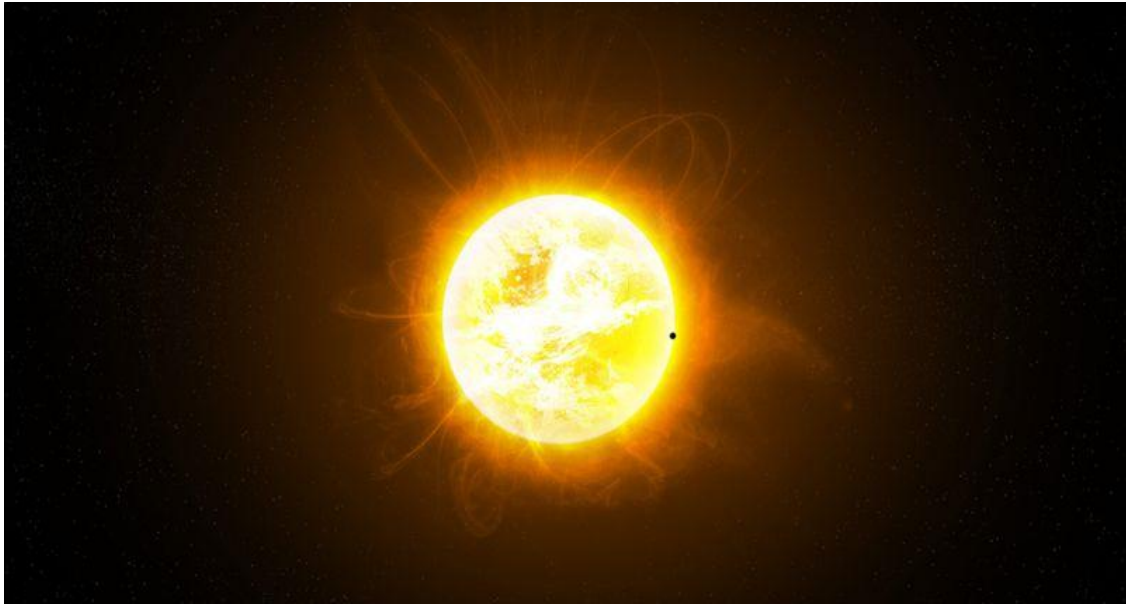


Фото 19.1. Грануляція в фотосфері на поверхні Сонця.

На основі теорії історичних циклів, які залежні від теорії сонячної активності, можна побудувати схему історичної циклічності конгломератів епохальних баз даних. Графік залежності історичних циклів від сонячної активності показаний на рис.19.2.

III-тисячоліття-епоха IV- історичний цикл. Так як цикли сонячної активності постійно повторюються, то можна скласти прогнозування з 2012 року.

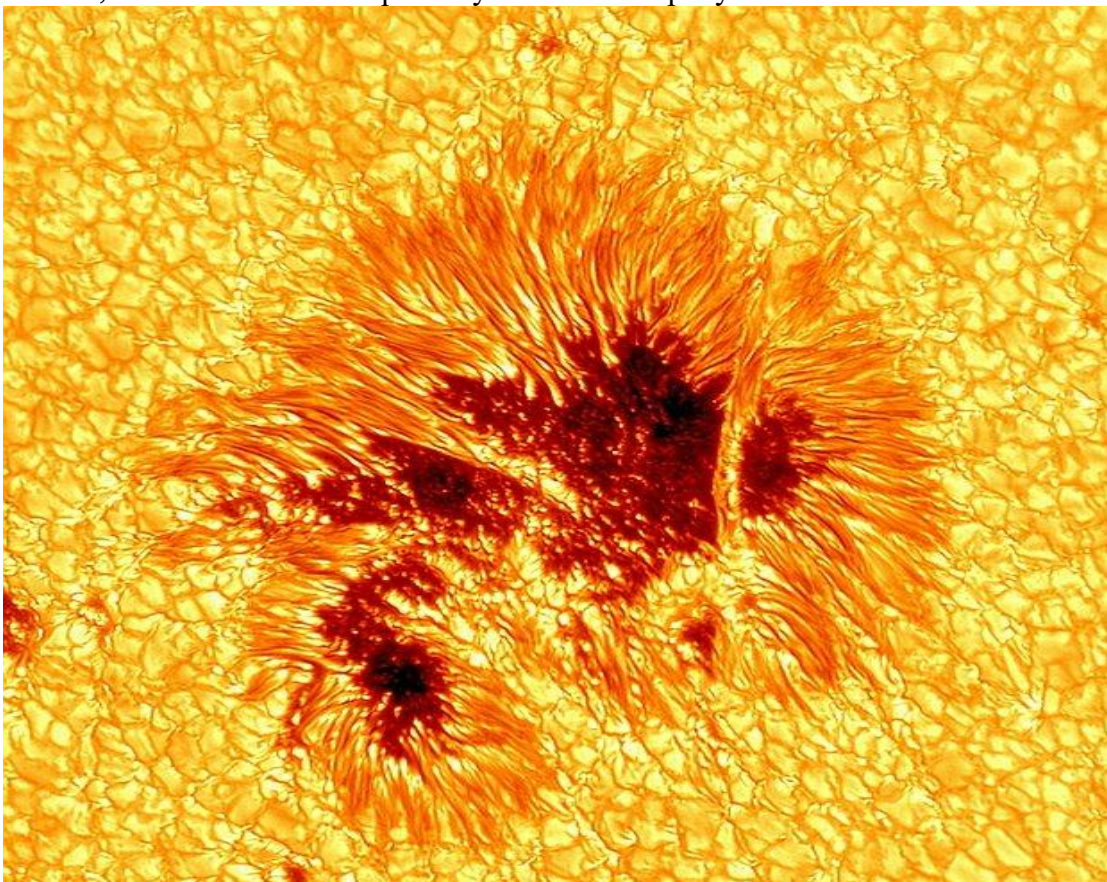


Фото 19.2. Перші такі, докладні зображення, як точну картину сонячних плям та гранул. Плями на поверхні зірки-Сонця. (Fot. New Jersey Institute of Technology / NST) JB 11.08.2013p 11:54

Перший такий, як точну картину сонячних плям дослідники з Нью-Джерсі технологічного інституту (NJIT) опублікували фотографії поверхні Сонця. Це перші зображення, як точно показує сонячних плям і впливу надзвичайно сильного магнітного поля зірки. Фотографії були зроблені новим сонячним телескопом у Big Bear Solar Observatory, яка фінансується, зокрема по NJIT. За визнанням проф. Венда Цао, нове покоління обладнання "дозволяє спостерігати сонячну атмосферу - від фотосфери до хромосфери - практично в режимі реального часу". Фото плями - темні області на поверхні зірки, яка має більш низьку температуру, ніж температура навколишнього магнітного поля і сильніше - це найбільш точне зображення, зняте досі. Іморталізованих пляма майже хрестоматійним прикладом цього явища, тому що - як зазначає проф. Цао - виглядає як ромашка з великою кількістю пелюсток. Новий телескоп Сонячний є найбільшим і найбільш точний сонячний телескоп. Його будівництво було завершено в 2009 році, але для цього детальні знімки фотосфери Сонця знадобилося чотири роки.

Сонячна активність

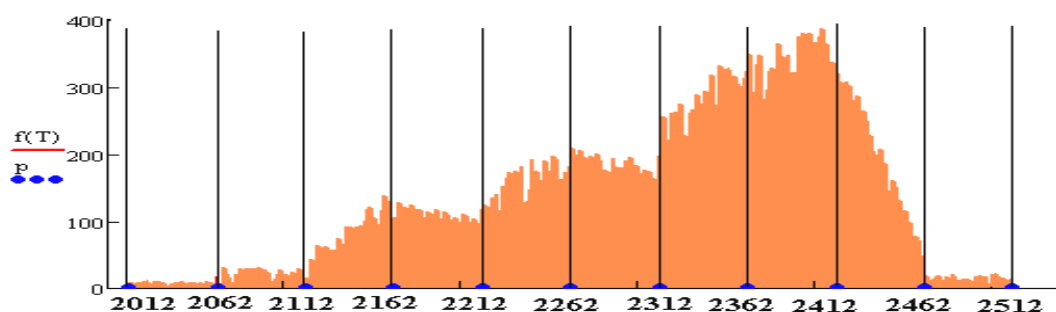


Рис.19.1. Величина часу (t) і кількість спалахів сонячних плям $f(T)$ на Сонці.

Де: t – величина часу яка поділяється на періоди 2012-2062 (50 років), століття 2012-2112 (100 років) і на цикли 2012-2512 (400 років);

$F(T)$ – Бали оцінювання по 10-бальній шкалі передбачаючого ступеня свого життєвого благополуччя в майбутньому. Через кожні 5 років — ступінь свого реального задоволення життям.

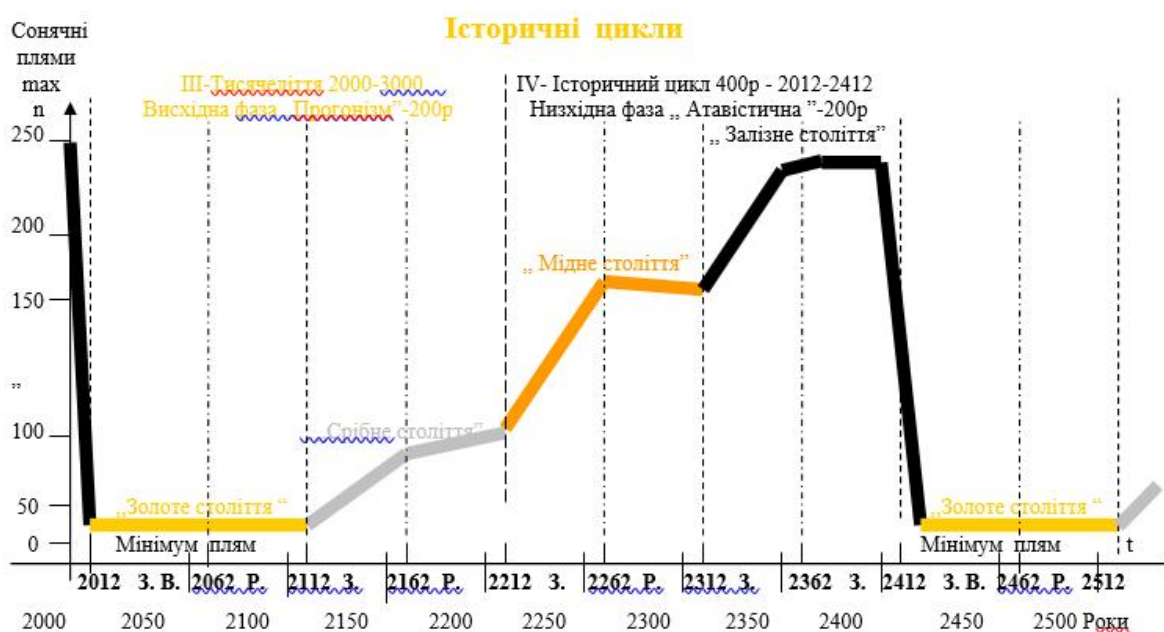


Рис. 19.2. Графік залежності історичних циклів від сонячної активності

„1-Золоте століття” –812 рік, 1212 рік, 1612 рік, 2012 рік – прогнозується появою нових і розквітом старих наук, мистецтва і культури, початком консолідації, «збиранням, зміцненням, відновленням і розквітом» держави Україна у Євросоюзі, зміцненням родинних, національних і політичних зв’язків, зростанням добробуту населення, промислового і сільськогосподарського виробництва.

„2-Срібне століття” – 2112 рік – прогнозується зміцненням та подальшим розквітом держави, яка є унітарна, демократична, єдиним і потужним національно-територіальним утворенням, зі строгою вертикальною владою, сильною ідеологією, яка базується на домінуючій у всіх сферах науки і мистецтва, сформованою національною ідеєю.

„3-Мідне століття”– 2212 рік– прогнозується початком занепаду держави, є початком недовіри до уряду, держави, батьківщини, співгромадян і навіть до членів власної сім’ї. У народу послаблюється відчуття патріотизма, а в самій державі pojawiaються відцентрові сили - воно починає проявлятися як стремління до роздрібнення на більш дрібні утворення. Поступово деградує наука і культура. Розпочинається боротьба за виживання. Виникає взаємна не приязнь між викладачами і учнями. Виникатимуть послаблення сімейних вуз, збільшиться кількість розпадаючих шлюбів, росте число беспризорних дітей. Улюбленим способом відпочинку складають азартна гра, пияцтво, розпуста. У громадян зникає чесність, а виникає брехня і обман. Раковою пухлиною в суспільстві розповсюджується заздрість і бажання обкрасти майно у іншого. Звідси і виникає ріст числа таких злочинів, як шантаж, вимагательство, шахрайство, пограбування та розбій. Бачучи безнадійність своєї служби, офіцери втрачають відчуття честі і повагу у солдат, починають тяжіти своїми обов’язками.

