УДК 681.3:519.15

О.Т. Велика, М.С. Міщенко, В.В. Різник[*]
Національний університет "Львівська політехніка",
кафедра автоматизованих систем управління,
[*]Аграрно-технічна академія, м. Бидгощ (Польща)

СИНТЕЗ ІДЕАЛЬНИХ КІЛЬЦЕВИХ В′ЯЗАНОК МЕТОДОМ СУПРОВІДНИХ МАТРИЦЬ

На основі методу супровідних матриць поліномів та відповідних алгебричних перетворень з урахуванням властивостей циклічних груп в розширених полях Галуа розроблено вдосконалений алгоритм синтезу ідеальних кільцевих в′язанок. Ці алгебричні конструкції можуть бути застосовані для проектування кодів зі самоконтролем.

Innovative synthesis of the Ideal Ring Bundles based on the polynomials accompany matrix method, including a property of cyclic groups in extensions of Galois fields and the appropriate algebraic techniques have been designed. These algebraic constructions provide opportunity to apply them to self-checking code design.

Постановка проблеми у загальномиу вигляді

Комбінаторні моделі та системи широко застосовують в задачах комп′ютерної інженерії та проектування засобів кодування та перетворення інформації, а комбінаторні методи оптимізації застосовуються в системах автоматизованого проектування та керування. Тому актуальними є дослідження існуючих та пошук нових комбінаторних моделей, способів їхньої побудови, класифікація, визначення умов існування, виявлення взаємних зв'язків та інтерпретацій. З'явились нові терміни: ідеальна кільцева в'язанка, магічне коло (magic circle), продовжується пошук зручних для дослідження та користування в інженерній практиці моделей і методів проектування засобів кодування та перетворення інформації, розробляються загальні основи теорії оптимальних комбінаторних систем, структура яких визначає оптимальне відношення інцидентності між елементами цієї системи. Таке трактування структури пов′язується зазвичай з такими критеріями, як мінімізація надмірності або досягнення максимальної комбінаторної різноманітності системи при встановлених обмеженнях на правила взаємодії елементів і функціонування системи загалом.

В основі розв'язання проблеми синтезу оптимальних комбінаторних систем лежать методи структурної оптимізації, які базуються на положеннях комбінаторного аналізу. Однак ці методи не завжди достатньо прості для практичного застосування під час розв′язування інженерних задач, оскільки вимагають ознайомлення з алгебричною теорією чисел, скінченних груп і полів Галуа.

Кодування інформації за ідеальними кільцевими в'язанками

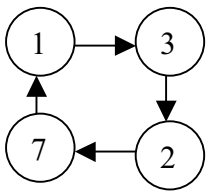

Рис. 1. Графічна схема одновимірної ІКВ четвертого (n=4) порядку

Ідеальна кільцева в'язанка (ІКВ) – це впорядкована послідовність цілих додатних чисел, розміщених у вигляді циклічної послідовності, всі числа якої, включно з множиною утворених сум із двох, трьох і т.д. поруч розміщених чисел, вичерпують натуральний ряд – від 1 до суми всіх чисел цієї послідовності [1].

Для початку розглянемо одновимірну (t=1) ІКВ четвертого (n=4) порядку, елементами якої є числова послідовність (1,3,2,7). Графічна схема цієї ІКВ наведена на рис. 1.

З рис. 1 випливає, що будь-якому натуральному числу від 1 до 13 можна поставити в однозначну відповідність певну кодову послідовність довжиною n=4 з "нулів" і "одиниць", ваги розрядів яких є числами ІКВ, причому усі "одиниці" будь-якої дозволеної кодової комбінації розміщені поруч одна однієї (а отже, і нулі також).

Нижче наведено табл. 1 для кодування чисел за допомогою ІКВ (1,3,2,7).

Таблиця 1

Кодування чисел за допомогою ІКВ (1,3,2,7)

Ваги розрядів	КОДОВІ КОМБІНАЦІЇ ЧИСЕЛ												
	1	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***	***10***	***11***	***12***	***13***
1	1	0	0	1	0	1	0	1	0	1	1	0	1
3	0	0	1	1	1	1	0	0	0	0	1	1	1
2	0	1	0	0	1	1	0	0	1	1	0	1	1
7	0	0	0	0	0	0	1	1	1	1	1	1	1

Легко побачити, що всі чотирирозрядні кодові комбінації чисел від 1 до 13, занесені в табл.1, утворюють так званий “монолітний” код, де кожна комбінація складається не більш ніж з двох блоків однойменних символів (“одиниць” і “нулів”). Така властивість монолітного коду дає змогу подавати будь-яке натуральне число від 0 до 13 у вигляді відповідної частини ІКВ. У загальному випадку можуть утворюватися більш складні комбінаторні конфігурації-в′язанки з десятками і сотнями елементів, кожен з яких – вже не число, а t-кортеж чисел, де t –кількість цілих додатних чисел у кортежі. За таких умов з’являється можливість кодування векторів за допомогою t-вимірних ІКВ [1].

