

ВИКОРИСТАННЯ МОНОЛІТНИХ КОДІВ В ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЯХ

Вступ

Під монолітним розуміємо код, дозволені комбінації якого складаються з пакетів однойменних символів, що знаходяться один поруч одного. Монолітний двійковий код має ряд переваг перед іншими кодами. Одна з них - простота виявлення та виправлення помилок на приймальній стороні, бо поява хоча б одного символу "1" серед нулів, або символу "0" серед одиниць у прийнятій кодовій комбінації вказує на помилку. Помилка не виявляється лише у тих випадках, коли хибний сигнал виникає в першому або останньому символах пакету (на межі між пакетами нулів та одиниць). Якщо в монолітному коді з'являються хибні символи, то всі вони або частина з них зразу ж виявляються, що спрощує виявлення помилок і забезпечує високу завадостійкість монолітного коду [1].

Використання монолітного коду на основі числових лінійок-в'язанок для кодування інформації

Досліджуючи системи кодування інформації з різними законами розподілу ваг розрядів монолітного коду, легко побачити, що в деяких випадках розподілу ваг монолітний код виявляється надміру надлишковим, бо одні й ті ж числа подаються кількома різними кодовими комбінаціями двійкового позиційного коду [1, 2].

Розв'язок поставленої задачі зводиться до пошуку оптимального комбінаторного варіанту ваг розрядів монолітного коду, при якому будь-яке натуральне число можна було б подати в монолітному коді єдино можливим способом. Виникає проблема вибору оптимальної системи ваг розрядів, суть якої полягає в тому, щоб множині кодових комбінацій монолітного коду взаємно однозначно відповідала множина чисел натурального ряду.

Інтерес для дослідження становлять системи кодування, які базуються на застосуванні комбінаторних властивостей ідеальних кільцевих в'язанок [2]. Ідеальна кільцева в'язанка (ІКВ) - це алгебраїчна структура, утворена на послідовності n цілих додатних

чисел, значення яких, як і значення сум поруч розміщених між собою чисел, вичерпують r разів натуральний ряд. Ідеальним кільцем монолітного коду (КМК) називається послідовність $K_n = (k_1, k_2, \dots, k_i, \dots, k_n)$ чисел, на якій всі можливі кільцеві суми вичерпують r разів значення чисел натурального ряду $1, 2, \dots, S_n^r = n(n-1)/r+1$. Наведемо табл.1 кодових комбінацій, утворених на ІКВ (1, 3, 2, 7, 8, 10) з параметрами $n=6, r=1$.

Таблиця 1

Кодові комбінації кільцевого монолітного коду (1, 3, 2, 7, 8, 10)

Число	Код	Число	Код	Число	Код	Число	Код
0	000000	8	000010	16	111001	24	111011
1	100000	9	001100	17	001110	25	000111
2	001000	10	000001	18	000011	26	100111
3	010000	11	100001	19	100011	27	001111
4	110000	12	011100	20	011110	28	101111
5	011000	13	111100	21	111110	29	110111
6	111000	14	110001	22	110011	30	011111
7	000100	15	000110	23	111101	31	111111

Потужність кільцевого монолітного коду, реалізованого на ІКВ n -го порядку, визначається загальним числом способів утворення кодових слів.

Алгоритм синтезу кільцевого монолітного коду

Крок 1. За значенням n обчислюється потужність кільцевого монолітного коду S_n^r ;

Крок 2. Визначається масив із n комірок, пронумерованих у зростаючому порядку;

Крок 3. В першу комірку масиву записується число, з якого починається відлік в КМК, в другу - наступне, в решту заносяться нулі;

Крок 4. Перший раз число A визначається як збільшене на одиницю найбільше число найкоротшого ряду послідовності чисел, утвореного на множині сум, які знайдені на всіх окремих послідовностях, що належать масиву. Наступний раз число A з тим самим значенням визначається як збільшене на одиницю наступне за найбільшим числом найкоротшого ряду послідовності чисел, поки