„4-Залізне століття ”– 2312 рік– прогнозується періодом „суцільного занепаду” і розпаду. Останнє завершальне століття циклу характеризується бунтами, революціями і бесконечними міжусібними війнами, світовими війнами, супроводжуваними розоренням країни, знущання над їхніми громадянами, голодом і епідеміями. Воно й зрозуміло, події «залізного століття» перепочинку не дають.



Рис.19.3. Схема будови конгломератів епохальних сховищ даних в базисі Галуа

Тому на основі теорії історичних циклів, які залежні від сонячної активності, можна побудувати історичні БД конгломератів епохальних сховищ даних в базисі Галуа (табл.19.1). Викладена схематична концепція „історичних циклів”, зображена проф. В.А. Мошковим в 1910 р. та підтверджена проф. А.Л.Чижевським.

Історична багатовимірна база даних: історія України – Руси

№ п / п	Число сонячних плям Sunspot Number, n	Назва циклів- 400 DATA, t	Назва півциклу-200	Назва століття- 100 Назва півстоліття-50	Історичні події України-Руси
	I- Тисячоліття			Стародавність	(до 300 н.е.)
1		I-Цикл-400	Раннє	Середньовіччя	(300-1240)
	Maunder Minimum	812-862	1-Висхідна фаза	Золоте - відновлення	Заснування РУСИ
	1-1	862-912	„Прогонізм”	Золоте - розквіт	Прийняття християнства
	5-100	912-962	1-Висхідна фаза	Срібне - занепад	Прихід печенігів на РУСЬ
	80-50	962-1012	„Прогонізм”	Срібне - розквіт	Хрещення РУСИ
	II - Тисячоліття				Розквіт РУСИ
	55-155	1012-1062	2-Нисхідна фаза	Мідне - занепад	Князювання Ярослава Мудрого
	135-125	1062-1112	“Атавістична”	Мідне - розквіт	Успішний похід князів на половців
	180-200	1112-1162	2-Нисхідна фаза	Залізне - занепад	Князювання Володимира Мономаха
	190-50	1162-1212	«Атавістична »	Залізне - занепад	Об'єднання Галичини і Волині
					Монголо-татарське їго
2		II-Цикл-400	Пізнє	Середньовіччя	(1240-1569)
	Maunder Minimum	1212-1262	1-Висхідна фаза	Золоте - відновлення	Падіння Києва
	1-1	1262-1312	„Прогонізм”	Золоте - розквіт	Приєднання Люблінщини і Закарпаття
	5-100	1312-1362	1-Висхідна фаза	Срібне - занепад	Завоювання Галичини польським королем Казимиром
	80-50	1362-1412	„Прогонізм”	Срібне - розквіт	Грюнвальдська битва
	55-155	1412-1462	2-Нисхідна фаза	Мідне - занепад	Приєднання до Польщі Західного Поділля
	135-125	1462-1512	«Атавістична »	Мідне - розквіт	Перемога війська кн.К.Острозьського над московським військом

	180-200	1512-1562	2-Нисхідна фаза	Залізне - занепад	Заснування Запорозької січі
	190-50	1562-1612	«Атавістична»	Залізне - занепад	Козацькі морські походи гетьмана П.К.Сагайдач -ного
3		III-Цикл-400		Козаччина	(1569-1775)
	Maunder Minimum	1612-1662	1-Висхідна фаза	Золоте-відновлення	Українсько-польська визвольна війна
	1-1	1662-1712	„Прогонізм”	Золоте - розквіт	Руїна на Правобержній Україні
	5-100	1712-1762	1-Висхідна фаза	Срібне - занепад	Полтавська битва
				Новий час	(1775-1917)
	80-50	1762-1812	„Прогонізм”	Срібне – розквіт	Руйнування Катериною II Запорізької січі
	55-155	1812-1862	2-Нисхідна фаза	Мідне – занепад	Кримська війна
	135-125	1862-1912	«Атавістична»	Мідне – розквіт	Заснування Львівської політехніки
				Новітній час	(1917-1991)
	180-200	1912-1962	2-Нисхідна фаза	Залізне - занепад	I і II світова війна
	190-50	1962-2012	«Атавістична»	Залізне - занепад	Проголошення незалежності України
	III - Тисячоліття			Сучасність	(після 1991)
4		IV-Цикл-400			
	Maunder Minimum	2012-2062	1-Висхідна фаза	Золоте - відновлення	Асоційоване членство України у Євросоюзі
	1-1	2062-2112	„Прогонізм”	Золоте - розквіт	Прогнозування
	5-100	2112-2162	1-Висхідна фаза	Срібне - занепад	Майбутнього
	80-50	2162-2212	„Прогонізм”	Срібне - розквіт	——//——
	55-155	2212-2262	2-Нисхідна фаза	Мідне - занепад	——//——
	135-125	2262-2312	«Атавістична»	Мідне - розквіт	——//——
	180-200	2312-2362	2-Нисхідна фаза	Залізне - занепад	——//——
	190-50	2362-2412	«Атавістична»	Залізне – занепад	——//——

5		V-Цикл-400			
	Maunder Minimum	2412-2462	1-Висхідна фаза	Золоте - відновлення	——//——
	1-1	2462-2512	„Прогонізм”	Золоте - розквіт	——//——
	5-100	2512-2562	1-Висхідна фаза	Срібне - занепад	——//——
	80-50	2562-2612	„Прогонізм”	Срібне - розквіт	——//——
	55-155	2612-2662	2-Нисхідна фаза	Мідне - занепад	——//——
	135-125	2662-2712	«Атавістична»	Мідне - розквіт	——//——
	180-200	2712-2762	2-Нисхідна фаза	Залізне - занепад	——//——
	190-50	2762-2812	«Атавістична»	Залізне - занепад	——//——
6		VI-Цикл-400			
	Maunder Minimum	2812-2862	1-Висхідна фаза	Золоте - відновлення	——//——
	1-1	2862-2912	„Прогонізм”	Золоте - розквіт	——//——
	5-100	2912-2962	1-Висхідна фаза	Срібне - занепад	——//——
	80-50	2962-3012	„Прогонізм”	Срібне - розквіт	——//——
	IV - Тисячоліття				
	55-155	3012-3062	2-Нисхідна фаза	Мідне - занепад	——//——
	135-125	3062-3112	«Атавістична»	Мідне - розквіт	——//——
	180-200	3112-3162	2-Нисхідна фаза	Залізне - занепад	——//——

19.1.Перспективні напрямки розвитку циклічних конгломератів епохальних моделей СД ГРКС у ТЧБ Галуа.

В епоху інформаційного суспільства щогодинно, у геометричній прогресії збільшується потік інформації, яку потрібно отримати, обробляти, передавати, використовувати та зберігати протягом годин, діб, місяців, пір року, років, десятиліть, періоди, століть, півциклів, циклів, епохи. . Час проходить по горизонтальній річній спіралі (колу–петлі), яка поступово щороку переходить у висхідну вертикаль наступного року (сегмент), тобто від минулого до майбутнього по вертикальній часовій спіралі. Виникає потреба створення нової структури бази даних, яка об'єднувала надвеликі бази даних– сховища даних в конгломерати епохальних сховищ даних у базисі Галуа. Поступає безліч інформації на протязі доби, яку із збільшенням об'ємів інформації стає все важче обробляти. Тому за допомогою паралельної та вертикальної обробки ця епохальна структура може бути ще однією ефективною стратегією обробки надвеликих об'ємів даних, розбиття даних на сегменти і побудови моделей для кожного сегмента окремо з подальшим об'єднанням результатів. В більшості об'ємів даних можна

виділити декілька відмінних один від одного підмножин. Це можуть бути, наприклад, групи інформацій, які є подібними і для яких доцільно будувати одну модель. В цьому випадку замість побудови однієї складної великої моделі для всіх, можна будувати багато простих моделей для кожного сегмента окремо. Подібний підхід дозволяє підвищити швидкість аналізу і знизити вимоги до пам'яті завдяки обробці менших об'ємів даних в один прохід. Крім того, в цьому випадку аналітичну обробку можна розпаралелити, що також позитивно впливає на витрачений час. До цього ж моделі для кожного сегмента можуть будувати різні аналітики. А при підвищенні швидкості цей підхід має ще одну важливу перевагу— багато простих моделей окремо легше створити і підтримувати, ніж одну велику. З місяцями, роками ці моделі будуть запускатися поетапно, отримуючи таким чином перші результати в максимально стислі терміни. Багатовимірні бази даних розглядають дані як куби, які є узагальненням електронних таблиць на велике число вимірів. Крім того, куби підтримають ієрархію вимірів і формул без дублювання їх визначень. Набір відповідних кубів складає багатовимірну базу даних - сховища даних.

19.2. Аналіз характеристики кодування моделей конгломератів епохальних сховищ даних в базисі Галуа у структурі світового господарства.

Конгломерат епохальних сховищ даних в базисі Галуа – нова абстракція керування даними. Логічна структура моделі конгломератів епохальних сховищ даних в базисі Галуа показана на рис.19.1.

Починаючи з 2012 року, розпочинається „Золоте століття” вузол–корінь „батьківський”, завершуючи в 2112 році–вузлом–вершиною утворюють ієрархічну структуру вузлів унітарного багаторівневого лінійно–рекурентного та багаторівневого вертикально–рекурентного часового дерева. А з 2112-2212 рік „Срібне століття”. З 2212-2312 рік. З 2312-2412 рік „Мідне століття”. З 2412-2512 рік „Залізне століття” завершується історичний 400-літній історичний цикл. А 2,5 цикла складають тисячоліття–епоху – конгломерати епохальних сховищ даних в базисі Галуа:

Багаторівнева лінійно-рекурентна модель бази даних в базисі Галуа

(вихідні дані: 24 години, 29-31день, 12 місяців, 1 рік–сегменти, 6,25 років,12,5-років, 25 років, 50 років-періоди, 100 років-століття, 200 років-півцикли, 400 років-цикли,1000 років-епоха (К.Е.С.Д. – Конгломерати епохальних сховищ даних).

- Однорівнева лінійно-рекурентна модель бази даних в базисі Галуа (0-24 години).
- Дворівнева лінійно-рекурентна модель бази даних в базисі Галуа (1-31 днів).
- Трьохрівнева лінійно-рекурентна модель бази даних в базисі Галуа (1-12-місяців).
- Чотирьохрівнева лінійно-рекурентна модель бази даних в базисі Галуа (1 рік-сегмент).
- Багаторівнева лінійно-рекурентна модель бази даних в базисі Галуа (6,25 років-сегментів-1/8 періода).
- Багаторівнева лінійно-рекурентна модель бази даних в базисі Галуа (12,5 років-сегментів-1/4 періода).
- Багаторівнева лінійно-рекурентна модель бази даних в базисі Галуа (25 років-сегментів-1/2 періода).
- Багаторівнева лінійно-рекурентна півстолітня модель бази даних в базисі Галуа (50 років-сегментів-період).
- Багаторівнева лінійно-рекурентна столітня модель бази даних в базисі Галуа (100 років-сегментів).

- Багаторівнева лінійно-рекурентна півциклічна модель бази даних в базисі Галуа (200 років-сегментів-півцикл).
- Багаторівнева лінійно-рекурентна циклічна модель бази даних в базисі Галуа (400років-сегментів-цикл).
- Конгломерати епохальних сховищ даних в базисі Галуа (1000 років-сегментів-2,5 цикла).

Таблиця. 19.2.

Багаторівневі лінійно-рекурентні моделі баз даних в базисі Галуа

№ п/п	Назва Багаторівневих лінійно-рекурентних моделей баз даних в базисі Галуа	Час збору, опрацювання і зберігання баз даних у роках	Структура БД	
1	Однорівнева лінійно-рекурентна модель БД БГ	0-24 години	—	
2	Дворівнева лінійно-рекурентна модель БД БГ	1-31 день	—	
3	Трьохрівнева лінійно-рекурентна модель БД БГ	1-12 місяців	—	
4	Чотирьохрівнева лінійно-рекурентна БД БГ	1 рік	сегмент	
5	Багаторівнева лінійно-рекурентна модельБД БГ	6,25	сегмент	1/8 періода
6	Багаторівнева лінійно-рекурентна модельБД БГ	12,5	сегмент	¼ періода
7	Багаторівнева лінійно-рекурентна модельБД БГ	25	сегмент	1/2 періода
8	Багаторівневалінійно-рекурентна -період БД БГ	50	сегмент	період
9	Багаторівнева лінійно-рекурентна—століття БД БГ	100	сегмент	століття
10	Багаторівнева лінійно-рекурентна-півцикл БД БГ	200	сегмент	півцикл
11	Багаторівнева лінійно-рекурентна-циклічна БД БГ	400	сегмент	цикл
12	Конгломерати епохальних сховищ даних в БГ	1000	сегмент	2,5 циклів

Багаторівнева вертикально -рекурентна модель бази даних в базисі Галуа

(Вихідні дані: 6,25 років, 12,5 років, 25 років, 50 років-період, 100 років-століття, 200 років-півцикл, 400 років-цикл, 1000 років-тисячоліття-епоха).

- Однорівнева вертикально-рекурентна модель бази даних в базисі Галуа (2 триади 3,125=6,25 років-касета-1/8 періода).
- Дворівнева вертикально-рекурентна модель бази даних (12,5 років-касета-1/4 періода).

- Трьохрівнева вертикально–рекурентна модель бази даних в базисі Галуа (25 років-касета-1/2 періода).
- Періодичні сховища даних в базисі Галуа (50 років-касета- період).
- Столітні сховища даних в базисі Галуа (100 років-касета)
- Півциклічні сховища даних базисі Галуа (200 років-касета-півцикл).
- Циклічні сховища даних в базисі Галуа (400 років-касета-цикл).
- Конгломерати епохальних сховищ даних в базисі Галуа (1000 років-касета - 2,5 цикли)

На основі наведених даних можна скласти таблицю 19.3.

Горизонтальна обробка сегментів баз даних. Обчислення проводяться за допомогою паралельної обробки, яка буде складатися з ще однієї ефективної стратегії обробки великих об'ємів даних. Розбиття даних на сегменти і побудова моделей для кожного сегмента окремо, з подальшим об'єднанням результатів, на основі двох і більше сегментів, яка дає можливість побудувати багаторівневу лінійно-рекурентну модель базу даних в базисі Галуа.

Таблиця 19.3

Багаторівневі вертикально-рекурентні моделі БД та СД в базисі Галуа.

№ п/п	Назва Багаторівневих вертикально-рекурентних моделей баз даних в базисі Галуа	Час збору, опрацювання і зберігання БД і СД у роках	Структу ра БД і СД	Форма БД і СД
1	Однорівнева вертикально-рекурентна модель БД БГ	6,25	касета	1/8 періода
2	Дворівнева вертикально-рекурентна модель БД БГ	12,5	касета	¼ періода
3	Трьохрівнева вертикально-рекурентна модель БД БГ	25	касета	1/2 періода
4	Багаторівневі вертикально-рекурентна модель БД БГ	50	касета	період
5	Столітні сховища даних в базисі Галуа	100	касета	століття
6	Півциклічні сховища даних в базисі Галуа	200	касета	півцикл
7	Циклічні сховища даних в базисі Галуа	400	касета	цикл
8	Конгломерати епохальних сховищ даних в базисі Галуа.	1000	касета	2,5 циклів

Вертикальна обробка касет сховищ даних (Вихідні дані (Сегменти 1-N), Моделі I –X Касети).

Результати. Побудова Конгломератів епохальних сховищ даних в базисі Галуа.

Логічна структура моделі циклічних конгломератів епохальних сховищ даних в базисі Галуа

Бази даних та сховища даних дозволяють опрацьовувати деталізовані інтегровані дані, що побудовані на основі наперед доступних моделей даних. Для побудови логічної моделі конгломератів епохальних сховищ даних в базисі Галуа потрібно перший етап розпочати зі створення добової однорівневої лінійно-рекурентної бази даних в базисі Галуа (0-24 год.)(1], яка потім ввійде до складу дворівневої лінійно-рекурентної бази даних в базисі Галуа (1-31 день) по зразку вкладених касет. У кожній добі погодинно можуть міститися 24 надвеликі бази даних - сховища даних, в які будуть поступати дані. В залежності від тривалості місяця. який

складається від 1 до 28, 29, 30, 31діб, можна побудувати дворівневу лінійно–рекурентну модель бази даних в базисі Галуа (1-31 день) [1,2], яка потім увійде до складу трьохрівневої лінійно–рекурентної моделі бази даних в базисі Галуа (1-12 місяців). На протязі 12 місяців можна побудувати чотирьохрівневу лінійно–рекурентну модель бази даних в базисі Галуа (0-1 рік). (Це буде перший сегмент (1рік) для полегшення паралельного обчислення. Такий об'єм інформації буде накопичений за рік) [2,3], яка увійде до складу багаторівневої лінійно–рекурентної бази даних в базисі Галуа (1-6,25 років, 6-12,5 років, 12,5-25 років, 25-50 років, 50-100 років, 100-200 років, 200-400 років, 400-1000 років) по зразку вкладених касет (рис.1–1). Обчислення проводяться за допомогою паралельної обробки, яка буде складатися з ще однієї ефективної стратегії обробки великих об'ємів даних. Розбиття даних на сегменти і побудова моделей для кожного сегмента окремо, з подальшим об'єднанням результатів, на основі двох і більше сегментів, можливість побудувати багаторівневу лінійно–рекурентну базу даних в базисі Галуа [2,3].

За 6,25 років накопичується 6,25 сегментів, які можна об'єднати і побудувати однорівневу вертикально–рекурентну базу даних в базисі Галуа (6,25 років–касета) [4], яка буде входити до складу дворівневої вертикально–рекурентної бази даних в базисі Галуа. За період 12,5 років–12,5 сегментів, увійдуть у дворівневу вертикально–рекурентну базу даних в базисі Галуа (12,5 років–касета) [5]. За період 25 років–25 сегментів, увійдуть до складу трьохрівневої вертикально–рекурентної моделі бази баних в базисі Галуа (25 років–касета). За період 50 років–50 сегментів увійдуть до складу чотирівневої вертикально–рекурентної моделі бази даних в базисі Галуа (50 років–період). При накопиченні 100 сегментів–100 років утворюється багаторівнева вертикально–рекурентна модель бази даних в базисі Галуа (100 років–століття, 200 років–півцикл, 400 років–цикл, 1000 років–тисячоліття- епоха) [6] (рис.1–2)–конгломерат епохальних сховищ даних в базисі Галуа.

Функції конгломератів епохальних сховищ даних в базисі Галуа реалізуються нагромадженням вихідних даних, побудовою річних сегментів за допомогою багаторівневих лінійно–рекурентних баз даних, які входитимуть у вертикальні моделі на базі багато–рівневих баз даних в базисі Галуа. Після збору вихідних даних сегментів, моделей отримують результати аналізів вертикальної обробки даних наступних періодів.

19.3. Принципи організації моделей циклічних конгломератів епохальних сховищ даних в базисі Галуа у структурі світового господарства

Теоретичні положення кодування даних у логічній моделі конгломератів сховищ даних в базисі Галуа побудовані на основі багаторівневої лінійно–рекурентної моделі бази даних та багаторівневої вертикально–рекурентної моделі бази даних. Такий підхід є досить ефективним та може забезпечити зменшення надлишковості кодування даних , які формуються , зберігаються і передаються міжнародними каналами зв'язку, на один– два порядки. При цьому, враховуючи відносно низьку вартість засобів зберігання даних на фізичних носіях, в діапазоні 10^6 – 10^{20} Гбайт основний ефект зниження собівартості руху даних в транснаціональних міжнародних корпораціях можуть досягти в каналах зв'язку локальні та глобальні комп'ютерні мережі та Інтернет, що при інтенсивному трафіку обміну даними 0 – 1 Гбайт/с може скласти сотні тисяч у.о. на рік.

19.4. Синтез циклічних конгломератів епохальних сховищ даних в базисі Галуа

Одним із вирішальних факторів забезпечення компактного та ефективного зберігання корпоративної інформації є вдосконалення сучасних технологій побудова баз даних. Одним із перспективних напрямків теорії організації баз даних є реалізація доступу до даних на основі нового синтезу конгломератів епохальних моделей сховищ даних в базисі Галуа. Конгломерати епохальних моделей сховищ даних в базисі Галуа, є новим типом структурної організації сховищ даних, яка формується на основі рекурентних властивостей теоретико-числового базису Галуа. Важливою характеристикою таких сховищ даних є максимальна компактність адресації даних, в тому числі при багаторівневій архітектурі. Позитивною характеристикою конгломератів епохальних моделей сховищ даних в базисі Галуа є можливості ефективного захисту від помилок та несанкціонованого доступу. При цьому особливістю використання базису Галуа при організації конгломератів епохальних моделей сховищ даних в базисі Галуа є можливість не тільки компактного кодування даних але і організований захист даних від помилок та несанкціонованого доступу. Тому дослідження ефективності є актуальною науково-технічною задачею.

19.5. Моделювання логічної структури циклічних конгломератів епохальних моделей сховищ даних в базисі Галуа.

При застосуванні базиса Галуа реалізується метод кодування елементів нової ієрархічної моделі бази даних в базисі Галуа. На її основі розроблені нові однорівневі, дворівневі і багаторівневі лінійно-рекурентні та однорівневі, дворівневі і багаторівневі вертикально-рекурентні моделі баз даних в базисі Галуа. На основі реляційної, розподіленої та комп'ютерної моделі бази даних реалізується нова реляційно-рекурентна модель бази даних глобальних розподілених комп'ютерних систем в базисі Галуа. Всі ці структури сховищ даних синтезуються у конгломерати епохальних сховищ даних в базисі Галуа та у майбутньому можна створити системи керування конгломератами епохальних сховищ даних в базисі Галуа.

Комп'ютерне моделювання циклічних конгломератів епохальних сховищ даних в базисі Галуа

На основі дослідження та аналізу характеристики кодування моделей циклічних конгломератів епохальних сховищ даних в базисі Галуа логічної структура моделі конгломератів епохальних сховищ даних в базисі Галуа можна побудувати комп'ютерну логічну модель конгломератів епохальних сховищ даних в базисі Галуа.

В результаті об'єднання лінійно-рекурентних, вертикально-рекурентних моделей баз даних та реляційно-рекурентної моделі бази даних глобальних розподілених комп'ютерних систем утворився синтез конгломератів епохальних сховищ даних в базисі Галуа.

На основі аналізу минулих історичних БД пов'язаної з історичною циклічністю в залежності від сонячної активності, яка повторюється через кожні 400 років, можна прогнозувати майбутнє наступних 400 років.

Довготривале зберігання обробленої інформації КЕСД в ТЧБ Галуа можна здійснювати на „Вічному” сапфіровому оптичному диску, створеного українськими вченими, інформація на якому може зберігатися десятки тисяч років.



Наша технологія досконала, оригінальна і не має аналогів у світі. Про це повідомив директор Інституту проблем реєстрації інформації, академик НАНУ Вячеслав Петров, 17 лютого 2014 20:16 передає [ZN.UA](http://zn.ua).

Запропонована та реалізована логічна структура моделі конгломератів епохальних сховищ даних в базисі Галуа, для дослідження та отримання результатів паралельної та вертикальної обробки. Модель конгломератів епохальних сховищ баз даних в базисі Галуа може бути застосована у кодуванні сховищ алфавітно-цифрових даних, кодуванню сховищ дворівневих зображень, кодуванню сховищ аудіо-відео інформації, застосуванню запропонованих КЕСД для збереження графічної інформації, організації КЕСД в ТЧБ Галуа для ведення комерційної інформації у глобальних розподілених комп'ютерних системах структури світового господарства на „Вічних” сапфірових оптичних дисках.