сума усіх чисел масиву L'_n менша за S'_n . При існуючих вільних комірках масиву число A записується у вільну комірку з найменшим порядковим номером і обчислюється нове значення суми L'_n усіх чисел масиву; у випадку коли $L'_n \leq S'_n$ виконується крок 5, а при $L'_n > S'_n$ - виконується крок 7;

Крок 5. Обчислюється нове значення суми елементів масиву КМК L'_n . При існуючих у масиві вільних комірках знаходяться всі суми на всіх послідовностях, а при їх відсутності - всі лінійні суми на єдиній послідовності:

а) якщо кожна зі знайдених сум зустрічається не більше одного разу і є вільні комірki, то здійснюється перехід до кроку 4. При відсутності вільних комірок і при виконанні умови, що нове значення суми L'_n дорівнює S'_n , отримується варіант КМК, після чого виконується крок 7. В іншому випадку виконується крок 6;

б) якщо хоча б одна зі знайдених сум з'являється більше одного разу, то виконується крок 6;

Крок 6. Знаходиться найбільше число B , потім визначається чи є вільний номер комірки з номером більшим ніж той, де розташоване число B ; якщо така комірка існує, то з комірки з меншим номером число B переноситься у вільну комірку з більшим номером, після чого виконується крок 5; в протилежному випадку виконується крок 7.

Крок 7. Звільняється комірка з числом B і виконується крок 6. Ознакою закінчення обчислень при побудові повної сім'ї КМК служить поява числа, з якого починається відлік в КМК в комірці $(n+3)/2$ при умові його відсутності в попередніх комірках для непарних значень n і аналогічно в комірці $(n+2)/2$ для парних значень n .

Алгоритм кодування за допомогою числових лінійок-в'язанок

Для побудови циклічного коду за допомогою ІКВ виділимо рядок із S_n пронумерованих у зростаючому порядку клітинок одновимірного масиву і заповнимо інформаційними "одиницями" клітинки, номери яких збігаються з числами, визначеними з ІКВ. У клітинки, що залишилися незаповненими, занесемо "нулі". Утворена послідовність

одиниць і нулів є S_n -розрядною кодовою комбінацією, циклічним зсувом якої можна одержати й решта дозволених комбінацій.

Кожна з $S_n(S_n - 1)/2$ різних пар кодових комбінацій містить точно r із n одиничних символів в однойменних розрядах, що впливає з властивостей ІКВ. Решта $n - r$ символів однієї і стільки ж іншої кодової комбінації відрізняються від символів, що містяться в однойменних розрядах. Тому мінімальна кодова відстань для даного коду визначається як:

$$d_{\min} = 2(n - r) \quad (1)$$

Число помилок, які можна виявити t_1 , і число помилок, що можна виправити t_2 за допомогою коректуючого коду, визначається мінімальною кодовою відстанню залежностями:

$$t_1 \leq d_{\min} - 1, \quad t_2 \leq (t_1 - 1)/2 \quad (2)$$

Визначення кількості помилок, які можуть бути виправлені t_2 або виявлені t_1 :

$$t_1 \leq 2(n - r) - 1, \quad t_2 \leq n - r - 1 \quad (3)$$

Прикладом коду є табл.2 кодових комбінацій, складена на основі ІКВ (1, 2, 6, 4):

Таблиця 2

Кодові комбінації, складені за допомогою ІКВ (1, 2, 6, 4)

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

Потужність побудованого за допомогою ІКВ коду можна збільшити вдвічі, якщо табл.2 доповнити комбінаціями, які утворюються зі знайденої таблиці шляхом заміни інформаційних одиниць нулями, і навпаки. Кодова відстань знаходиться як:

$$d = S_n - 2(n - r) \quad (4)$$

Формули визначення числа помилок, які підлягають виявленню чи виправленню:

$$\left. \begin{array}{l} t_1 \leq 2(n-r)-1 \\ t_2 \leq n-r-1 \end{array} \right\}, \text{ якщо } S_n \geq 4(n-r);$$

$$\left. \begin{array}{l} t_1 \leq S_n - 2(n-r) - 1 \\ t_2 \leq \frac{S_n - 2(n-r+1)}{2} \end{array} \right\}, \text{ якщо } S_n < 4(n-r); \quad (5)$$