РОЗДІЛ 20

ФОРМУВАННЯ І ОРГАНІЗАЦІЯ БАЗ ДАНИХ РОЗПОДІЛЕНИХ КОМП'ЮТЕРНИХ СИСТЕМ

20.1.Порівняння типових БД з запропонованою організацією БД у ТЧБ Галуа

Складене порівняння існуючих типів моделей БД з запропонованими організаціями БД у ТЧБ Галуа в табл.20.1

Таблиця 20.1

Типові моделі баз даних

№ п/п	Типові моделі БД	Переваги моделі БД	Недоліки моделі БД
1	Ієрархічні	Досягли найвищої ефективності функціонування	Виразові можливості цих систем лишилися відносно низькими.
2	Мережеві	Значно кращі виразові можливості.	Низька ефективність функціонування
3	Реляційні	Найпростіша структура даних (таблиця, або плоский файл), високий рівень мов маніпулювання даними, виразові можливості максимально потужні.	Недоліки: неможливість адекватного відображення семантики предметної галузі, знизило ефективність функціонування
4	Об'єктні	Можливість для користувача системи визначити свої скільки завгодно складні типи даних (використовуючи синтаксис і властивості спадкування та інкапсуляції) Спадкування властивостей об'єктів.	Відсутність загальноприйнятої моделі даних; Відсутність єдиної формальної теорії; Відсутня активних експериментаторських дій. Відсутність строгих визначень, різне розуміння термінів і розходження в
5	Об'єктно – орієнтовані	Повторне використання програмного опису типів об'єктів при звертанні до інших типів, які на них посилаються	Термінології: Відсутність загальновикористовуваних стандартів, що дозволяють зв'язувати конкретні ООР-.
6	Об'єктно-реляційні	Поєднують у собі риси реляційної й об'єктної моделей. Їхнє виникнення породжене тим, що реляційні бази даних добре працюють із убудованими типами даних - можливість включення детальної семантичної інформації в БД;- можливість розширення базових типів даних, підтримка складних вкладених структур, визначення довільних структур (мультимедіа і т.ін.);- можливість багаторазового використання класів, прискорення розробки застосувань за рахунок наслідування.	Набагато гірше - з користувальницькими, нестандартними. Коли з'являється новий важливий тип даних, доводиться або включати його підтримку в СКБД, або змушувати програміста самостійно управляти даними в додатку.- відсутність достатньо проробленої теоретичної бази у об'єктної моделі;- складність структури;- навігаційне управління даними, складність

			додавання атрибутів і методів;- відсутність стандартної мови запитів.Об'єктно-реляційні бази даних
7	Напів-структуровані	Модель напівструктурованих даних відрізняється набагато більш високою гнучкістю представлення компонентів інформації. Найбільш яскравим практичним втіленням моделі напівструктурованих даних є XML (eXtensible Markup Language-росширювана мова розмітки).XML - це специфікація опису „документів”, що представляють собою набір вкладених елементів даних, функції яких визначаються відповідними тегами. Підтримка XML є найважливішим компонентом систем, здійснюючих посередницькі функції між різнорідними джерелами даних, служить цілям гнучкого представлення інформації в базах даних.	
8	Розподілені	У випадку розподілення користувачів між різними вузлами мережі Інтернету є можливими різні організації бази даних. Найпростіше рішення - це централізована база даних, де дані запам'ятовуються в одному вузлі мережі. В даному випадку ми можемо утримувати одну централізовану систему елементів управління даними. Опрацювання транзакцій та відновлення після відмови, базуються на відомих алгоритмах і протоколах надійності. Якщо транзакції, направлені від вузлів мережі до бази даних, стосуються всіх даних (істотно не залежать від вузла безпосередньо) то добрим є централізоване рішення. Прикладами таких систем є національні записи: компаній, людей, автомобілів.	Істотно тривалий час чекає результатів від видаленого вузла мережі - дуже довго у разі мережі або централізованої відмови вузла. Відсутність елемента управління над специфічними даними для даного розташування. Генерація великої кількості трафіку в мережі.
9	Багатовимірні	Багатовимірна база даних – база даних, що підтримує багатовимірну модель даних на концептуальному рівні та призначена для інтерактивного аналітичного опрацювання агрегованих історичних і прогнозованих даних. Основними поняттями багатовимірної моделі даних є: - гіперкуб даних rel	

		- вимір V, - атрибут A, - комірка X, - значення $rel(V,A)$.	
10	Куби даних	OLAP-куб — многомерный массив данных, как правило, разреженный и долговременно хранимый. Может быть реализован на основе универсальных реляционных СУБД или специализированным программным обеспечением	
11	Простори даних	Бази даних, і сховища даних дозволяють опрацьовувати деталізовані та інтегровані дані, що побудовані на основі наперед допустимих моделей даних. У випадку роботи у всесвітній мережі з величезною кількістю ресурсів (прикладом таких задач є туристичний бізнес – збирання інформацію про місця відпочинку, її інтеграція та зберігання у внутрішніх базах даних, геоінформаційні системи – на сьогодні ще не розроблено єдиних стандартів подання такої інформації, а її збір також проходить із джерел з наперед невідомими моделями даних) неможливо визначити, які саме моделі даних використовуватимуться. Т	
12	Надвеликі	Надвеликі бази даних Oracle (Very Large Database, надалі – VLDB) для досягнення високої продуктивності і високої доступності при низьких витратах на експлуатацію. Вона описує рішення вибору розміру блоку даних Oracle, застосування RAID-технологій, використання «лінійних» пристроїв (raw-devices), конфігурування журнальних файлів, розбиття табличних просторів на розділи, вибору параметрів зберігання та налаштування сегментів відкату. Стаття описує технології і пов'язані з ними обмеження, а також технічно детальні методи для оптимізації конфігурації в рамках цих обмежень	
13	Сховища даних	Основа для побудови систем підтримки прийняття рішень. Основна мета створення Data Warehouse в тому, щоб зробити усі значимі для управління бізнесом дані доступними в стандартизованій формі, придатними для аналізу та отримання необхідних	

		звітів. Для досягнення цього потрібно отримати дані із існуючих внутрішніх та зовнішніх, доступних для комп'ютера, джерел. Незважаючи на відмінності в підходах та реалізаціях, усім сховищам даних властиві такі спільні риси ^[2] : предметна орієнтованість, інтегрованість, прив'язка до часу, незмінність.	
14	Комп'ютерні	Комп'ютерні бази даних діляться на бази даних з доступом в режимі online, offline і через Internet (рис.1.17.). Бази даних з доступом в режимі online (online databases) зберігаються в центральному банку даних. Доступ до них здійснюється посередністю комп'ютера (або іншого термінала) через телекомунікаційну мережу. Доступ, пошук і аналіз Internet-баз даних (internet databases) здійснюється посередністю Internet. А також доступ, пошук і аналіз баз даних глобальних комп'ютерних систем здійснюється посередністю глобальних комп'ютерних систем [108,110,113]. Відомості із них можна завантажувати і зберігати на комп'ютері або допоміжним запам'ятовуючий пристрій. Описані види баз даних в свою чергу поділяються на бібліографічні, цифрові, текстові, довідкові і спеціалізовані.	

Таблиця 20.2

Запропонована організація моделей БД у ТЧБ Галуа

№ п/п	Запропоновані моделі БД у ТЧБ Галуа	Переваги моделі БД
1	Ієрархічні	Результати досліджень показали на основі проведених досліджень
2	Ієрархічно–рекурентна	ієрархічної бази даних, перевагу ТЧБ Галуа ефективнішого у 17,98
3	Ієрархічно–реляційна	рази над кодуванням ідентифікаційних даних ТЧБ Радемахера.
4	Однорівнева лінійно–рекурентна	На основі ієрархічно–рекурентної моделі БД у ТЧБ Галуа можна побудувати нову Однорівнева лінійно–рекурентну модель
5	Дворівнева лінійно–рекурентна модель	Побудована логічна структура нової ієрархічно–рекурентної моделі БД у ТЧБ Галуа.
6	Багаторівнева лінійно–рекурентна модель	Запропоновані багаторівневі лінійно–рекурентні моделі БД в базисі Галуа можуть бути ефективно застосовані для компактного завадозахищеного (таємного) кодування даних

7	Однорівнева вертикально-рекурентна	БД, нову вертикально-рекурентну модель БД та нову ієрархічно-реляційну модель БД, вони будуть основою для побудови інших баз
–8	Дворівнева вертикально-рекурентна	даних у ТЧБ Галуа і які можуть широко застосовуватися для освіти і науки України. Перевага вертикально-рекурентної моделі БД в базисі Галуа буде більш ефективна в 14,42 рази ніж кодування ідентифікаційних даних
9	Багаторівнева вертикально-рекурентна	унітарного дерева СЕС корпорації „ Pallada Travell ” USA. Проведені дослідження моделей БД показують, що перспективним є розвиток та досконалення нової вертикально-рекурентної моделі БД в базисі Галуа соціально-економічного структурування
10	Реляційні	На основі порівняння базиса Радемахера і базиса Галуа
11	Відношення	побудуємо відношення реляційних БД двомірних кодів в базисі Галуа
12	Реляційно рекрентної	Реляційно-рекурентна модель бази даних двомірних кодів в ТЧБ Галуа у вигляді Відношення реляційних баз даних в базисі Галуа
13	Об’єктно-реляційно-рекурентна	На основі моделі глобальної комп’ютерної системи, розроблена об’єктно –реляційно–рекурентна модель сховищ даних глобальних розподілених комп’ютерних систем в теоретико–числовому базисі Галуа, для побудови моделі та системи управління конгломератами епохальних сховищ даних в базисі Галуа.Широкому використанню моделей руху даних сприяє швидкий розвиток комп’ютерних систем найрізноманітнішого призначення. До важливого класу таких КС слід віднести системи реального часу, проблемно-орієнтовані комп’ютерні системи, низові комп’ютерні та малі локальні мережі, а також спеціалізовані комп’ютерні системи.
14	Конгломерати епохальних сховищ даних у ТЧБ Галуа	Запропонована та реалізована логічна структура моделі конгломератів епохальних сховищ даних в базисі Галуа, для дослідження та отримання результатів паралельної та вертикальної обробки. Модель конгломератів епохальних сховищ баз даних в базисі Галуа може бути застосована у кодуванні сховищ алфавітно-цифрових даних, кодуванню сховищ дворівневих зображень, кодуванню сховищ аудіо-відео інформації, застосуванню запропонованих КЕСД для збереження графічної інформації, організації КЕСД в ТЧБ Галуа для ведення комерційної інформації у глобальних розподілених комп’ютерних системах структури світового господарства.

20.2. Кодування та доступ до сховищ алфавітно-цифрових даних, дворівневих зображень, аудіо-відео інформації

Розробка критеріїв ефективності та дослідження системних характеристик клавіатур для вводу алфавітно-цифрових даних

При створенні мобільних програмно-апаратних засобів формування та дистанційного передавання СД важливою задачею є аналіз та дослідження ефективності процесів вводу алфавітно-цифрових даних оператором за допомогою абонентських пунктів, обладнаних клавіатурою та дисплеєм. Досвід розробки засобів вводу алфавітно-цифрових даних показує, що на практиці найбільшого поширення набули наступні класи пристроїв з вбудованими мікропроцесорами: мікрокалькулятори, мобільні телефони, мобільні комп'ютери (Noutbook) та спеціалізовані абонентські пункти. Прикладом клавіатури для спеціалізованих мобільних абонентських пунктів для вводу СД є синтезована клавіатура.

Важливими характеристичними параметрами ефективності формування ТЕД на основі клавіатур є:

- інформаційна ємність клавіатури – I ;
- загальне число клавiш – N ;
- число символних клавiш – N_S ;
- число функціонально-регістрових клавiш N_f ;
- відносні габарити по вертикалі та горизонталі – l, L ;
- число генерованих символів – N_0 ;
- число біт на один символ – b ;
- число символів на одній клавiші – S ;
- число натискань клавiші при формуванні одного символу – n .

Перелічені характеристики типових клавіатур названих чотирьох класів приведені в табл. 20.3.

Таблиця 20.3

Характеристики клавіатур вводу даних

Тип клавіатури	N	N_S	N_f	S	n	$l*L$
Калькулятор	26	11	15	1	2	4*6
Моб.телефон	20	10	10	8	4	6*3
Комп'ютер	86	44	42	4	2	6*16
Синтезована клавіатура	16	12	4	2	2,5	4*4

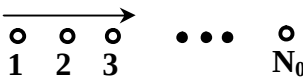
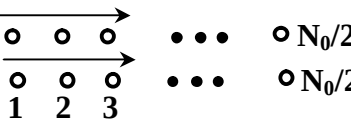
Для обґрунтування критерію ефективності клавіатури розглянемо можливі варіанти її реалізації в залежності від різного числа клавiш N та числа генерованих символів N_0 при використанні різних ТЧБ в процесі їх формування.

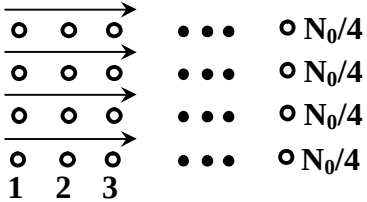
Потенційні характеристики клавіатур при використанні базису Хаара.

Нехай маємо 1 клавiшу ($N=1$), а число генерованих символів ($n=1024$) (табл. 20.4).

Таблиця 20.4

Параметри генерування символів в базисі Хаара

Характеристики клавіатури	Граф генерування символів в базисі Хаара	N
$N=1$		$\frac{N_0}{2n}$
$N=2$		$\frac{N_0}{4n}$

N=4		$\frac{N_0}{8n}$
...	...	...

З табл.20.4 видно, що при зростанні числа клавіш, які натискаються неодноразово, а символи генеруються клавіатурою згідно рівномірного розподілу, число натискань клавіш записується аналітично:

$$n = \frac{N_0}{2N}. \quad (20.1)$$

У випадку, коли, крім однократного натискання N клавіш, можна використовувати натискання одночасно двох клавіш:

- при N=4,

$$n = \frac{N_0}{4(N+1)}. \quad (20.2)$$

У випадку, коли, крім однократного натискання N клавіш, можна використовувати всі комбінації одночасного натискання N-i клавіш (i=0,1,2,...N-1)

$$n = \frac{N_0}{4(N+3)}. \quad (20.3)$$

Графіки необхідного числа натискань клавіш, розрахованих на основі виразів (20.1-20.3) приведені на рис.20.1 (1,2).

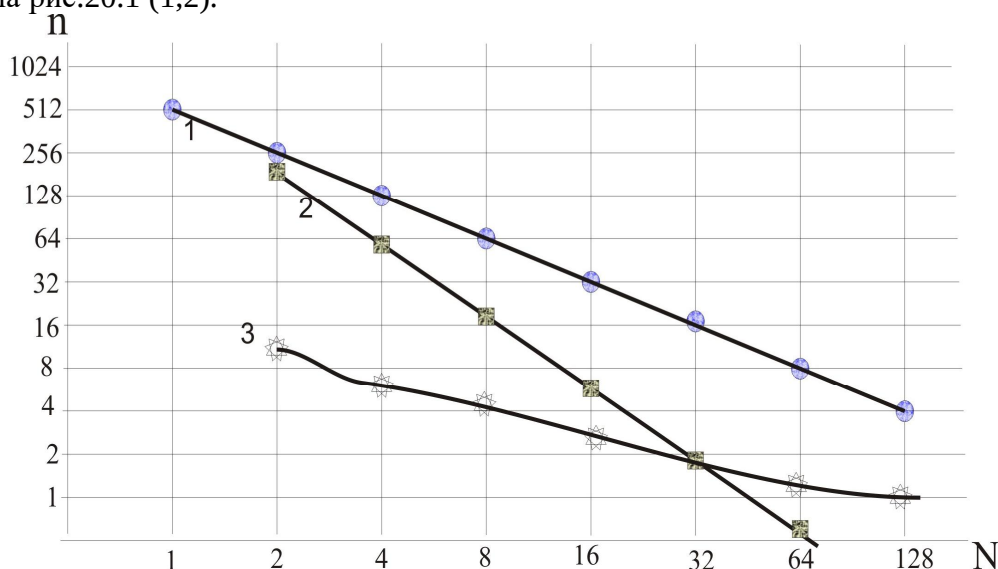


Рис.20.1. Залежність числа натискань клавіш від їх кількості на клавіатурі та числа генерованих символів у базисі Хаара.

При генерації символів на основі кодів базису Радемахера мінімальне число клавіш $N_0=2$. При цьому для генерації одного символу необхідно виконати $p=8$ натискань, що відповідає числу генерованих символів $N=256$. При $N=1024$ необхідно формувати 4 регістрові ознаки, що відповідає додатковому двократному натисканню клавіш ($p=10$). На рис.20.1 дана характеристика представлена графіком 3.

В якості критерію ефективності клавіатури застосований мультиплікативно-адитивний вираз:

$$k = \frac{N_S \cdot 8 + N_f \cdot \alpha}{l \cdot L \cdot S \cdot n}, \quad (20.4)$$

де α - коефіцієнт інформативності функціональних клавіш $\frac{1}{N_f} \leq \alpha \leq 1$.

На основі аналітичного виразу (20.4) при різних значеннях α розрахований коефіцієнт ефективності реалізації чотирьох класів клавіатури, які використовуються в калькуляторах, мобільних телефонах, комп'ютерах та запропонованих спеціалізованих засобах вводу СД на основі синтезованої клавіатури (патент України на корисну модель № 25291.- 2007р) (рис.20.2).

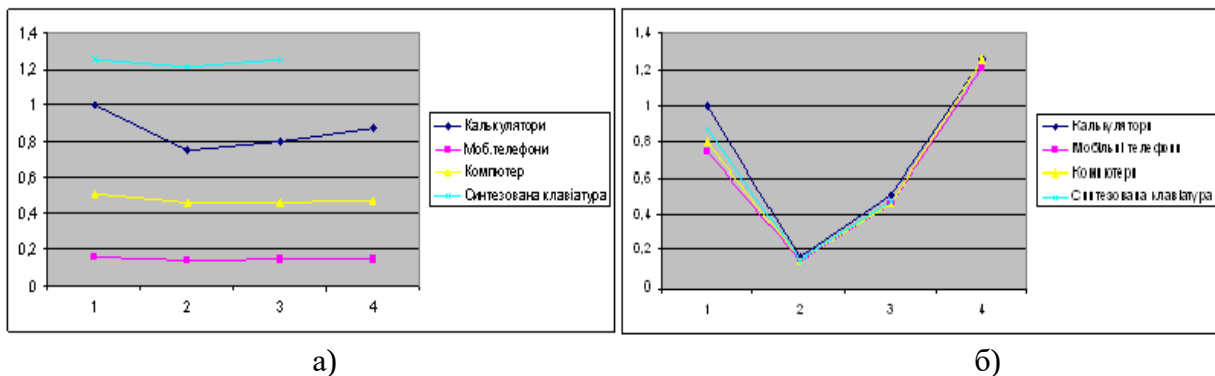


Рис.20.2. Оцінка коефіцієнта ефективності реалізації різних класів клавіатур.

Розрахунок коефіцієнта навантаження на одну клавішу клавіатури виконаємо згідно виразу:

$$K_i = \frac{N_0}{N_j}, \quad (20.5)$$

де $N_j = N_S + N_f$.

Графік коефіцієнта K_i приведено на рис. 20.3.

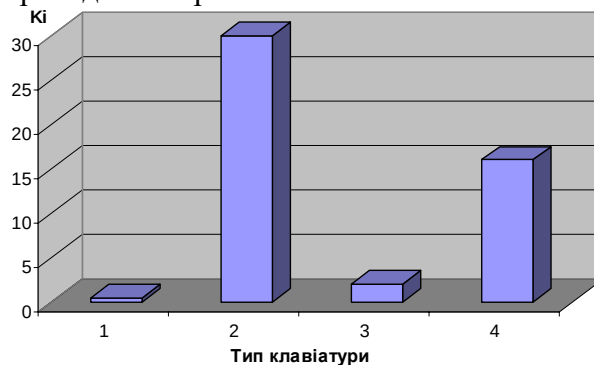


Рис.20.3. Коефіцієнт навантаження на одну клавішу різних класів клавіатур: 1- калькулятори, 2 – мобільні телефони, 3 – комп'ютери, 4 – синтезована клавіатура.

З рис.20.3 видно, що найбільше інформаційне навантаження на одну клавішу реалізовано в мобільних телефонах, наближена до оптимальної дана характеристика для запропонованої синтезованої клавіатури, низька - для калькуляторів та стандартних клавіатур комп'ютерів.

Загальну ефективність клавіатур різних класів оцінимо на основі мультиплікативного критерію згідно виразу:

$$P_k = \frac{1}{S} \cdot N_0 \cdot K_i,$$

в якому ефективність клавіатури для вводу алфавітно-цифрових даних ТЕД та СД пропорційна числу генерованих символів, значенню коефіцієнта ефективності, розрахована згідно коефіцієнта ефективності (20.4) і обернено пропорційна числу необхідних натискань клавіші для формування коду одного символу (табл.20.5).

Таблиця 20.5

Розрахункові характеристики ефективності різних клавіатур

Тип клавіатури	S	N ₀	K _i	P
Калькулятор	1	10	0,42	4,2
Мобільний телефон	6	256	30	64
Комп'ютер	2	1024	2,05	11,9
Синтезована клавіатура	2	256	16	128

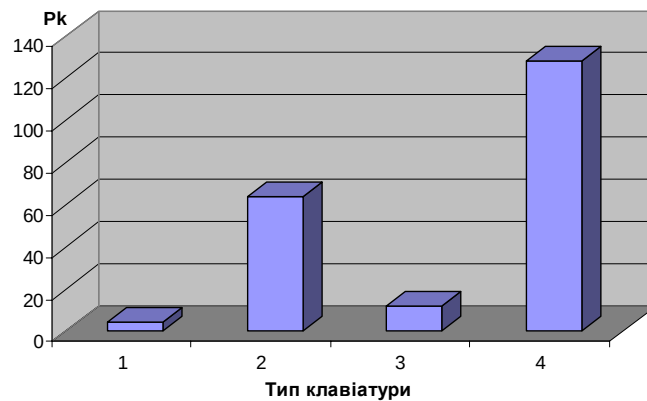


Рис.20.4. Коефіцієнт ефективності різних типів клавіатур:

1- калькулятори, 2 – мобільні телефони, 3 – комп'ютери, 4 – синтезована клавіатура.

Проведений аналіз критеріїв ефективності та характеристик різних класів клавіатур показує, що запропонована синтезована клавіатура для формування СД на низових рівнях РКС є найбільш ефективною при числі генерованих символів $N_0=256$, загальному числі клавіш $N_j=12+4$ та числу натискань клавіш при формуванні одного символу $S \leq 2$, тобто $P_{k\max} = 128$.

В РКС формування структуризованих даних (СД) виконується у вигляді фреймів, які класифікуються за призначенням використання:

- фрейм оператора низового рівня;
- фрейм системи передавання даних;
- фрейм бази даних;
- фрейм управління об'єктами.

У табл.20.6 приведені структури названих фреймів.

У табл.20.6 використані наступні параметри фреймів: атрибути обміну потоками даних між вузлами КС: start, stop – границі інформаційного файлу фрейма-оператора, Ф – границі пакету даних системи СПД, Т – реальний час, N – номер об'єкта, S – тип виконуваної операції, X – масив технологічних даних, М – сукупність інформаційних моделей об'єкта, L - сукупність логіко-статистичних інформаційних моделей, Сі, Сj – коди станцій КС, яка передає і приймає дані, ТЕД – техніко-економічні дані, які формуються оператором, І – ентропійні моделі та характеристики об'єкта, Y – команди управління, G – готовність виконання команди управління, V – дозвіл виконання команди управління, W – підтвердження виконання команди

управління на об'єкті. Приведені структури фреймів реалізуються на різних рівнях РКС згідно стандартних інтерфейсів, протоколів обміну даними, структури файлів БД, структури файлів використовуваних БД, а також стандартизованих протоколів спеціалізованих комп'ютерних систем.

Таблиця 20.6

Фрейми РКС

№ п/п	Тип фрейма	Структура фрейма
1	F оператора	
2	F системи передавання даних	
3	F бази даних	
4	F управління	

Крім приведених в табл.20.6 об'єктів, в проблемно-орієнтованих КС на рівні фреймів формуються СД, які охоплюють класи статистичних, кореляційних, спектральних, кластерних, сукупність моделей руху даних та інші моделі. При цьому інформаційна технологія формування ТЕД сумісно з технологічними моделями об'єктів є однією з найважливіших задач підвищення ефективності організації та зниження собівартості руху даних в КС.

Викладені теоретичні основи оптимізації характеристик клавіатур для вводу алфавітно-цифрових даних, а також запропоновані принципи кодування та синтезованого вводу алфавітно-цифрових даних є основою для створення нових апаратних засобів формування СД та реалізації мобільних адаптерів на основі використання сучасних процесорних платформ.

20.3. Організація БД в ТЧБ Галуа для ведення комерційної інформації в ГРКС

Потужний розвиток сучасних інформаційних систем потребує відповідного розвитку теоретичних засад реалізації бази даних (БД) та бази знань (БЗ). Особливо великі масиви даних формуються, підлягають зберіганню та оперативно циркулюють в каналах зв'язку міжнародних корпорацій, до яких належить міжнародна китайська корпорація „ТЯНЬ ШІ”.

Аналіз організаційної структури БД, якими користуються великі міжнародні корпорації показує, що найбільш широкоживаною для кодування ідентифікаційних даних є багаторівнева ієрархічно-реляційна структура. При цьому очевидно є велика надлишковість кодування та їх низька завадозахищеність, де на основі ієрархічних структур обґрунтовані принципи та показана ефективність кодування ІД на основі багатомірної рекурентної організації БД в базисі Галуа. Наприклад, подаємо схему діяльності інфраструктури корпорації „Тянь Ші” зображена на рис.20.5.