У розглянутих випадках значення параметрів n і r не зв'язані між собою будь-якою залежністю і можуть вибиратися довільно. При цьому виникає питання про встановлення оптимального співвідношення між n і r , за дотримання якого розглянутий код набуває додаткових переваг. Завадостійкість коду зростає зі збільшенням різниці $\delta = n - r$. Максимальне значення δ досягається за умови:

$$S_n = 2n \quad (6)$$

Співвідношення між параметрами n і r , коли код набуває здатності виявляти та виправляти максимально можливу кількість помилок:

$$P = \begin{cases} n/2, & n - \text{парне} \\ (n-1)/2, & n - \text{непарне} \end{cases} \quad (7)$$

Алгоритм кодування за допомогою числових лінійок-в'язанок базується на основі поняття монолітного коду та представленні кодуєчого числа ASCII – формату та запропонованого методу кодування з використанням числових лінійок-в'язанок та архівації отриманих даних [1].

Алгоритм кодування можна умовно розбити на блоки:

Блок “Ініціалізація бази даних ІКВ” – завантаження ідеальних кільцевих в'язанок в масив рядків;

Блок “Ввід поточної інформації для кодування. Обчислення суми елементів ІКВ” – Введення числа для кодування та вибір відповідної в’язанки. Здійснюється перевірка на пусті поля вводу (обробка виняткових ситуацій) та ознаку вибору в’язанки;

Блок “Порівняння” – здійснюється перевірка суми елементів ІКВ з введеним числом. Якщо введене число більше суми елементів, то повторний ввід числа;

Блок “Обчислення позицій суми проіндексованої в’язанки” – індексуються позиції у в’язанці та знаходяться відповідні індекси суми елементів, що відповідає введеному числу;

Блок “Побудова таблиці монолітного коду згідно позицій” – генерується таблиця можливих комбінацій в’язанки та монолітний код (від першої позиції індексів до другої заповнюються “1”, все решта “0”);

Блок “Генерація кодових комбінацій відповідно до індексів числової в’язанки” - генерується таблиця кодових комбінацій, де кожна позиція-індекс відповідає коду числа;

Блок “Побудова кодової комбінації згідно індексів в таблиці кодів” – остаточний результат згенерованого коду шляхом з’єднання двох кодів-позицій;

Блок “Цикл архівації” – цикл обробки згенерованої комбінації. Заміна блоків “1” та “0” відповідними символами ASCII (груповані блоки “0” та “1” замінюються обраним відповідно для “0” та “1” ASCII-кодом та кількістю нулів та одиниць). В результаті зменшується об’єм закодованої інформації та ускладнюється процес підбору ключа для стороннього декодування.

Побудовані за допомогою ІКВ завадостійкі коди дають змогу виявляти до $n-1$ або виправляти до $n/2-1$ помилок для парних, і виявляти до n або виправляти до $(n-1)/2$ помилок для непарних значень n .

Аналіз результатів дослідження властивостей ІКВ підтверджує актуальність і перспективність впровадження монолітних кодів та створення на їхній основі нових інформаційних технологій з використанням спеціальних систем кодування інформації з високою захищеністю від завад й стороннього декодування, що дає змогу розширити сферу застосувань комбінаторних методів оптимізації в інформатиці.

Висновки

1. Показана можливість спрощеної побудови за допомогою ІКВ моделей монолітного коду розширеного класу заводостійких кодів, створення ефективних алгоритмів кодування і декодування інформації.

2. Дослідження моделей і методів комбінаторної оптимізації розширює сферу практичних застосувань монолітних кодів в задачах інформаційної техніки і проектування систем кодування.

ЛИТЕРАТУРА

1. Різник О.Я. Заводостійкий спосіб перетворення сигналів // Матеріали Четвертої укр. конф. з автоматичного керування ("Автоматика-97"). - Черкаси. - 1997. - С.34.
2. Різник В.В. Синтез оптимальних комбінаторних систем. - Львів, 1989.
3. Кісь Я.П. Синтез заводостійких кодів за комбінаторними конфігураціями// Комп'ютерна інженерія та інформаційні технології. Вісник ДУ "Львівська політехніка". № 294. – 1995. – С. 60-63