1.Аналіз характеристики кодування ІД в ієрархічно-реляційній БД структурі багаторівневого маркетингу корпорації „ТЯНЬ ШІ ”

Функції корпорації „ТЯНЬ ШІ” реалізуються п'ятьма ієрархічними рівнями дистриб'юторів на основі тетраедного дерева, що забезпечує структуру дистриб'юторської організації у складі 1024 осіб. Дистриб'ютор 4*- зірок І – рівня (наставник) традиційно в 1-рівень підписує нових 4 заявників (нових дистриб'юторів1*), на 2 –рівні –16 дистриб'юторів, 3 – 64 дистриб'ютори, 4 – 256 дистриб'юторів, 5 – 1024 дистриб'юторів, при цьому число рівнів не перевищує п'яти показана в табл.20.7. Маркетинг-план мережевого маркетингу корпорації „ТЯНЬ ШІ” на конференції 6-річчя „ТЯНЬ ШІ” 03.08.2001року признаний найкращою моделлю багаторівневого маркетингу в світі. Тому авторами розроблена і вперше запропонована в західному регіоні „ТЯНЬ ШІ-Україна” багаторівнева ієрархічно-реляційна структура БД. Як найбільш ефективна в роботі мережевого маркетингу корпорації показує застосування ієрархічно-рекурентної організації БД в базисі Галуа є найбільш ефективною та може забезпечити зменшення надлишковості кодування даних, які формуються, зберігаються і передаються по міжнародних каналах зв'язку більше, ніж на один - два порядки.

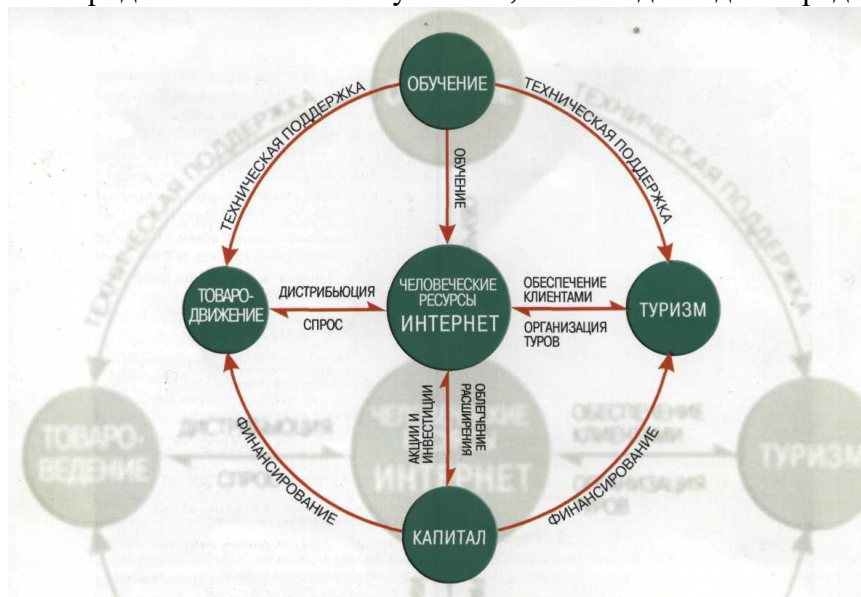


Рис.20.5. Циклічний граф шести мереж міжнародної корпорації „Тянь Ші”.

Таблиця 20.7

Багаторівнева ієрархічно-реляційна структура БД корпорації „Тянь Ші”-Україна

Ступінь Зірка*	Звання дистриб'юторів	Кількість дистриб'юторів із званням	Кількість дистриб'юторів по рівнях	Частотність, %
8* 5	Лідер дистриб'юторської організації	1	1	0,98
7* 4	Лідер дистриб'юторської структури	4	16	1,56
6* 3	Керівник офісу	16	48	4,68
5* 2	Верхній наставник	64	129	12,98
4* 1	Наставник	156	345	33,69
3* 0	Дистриб'ютор	420	285	27,83
2* 0	Споживач	280	105	10,25
1* 0	Заявник	83	95	9,27
Всього:		1024	1024	100

Характеристика частотності $F(r)$ розраховується у відсоткових значеннях реального числа дистриб'юторів на кожному рівні R^* від їх загального числа, що наближається до логарифмічного Гаусівського розподілу згідно рис.20.6 .

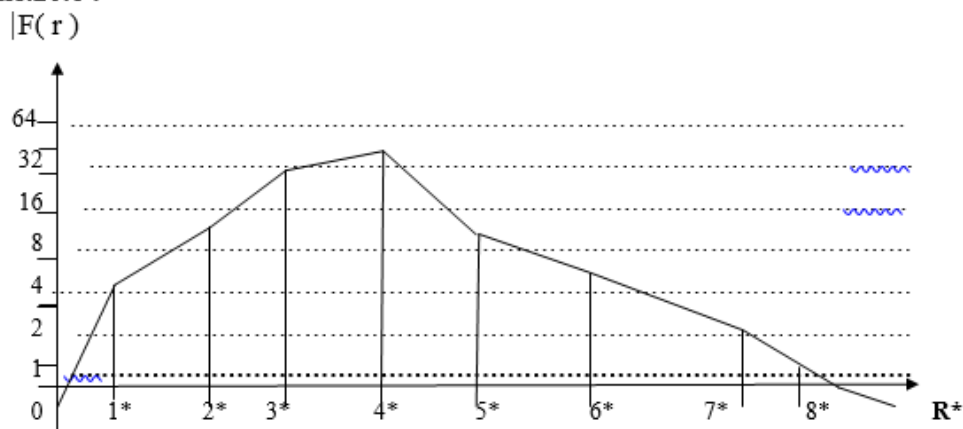


Рис.20.6. Характеристика розподілу числа дистриб'юторів $F(r)$ на різних рівнях * структурного дерева корпорації „Тянь Ші”.

На рис.20.7 відображена ієрархічна структура корпорації „Тянь Ші” системи ідентифікації дистриб'юторської організації багаторівневого маркетингу- (БРМ), яка належить до класу дерев зі зв'язаними списками, де: n -корінь –президент міжнародної китайської транснаціональної корпорації „Тянь Ші”, $n_1, \dots, n_k$ підкорені –віце-президенти корпорації, почесні члени правління корпорації, ЗЛ6*-, „Золоті льви шісти зірок”, $T_1, \dots, T_L$ дерева- ЗЛ5* - виконавчі директори, які є менеджерами вищого рівня.

ЗЛ4* -ЗЛ- піддерева –„Золоті льви чотирьох зірок”, СрЛ -„Срібні льви”, міжнародні лектори, БрЛ – „Бронзові льви” -керівники середніх підрозділів, національні лектори, які є менеджерами середнього рівня. $k_1, \dots, k_k$ - вузли- дистриб'ютори 8*- зірок, регіональні лектори, 7*-лідери дистриб'юторських організацій, 6*-майстри, утримувачі офісів, завідувачі майстерень, відділів, 5*-бригадири, керівники –координатори груп, 4*- ланкові, керівники груп, які є менеджерами нижчого рівня.

3*- наставники, які працюють по спонсорській програмі, 2*- дистриб'ютори , які працюють по торговій програмі, - дистриб'ютори 1* зірки, заявники, учні, які працюють по споживчій програмі. $l_1, \dots, l_k$ - листя - кандидати в дистриб'ютори , споживачі.

Наприклад, в Україні існують 5 регіональних представництв корпорації „ТЯНЬ ШІ ” (східний, північний, центральний, південний та західний), які охоплюють усі області України.

В табл.20.8 подана структура ієрархічно-реляційної організації ІД корпорації, яка включає наступні дані : НП-номер підприємства, НКД –ідентифікаційний номер дистриб'ютора, ФІОД –ПІБ, дистриб'ютора, НСД–ідентифікаційний номер наставника , ФІОСД –ПІБ, наставника, СТЕ- ступінь, ПОН -відстань до верхнього наставника , ЛЗ - особисті закупки , ОМЗ -товарообіг місячних закупок, СО -структурний об'єм , ЛНО - особистий нагромаджувальний об'єм , ОНОС - основний нагромаджувальний об'єм структури.

Таблиця 20.8

Середньостатистичні параметри даних ієрархічно –реляційної БД п'ятирівневої системи БРМ

НП	НКД	ФІОД	НСД	ФІОСД	СТЕ	ПОН	ЛЗ	ОМЗ	СО	ЛНО	ОНОС
6-X	8-X	24-X	8-X	24-X	1-X	2-X	3-X	5-X	5-X	9-X	12-X
...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...
6-X	8-X	24-X	8-X	24-X	1-X	2-X	3-X	5-X	5-X	9-X	12-X

{ ІД
{ ОД

де: X – байт, ІД – ідентифікаційні дані дистриб'ютора , ОД – організаційні дані дистриб'ютора.

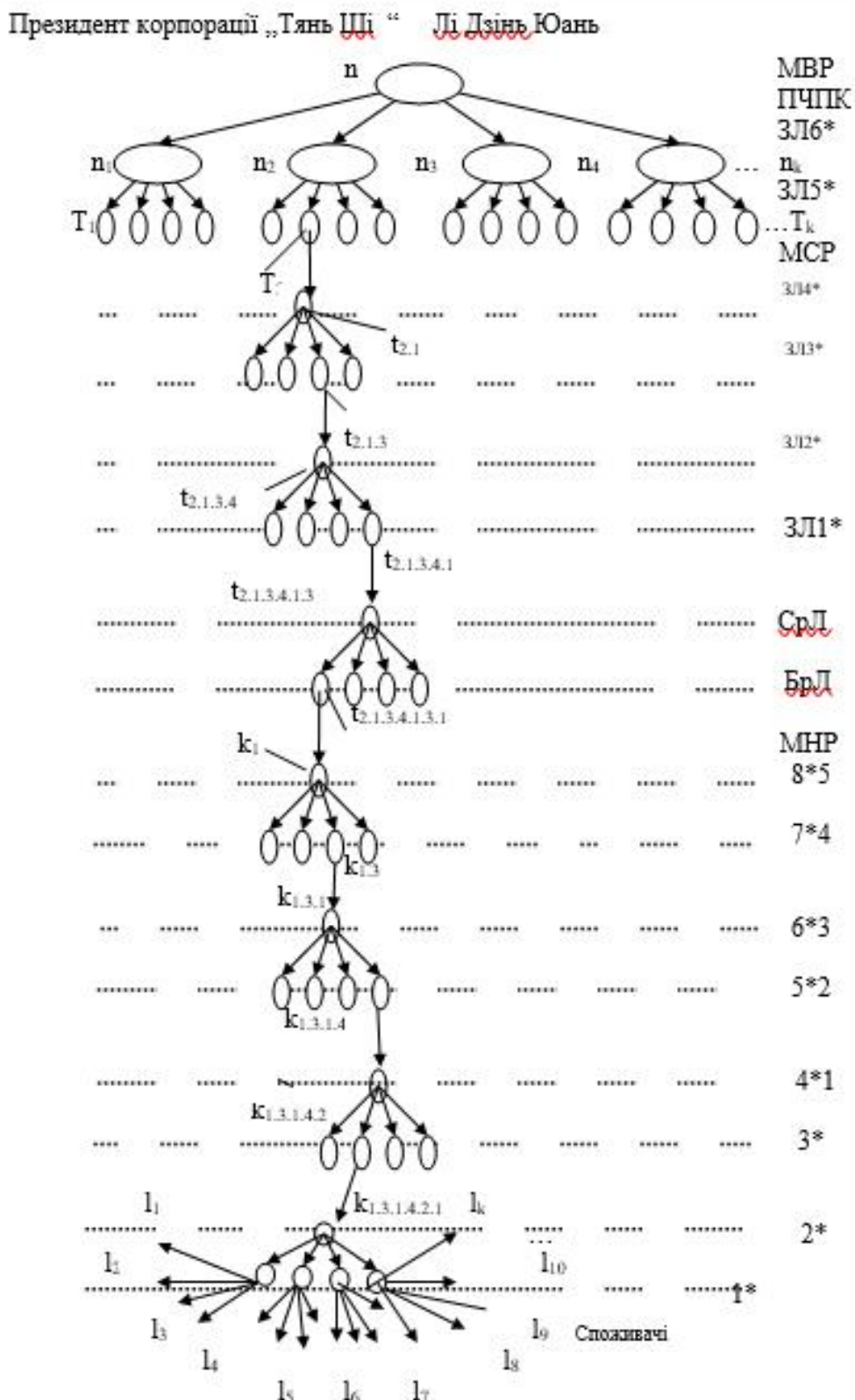


Рис.20.7. Ациклічний граф. Велике дерево Т- організаційної структури корпорації „Тянь Ші”.

В табл.20.8 символ $n \cdot X$ - означає число алфавітно-цифрових даних в кожній позиції двомірного файла ($n = 1, 2, \dots$). Таким чином при заданій кількості користувачів можна розрахувати об'єм алфавітно-цифрових даних одного кортежу БД корпорації „ТЯНЬ ШІ”.

Для зберігання даних кортежа реляційної БД, виходячи з умови байт-орієнтованого кодування, загальний об'єм даних може бути обчислений згідно формули:

$$I_j = \sum_{j=1}^m \sum_{i=1}^k d_{ij}, \quad (20.7)$$

де d_{ij} - об'єм даних ij - го атрибута кортежу ($i=1,2,\dots, k$) - число атрибутів, ($j = 1,2,\dots, m$) - число кортежів.

Аналіз табл.20.8 показує, що об'єми оперативних даних (ОД) діяльності дистриб'юторів визначаються доменами : ЛЗ, ОМЗ, СО, ЛКО та ОНОС. , а ідентифікаційні дані описуються доменами : НП, НКД, ФІОД, НСД, ФІОСД, СТЕ та ПОН. Таким чином ефективність кодування БД ієрархічно-реляційної архітектури, можна розрахувати за формулами відносної та приведеної оцінки:

$$K_{e_1} = \frac{I_{ID}}{I_{OD}}; \quad K_{e_2} = \frac{I_{ID} - I_{OD}}{I_{OD} + I_{ID}} \cdot 100\%. \quad (20.8)$$

Ефективність кодування БД ієрархічно-реляційної архітектури, яка відповідає корпорації „ТЯНЬ ШІ ” можна розрахувати за формулою (20.9):

$$I_{OD} = 34 \cdot X; \quad I_{ID} = 73 \cdot X. \quad (20.9)$$

Підставляючи дані у вирази формули отримуємо значення коефіцієнта ефективності у вигляді .

$$K_{e_1} = 2,147; \quad K_{e_2} = 36,4 \%;$$

що відповідає більш ніж двохкратній надлишковості кодування (ІД) по відношенню до (ОД) або 36,4 % відносній надлишковості кодування (ІД).

На рис.20.8 відображена ієрархічна структура менеджерів нижчого рівня корпорації „Тянь Ші”.

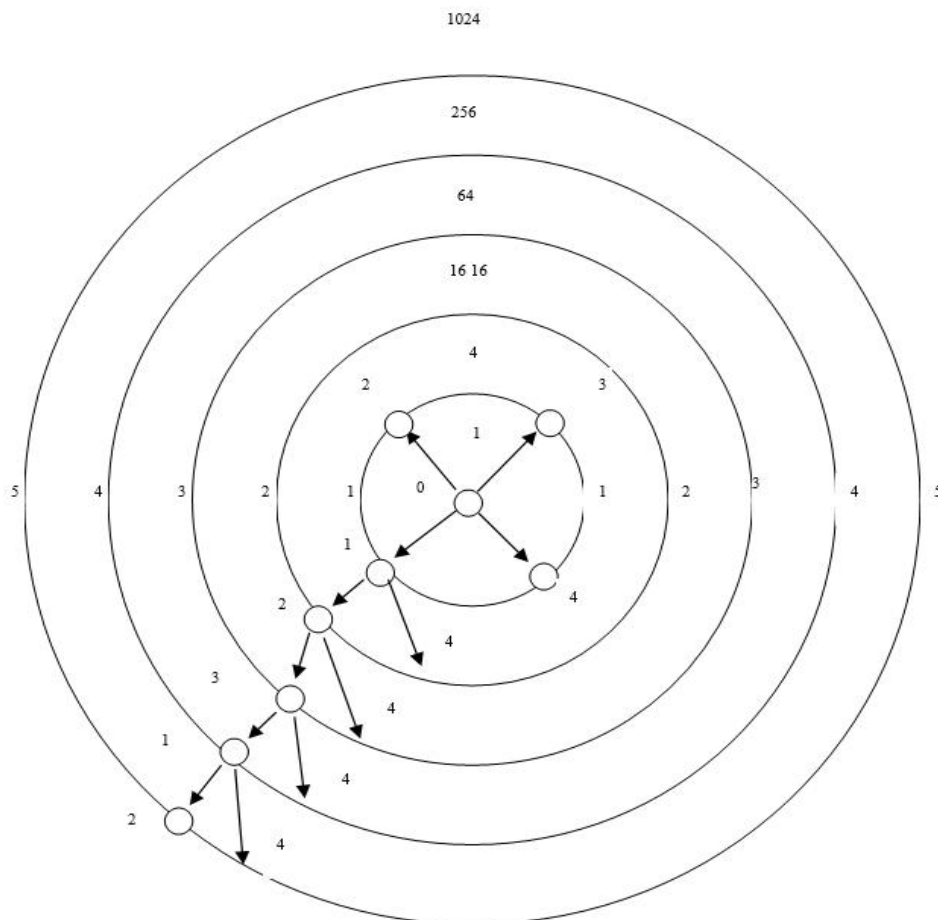


Рис.20.8. Ациклічний граф. Вузли $k_1, \dots, k_k$ Дерева T - Ієрархічна структура менеджерів нижчого рівня дистриб'юторів $8 \times 5 - 4 \times 1, 3^*, 2^*$, та $l_1, \dots, l_k$ - листів -заявників 1^* зірок, споживачів $1-1024 \dots$

Практично у корпорації „Тянь Ші” використовується універсальна система кодування ІД у вигляді двох 8 – розрядних десяткових кодів, які ідентифікують код наставника і дистриб’ютора в будь-якій структурі і підрозділі в будь –якій країні світу на ринку яких працює корпорація „Тянь Ші”. Це складає 64 біт двійково- десяткового коду, або символічно-байтового стандартного коду. При цьому об’єм бази ідентифікаційних даних менеджерів нижчого рівня корпорації „Тянь Ші” складає $11200 \times 128 = 1433600$, а загальна кількість можливих ідентифікованих членів складає 2^{64} осіб дистриб’юторів і споживачів населення планети Земля.

Для визначення певної ланки (наприклад дистриб’юторської організації №1) структури від лідера до дистриб’ютора і заявника згідно рисунка відслідковуємо шлях : 0 – 1 – 2 – 3 – 1 – 2 - ... тобто лідер – дистриб’ютор підписує в свою групу 1 – 4 - .. нових заявників, які дублюють своїх наставників. Таким чином , наприклад, відслідковуючи шлях присвоюємо номер дистриб’ютору 1 – рівня – код 01, 2 – рівня – код 012, 3 – рівня – код 0123, 4 – рівня - код 01231, 5 – рівня – код 012312, новому заявнику – код 0123124. Відповідно з десяткової системи числення переходимо на кодування даних в двійковій системі числення в базисі Галуа, що показано в табл.4.

Якщо взяти окрему структуру дистриб’юторської організації (вузла k_1) і починаючи з лідера і $1^* 0$ -рівня та пройти шлях до $8^* 5$ - рівня зафіксувавши шлях та згідно порядку підписання контракту та розташуванні на певному рівні у групах буде присвоєний відповідний номер , який можна закодувати в базисі Галуа. Відкриваючи код ієрархічної структури по його номеру, відкривається реляційна БД всієї структури кожного дистриб’ютора світу корпорації „ТЯНЬ ШІ”.

20.4. Принципи організації багаторівневої рекурентної БД в базисі Галуа

Рекурентно-реляційна організація БД, яка еквівалентна ієрархічно-реляційної організації, що використовується в корпорації „ТЯНЬ-ШІ” представлена в табл.20.9.

Таблиця 20.9

Кодування символів дистриб’юторів БРМ та їх структур

№1 Д*	Ідент.№д истри- б’ютора	П.І.Б. дистри- б’ютора	Ідент.№ наставника	П.І.Б. наставника	Ключі - коди	Символи- коди	КодиГа луа
8*5	8·X	24·X	8·X	24·X	0	0	00
7*4	8·X	24·X	8·X	24·X	01	.1	01
6*3	...	...	...	...	012	..2	10
5*2	...	...	...	...	0123	...3	11
4*1	...	...	...	...	01231	1	01
3*	8·X	24·X	8·X	24·X	012312	2	10
2*	8·X	24·X	8·X	24·X	0123124	4	00
1*	8·X	24·X	8·X	24·X			...

Числовий аналіз кодування ідентифікаторів НП –11200·X, показує, що число символів, які потрібно зберігати в БД корпорації „ТЯНЬ ШІ” про код країни та ідентифікаційний номер дистриб’ютора дорівнює –11200 X. При кодуванні аналогічних даних на основі рекурентної БД в базисі Галуа маємо $208 \cdot X \cdot 300$.

Звідки коефіцієнт надлишковості кодування ІД в БД корпорації „ТЯНЬ ШІ” згідно виразу складає : $K_{e_1} = \frac{11200}{208} = 53,85$.

Відповідно коефіцієнт надлишковості кодування ідентифікаційних кодів дистри - б’юторів в першому випадку складає : $24 \cdot X$, а в другому випадку: $11 \cdot X$.

Результати розрахунків кодування ідентифікаційних даних по рівнях дистриб'юторської організації на основі рекурентної БД у базисі Галуа відображено в табл.20.10, де G_i - послідовність бітів коду Галуа згідно певного ключа, а $\overline{G_i}$ - інвертовані біти Галуа, які вказують початок ієрархічного підрівня БД.

Таблиця 20.10

Кодування ідентифікаційних даних на основі рекурентних кодів поля Галуа.

XXX	0	1	2	3	4	5	Код Галуа
...0	$8 \cdot X$						$GGGGGG...G \overline{G} GGGGG$
	...						
...1	$8 \cdot X$	$8 \cdot X$					$GGGGG...GG \overline{G} GGGGG$
		...					
...2		$8 \cdot X$	$8 \cdot X$				$GGGG...GGG \overline{G} GGGGG$
			...				
...3			$8 \cdot X$	$8 \cdot X$			$GGG...GGGG \overline{G} GGGGG$
				...			
...4				$8 \cdot X$	$8 \cdot X$		$GG...GGGGG \overline{G} GGGGG$
					...		
...5					$8 \cdot X$	$8 \cdot X$	$G...GGGGGG \overline{G} GGGGG$
						...	
XXX						$8 \cdot X$	...

В 21 столітті корпорацією планується реалізація 80% роздрібною торгівлі в мережевому маркетингу і 20% в класичній роздрібній торгівлі. Завдяки мережевому п'ятирівневому маркетингу тетраїдних груп дистриб'юторських організацій компанії вперше пропонувано ієрархічно-рекурентну організацію БД в базисі Галуа.

Проведений аналіз оцінки ефективності кодування ІД та ОД на прикладі корпорації „ТЯНЬ ШІ“ показує що застосування ієрархічно-рекурентної організації БД в базисі Галуа є найбільш ефективними та може забезпечити зменшення надлишковості кодування даних, які формуються, зберігаються і передаються по міжнародних каналах зв'язку більше на один два порядки. При цьому враховуючи відносно низьку вартість засобів зберігання даних на фізичних носіях, які в діапазоні 10^6 - 10^{20} Гбайт основний ефект зниження собівартості руху даних в транснаціональних міжнародних корпораціях можуть досягти в каналах зв'язку локальні та глобальні комп'ютерні мережі та Інтернет, що при інтенсивному трафіку обміну даними (0 – 1 Гбайт/с) може скласти сотні тисяч у.о. на рік.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Black, U. The V Series Recommendations: Standards. New York: McGraw-Hill, 1995.
2. Grynchyshyn T. Methods of Manipulation of Signals in Optical Channels of Communications. // Матеріали міжнародної конференції TCSET'2004. – Львів-Славське, 2004. С.363-365.
3. Акушский И.Я., Юдицкий Д.М. . Машинная арифметика с остаточных классов. М.: Советское радио, 1968, - 440с.
4. Ахо А. и др. Структуры данных и алгоритмы. Пер. с англ. – М.: Вильямс, 2001. – С. 384.
5. Бабак В.П. , Хандецкий В.С., Шрюфер Е. Обробка сигналів. – Київ: Либідь, 1996. – 320 с.
6. Блейхаут Р. Теория и практика кодов, контролирующих ошибки / Блейхут Р. - М.: Мир, 1986. — С. 576.
7. В.В. Шаряк, Я.М. Николайчук. Системні характеристики баз даних та перспективні напрямки їх розвитку. // Моделювання та інформаційні технології. – Київ, ІЕД НАН України. - №1(1), 2007. - С.38-45.
8. В.В. Шаряк, Я.М. Николайчук. Теоретико-числові бази та їх застосування при організації бази даних. // Оптико-електронні інформаційно-енергетичні технології. - Вінниця, ВНТУ. - №2(12), 2006. - С.59- 66.
9. В.В. Шаряк Методи кодування елементів ієрархічних моделей баз даних в базисі Галуа // Наукові вісті ІМЕ,,ГА''. - Івано-Франківськ. - №2(10), 2006. – С.18-23.
10. В.В.Шаряк. Архітектура і кодування баз даних на основі теоретико-числових базисів // Вісник ТДТУ ім. І.Пулюя. - Тернопіль - Том.12, №1, 2007. –С.171-179.
11. В.В.Шаряк. Методи дослідження системних характеристик моделей баз даних // Вісник ТДТУ. - Тернопіль. - №2(13). 2008.-с.116-121.
12. В.В.Шаряк. Принципи рекурентного кодування ідентифікаційних даних у базисі Галуа // Поступ в науку. - Бучач. - Том 1. №4, 2008 – С.63-64.
13. Гарсия-Молина и другие. Системы баз данных. - М.: Вильямс, 2004. – 1088с.
14. Дейт К. Введение в системы баз данных. –М.: Вильямс ,2005. – 1328 с.
15. Дж.Мартин. Организация баз данных в вычислительных системах. – М.: „Мир”, 1980. – 662 с.
16. Жураковський Ю.П. Теорія інформації кодування: Підручник / Ю.П. Жураковський, В.П. Полторак/ -К.: Вища шк., 2001.-255с.
17. Залмазон Л.А. Преобразование Фурье, Уолша, Хаара и их применение в управлении, связи, и других областях. - М.: Наука, 1989. – 496 с.
18. Заставний О.М. Аналіз системних характеристик спецпроцесорів формування вихідних даних аналого-цифрових кодерів // Вісник Технологічного університету Поділля, Хмельницький, 2005, №4, ч.1, Т2. С. 223-226.
19. Заставний О.М. Синтез та проектування аналого-цифрового кодера автономного сенсора з вихідними двовірними шумоподібними сигналами // Збірник наукових праць інституту проблем моделювання в енергетиці ім. Г.Є. Пухова. - Київ,2006. – С.34 – 42.
20. Теорія та принципи побудови спецпроцесора на основі базисів Радемахера, Крестенсона, Галуа / О.М. Заставний, Я.М. Николайчук, Н.Д. Круцкевич, Р.І. Король // Тези доповідей сьомої міжнародної науково - технічної конференції – Вінниця: «УНІВЕРСУМ – Вінниця», 2003 – С.114.
21. Зюко А.Г. и др. Помехоустойчивость и эффективность систем передачи информации. – М.: Радио и связь, 1985. - 272с.
22. Игнатов В.А. Теория информации и передачи сигналов/ Игнатов В.А./ - М.: Радио и связь, 1991. - 280с.
23. Кислов В.Я. и др. Корреляционные свойства шумоподобных сигналов, генерируемых системами с динамическим хаосом // Радиотехника и электроника. 1997. Том 42. С. 134.
24. Кларк Д. Кодирование с исправлением ошибок в системах цифровой связи/Д. Кларк., Д. Кейн/: Пер. с англ. М.: Радио и Связь,1987. - 300 с.

25. Кузьмин И.В. Кодирование и декодирование в информационных системах / Кузьмин И.В., Ключко В.И., Литвин В.А. / К.: Вища школа, 1985. - 320с.
26. Мартин Дж. Введение в организацию баз данных. - М.: Мир, 1980. - 662 с.
27. Мейер Д. Теория реляционных баз данных. - М.: Мир, 1987. - 328с.
28. Мельничук С.І. Декодування двійкових циклічних М-послідовностей методом збіжності за ключем/ Мельничук С.І./ – Львів, 2001. - №37. – С.122 – 126.
29. Мельничук С.І. Методи динамічного декодування одновимірних М-послідовностей / С.І. Мельничук / Методи та прилади контролю якості. – Івано- Франківськ, 1998. - №2. – С.66 – 68.
30. Мельничук С.І. Декодування одновимірних циклічних М- послідовностей методом адаптивного сходження // Автоматизація виробничих процесів у машинобудуванні та приладобудуванні/ Мельничук С.І., Таянов С.А. /– Львів. – 2001. - №36. – С.117 – 121.
31. Міжнародна транснаціональна корпорація „Pallada travel”.co.usa. <http://www.pallada travel.com.us>
32. Морзе Н.В. Методика навчання інформатики. Навчальний посібник / Н.В. Морзе / - Київ, 2003. - 287с.
33. Мошков В.А. Происхождение человека./ В.А. Мошков / Варшава. 1907. Т.1.
34. Мошков В.А. Земля в объёмах Солнца / В.А. Мошков / Варшава., 1910. Т.2.
35. Николайчук Я. М. Принципи ідентифікації реляційних баз даних на основі двовірних кодів поля Галуа / Я.М. Николайчук, В.В. Шаряк // Хмельницький // Вісник ХНУ. – Том 1. № 2, 2007. – С.53- 59.
36. Николайчук Я. М. Принципи ідентифікації реляційних баз даних на основі двовірних кодів поля Галуа/ Я.М. Николайчук, В.В. Шаряк/ Хмельницький. - Вісник ХНУ. - Том 1, №2.
37. Николайчук Я.М. Архітектура багаторівневої лінійно - рекурентної структури бази даних в базисі Галуа / Я.М.Николайчук, В.В. Шаряк // Матеріали XII наукової конференції ТДТУ ім. І.Пулюя, 2008. – С.118.
38. Николайчук Я.М. Проектування спеціалізованих комп'ютерних систем. Навчальний посібник./ Я.М. Николайчук, Н.Я. Возна, І.Р. Пітух / -Тернопіль: ТзОВ „Терно- граф”, 2010. – 392с.
39. Николайчук Я.М. Теорія джерел інформації. Монографія. Видання друге, виправлене / Я.М.Николайчук. - Тернопіль: ТзОВ „Терно-Граф”, 2010. - 563 с.
40. Николайчук Я.М., Заставний О.М. Дослідження системних характеристик двовірних кодів з особливими кореляційними властивостями // Вісник технологічного університету Поділля, Хмельницький, 2004, №2, ч.1, Т2. С. 107-110.
41. Николайчук Я.М., Король Р.І. Вертикальна інформаційна технологія в базисі Галуа – новий напрямок в розвитку комп'ютерних машин. Матеріали симпозіуму: Сучасні проблеми в комп'ютерних науках (CCU- 2000). – Львів. - С.36-38.
42. Николайчук Я.М., Шаряк В.В. Проблеми захисту даних від помилок при передачі їх по каналах зв'язку / Я.М. Николайчук, В.В. Шаряк / Матеріали 5-міжнародної конференції з оптико-електронних інформаційних технологій „Photonics-ODS 2010”. - ВНТУ Вінниця, Україна. - С.48-49.
43. Николайчук Я.М. Методи стиснення даних в багатоканальних системах на основі кодів Галуа / Я.М. Николайчук, Н.Г. Яцків // Вісник національного університету „Львівська політехніка”. Радіотехніка та телекомунікації. - Львів. - №443. - С.135-138.
44. Николайчук Я.М., Шаряк В.В. Дослідження ієрархічно–рекурентної моделі бази даних в базисі Галуа / Матеріали міжнародної науково–практичної конференції „актуальні питання теорії та практики неперервної ступеневої підготовки фахівців в системі вищої школи ТДТУ ім. І.Пулюя, 2012 - С. 127.
45. Николайчук Я.М. Шаряк В.В.. Дослідження ієрархічно–реляційної моделі бази даних в теоретико–числовому базисі Галуа / Матеріали – III Науково– технічної конференції. Інформаційні моделі, системи та технології. ТНТУ ім. І.Пулюя. 24 квітня 2013. – С.26.

46. О. Погонєць, Я.М. Николайчук Методи визначення ентропії джерел інформації. – Хмельницький: Вісник Хмельницького національного університету, 2007, № 2, Т.1(90). - С.93–99.
47. Пасічник В.В. Організація баз даних та знань. Підручник для ВНЗ / В.В. Пасічник ., В.А. Резніченко/ Київ. Видавництво ВНУ, 2006. – 360 с.
48. Петришин Л.Б. Николайчук Я.Н., Ищеряков С.М., Цифровая обработка сигналов на основе преобразования кодов поля Галуа / Методы и микроэлектронные средства цифровой обработки и преобразования сигналов.– Рига: ИЭВТ АН Латвии, 1989. – С.130 – 132.
49. Петришин Л.Б. Перетворення форми та цифрової обробки інформації в кодових системах базису Галуа. – Івано-Франківськ: ІФНТУНГ МОУ, 1997. - 236с.
50. Петришин Л.Б. Теоретичні основи перетворення форми та цифрової обробки інформації в базисі Галуа: Навч. посібник. – Київ.: ІЗіМН МОУ, 1997. – 237 с.
51. Питерсон У. Коды, исправляющие ошибки/ У. Питерсон, Э. Уэлдон/ Пер. с англ. М.: Мир, 1976. - 600 с.
52. Погонєць І.О., Воронич А.Р. Метод побудови ентропійної моделі квазістаціонарного джерела інформації. Праці міжнародного симпозиуму: Питання оптимізації обчислень, 2009. – Т2. – С.214-219.
53. С.В. Глушков, Д.В. Ломотько. Базы данных. - Х.: Фолио АСТ, 2002. – 320 с.
54. Скляренко С.М. Поштовий зв'язок: Підруч.для вищ.навч.закл.для спец. за напрямом „Телекомунікації”/ С.М.Скляренко, В.К.Стеклов, Л.Н.Беркман / за заг.ред. В.К.Стеклова.-[2-ге вид.,стереотип.]-К.: Техніка, 2004.-904с.
55. Стеклов В.К. Теорія електричного зв'язку. Підруч. для вищ.навч. зал.для спец.за напрямом „Телекомунікації”/ В.К. Стеклов, Л.Н. Беркман / за заг. ред. В.К. Стеклова. - К.: Техніка. 2006.-552с.
56. Столлингс В. Передача данных. 4-е издание – СПб:Питер, 2004 – 750 с.
57. Стратонович Р.Л. Теория информации / Стратонович Р.Л./ - М.: Сов. радио, 1975.- 420с.
58. Т.М. Гринчишин. Кодування даних в комп'ютерних розподілених системах з відкритим оптичним каналом зв'язку на основі рекурентних ГК-кодів.// Вісник Технологічного університету Поділля.Хмельницький – 2007. –Т1 ч.1, №2. – с. 74-77.
59. Теория кодирования / Кассами Т., Токура Н., Ивадари Е, Инагаки Я.. - пер. с яп. - М.: Мир, 1978. - 576 с.
60. Цымбал В.П. Основы теории информации и кодирование/ В.П. Цымбал / - К.: Вища шк., 1992. - 263 с.
61. Шаряк В.В. Дослідження вертикально-рекурентної моделі бази даних в базисі Галуа. Матеріали міжнародної науково-технічної конференції: фундаментальні та прикладні проблеми сучасних технологій/ В.В.Шаряк./– Тернопіль: ТНТУ. Україна, 2010. – С.326–327.
62. Шаряк В.В. Моделювання лінійно-рекурентної бази даних в базисі Галуа / В.В.Шаряк // Наукові вісті. ПВНЗ „ГА”. - Вип.№17(1). - Івано-Франківськ, 2010. – С.99–102.
63. Шаряк В.В. Реалізація багаторівневої рекурентної бази даних у базисі Галуа / В.В.Шаряк / Вісник ТДТУ ім.І.Пулюя. - №3, 2009. – Тернопіль. – С.34-44.
64. Шаряк В.В. Архітектура і кодування баз даних на основі теоретико-числових базисів / В.В.Шаряк / Вісник ТДТУ. - №1 (3), 2007. – Тернопіль. - С.17-23.
65. Шаряк В.В. Реалізація багаторівневої вертикально-рекурентної моделі бази даних в базисі Галуа / В.В. Шаряк // Поступ в науку. - Вип. №7, 2011. – С.360-365.
66. Я.М.Николайчук, В.В.Шаряк. Особливості архітектури та характеристик лінійно-рекурентної структури бази даних // Вісник Хмельницького національного університету. - №3,Том.1, 2007. - С.117-119.
67. Яцків В.В. Оцінка ефективності двомірних методів маніпуляції сигналів з самосинхронізацією. - Івано-Франківськ.: ІФДТУНГ, 1998.

ДЛЯ ЗАМІТОК

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Blank lined paper with horizontal ruling lines.

Наукове видання

В.В.Шаряк, Н.Я.Возна, Я.М.Николайчук

**КОДУВАННЯ ДАНИХ ТА ОРГАНІЗАЦІЯ
РОЗПОДІЛЕНИХ БАЗ ДАНИХ
У РОЗШИРЕНИХ ПОЛЯХ ГАЛУА**

Монографія

Підписано до друку 01.03.2024 р.
Формат 60х84/16. Папір офсетний.
Друк офсетний. Зам. № 24-359
Умов.-друк. арк. 15,9. Обл.-вид. арк. 13,1.
Тираж 100 прим.

Віддруковано ФО-П Шпак В. Б.
Свідоцтво про державну реєстрацію В02 № 924434 від 11.12.2006 р.
м. Тернопіль, бульвар Просвіти, 6/4. тел. 097 299 38 99.
E-mail: tooums@ukr.net

*Свідоцтво про внесення суб'єкта видавничої справи до державного
реєстру видавців, виготовлювачів і розповсюджувачів видавничої продукції
ДК № 7599 від 10.02.2022 р.*