Сучасні методи кодування та пересилання інформації характеризуються підвищеними вимогами щодо надійності її збереження у реальному часі інтенсивних потоків даних в умовах діяння зовнішніх впливів та завад. Для забезпечення ефективної системи контролю під час управління будь-яким процесом важливого значення набуває не лише захист даних від завад, але й забезпечення високої швидкості виявлення та виправлення помилок. Одним із підходів до вирішення цієї проблеми є застосування кодів зі самоконтролем, до яких належить кільцевий монолітний код [1]. На відміну від більшості відомих багаторозрядних надлишкових кодів, розшифровування яких автоматично призводить до виявлення помилок [2], монолітний код не потребує здійснення операцій, які пов′язані з визначенням ідентичності або відмінності кодових комбінацій між собою. Це дає змогу значно спростити алгоритм самоконтролю і тим самим забезпечити високу ефективність системи контролю у реальному масштабі часу. Однак для побудови монолітного коду з широкими функціональними можливостями потрібно спочатку знайти його числову модель, а для цього доводиться звертатися до алгебричної теорії скінченних груп та розширених полів Галуа [3]. Класична теорія скінченних полів хоча і має фундаментальне значення для теорії кодування, однак не завжди є достатньо доступною для інженерів та спеціалістів у галузі інформаційних технологій. Тому актуальною проблемою є розроблення ефективних алгоритмів синтезу ідеальних кільцевих в′язанок високих порядків як зручних моделей для проектування кодів зі самоконтролем.

Аналіз останніх досліджень

Дослідження методів ефективного синтезу завадостійких кодів тісно пов′язано з теорією скінченних полів, оскільки вартість опрацювання кодів значною мірою залежить від способу подання полів, для яких, як відомо, властивий ізоморфізм, що дозволяє розглядати усі можливі поля фіксованого порядку як різні подання одного й того ж поля [3]. Для урахування цієї обставини використовується супровідна матриця полінома в натуральному базисі [4]. Один з алгоритмів синтезу завадостійкого коду на основі вищезгаданого підходу полягає в побудові поля GF (p^n), знаходженні первісних незвідних над цим полем полінома 3-го степеня та утворення досконалої різницевої множини з виділенням усіх індексів відповідних елементів поля GF [$(p^n)^3$] [3]. Для побудови первісних незвідних поліномів не існує регулярних методів, а відомі підходи стосуються знаходження різних полів за допомогою різних методів.

Цілі статті

Предметом дослідження є алгоритм синтезу ідеальних кільцевих в′язанок високих порядків з метою подальшого їх застосування для проектування кодів зі самоконтролем. З цією метою передбачається використання методу супровідних матриць поліномів, відповідних алгебричних перетворень та деяких властивостей циклічних груп в розширених полях Галуа.

Основний матеріал

Обраний підхід полягає у використанні первісних незвідних поліномів розширених полів Галуа та відповідних матриць для синтезу ідеальних кільцевих в'язанок (ІКВ).

Вхідними даними є такі параметри:

p — будь-яке просте число;

α — будь-яке ціле число;

a_0, a_1, a_2 — коефіцієнти полінома.

Параметри можуть змінюватися в межах обраного діапазону.

Вихідними даними є синтезовані ідеальні кільцеві в'язанки, які задовольняють вхідні параметри.

Алгоритм синтезу ідеальних кільцевих в′язанок високих порядків полягає у виконанні таких операцій.

1. Знайти первісний незвідний над полем $GF(p^{\alpha})$ поліном $f(x)$ третього степеня, де p — просте число, що визначається з рівняння $n = p^{\alpha} + 1$, де n — порядок ІКВ, а a — ціле число.

2. Записати супровідну матрицю A первісного незвідного над полем $GF(p)$ полінома $f(x)=x^3+a_2x^2+a_1x+a_0$ у натуральному базисі:

$$A = \begin{Vmatrix} 0 & 0 & -a_0 \\ 1 & 0 & -a_1 \\ 0 & 1 & -a_2 \end{Vmatrix}. \qquad (1)$$

3. Помноживши матрицю A на вектор-стовпець $b_1 = \begin{vmatrix} 1 \\ 0 \\ 0 \end{vmatrix}$, записати наступний вектор-стовпець b_2,. Продовжити далі запис до заповнення $S_n=n(n-1)+1$ послідовними стовпцями, де кожний наступний стовпець є результатом множення матриці A на попередній стовпець.

4. Записати порядкові номери $i_1, i_2, \ldots, i_n$ вектор-стовпців, що мають в останньому рядку нуль, і знайти елементи кільцевої в'язанки за формулою:

$$5.\ k_i = \begin{cases} b_{i+1} - b_i, & i = \overline{1, n-1} \\ b_1 - b_n \pmod{S_n}, & i = n \end{cases} \qquad (2)$$

6. Перевірити, чи знайдена кільцева в'язанка є ідеальною. Для цього будують спеціальну таблицю (суміжних сум), до якої числове значення кожної суми повинно входити рівно R разів.

7. Якщо вимога п.5 задовольняється, знайдена кільцева в'язанка є ідеальною, а у протилежному випадку необхідно здійснити чергову зміну в наборі коефіцієнтів полінома $f(x)=x^3+a_2x^2+a_1x+a_0$.

Блок-схему алгоритму синтезу ідеальних кільцевих в′язанок високих порядків наведено на рис. 1.

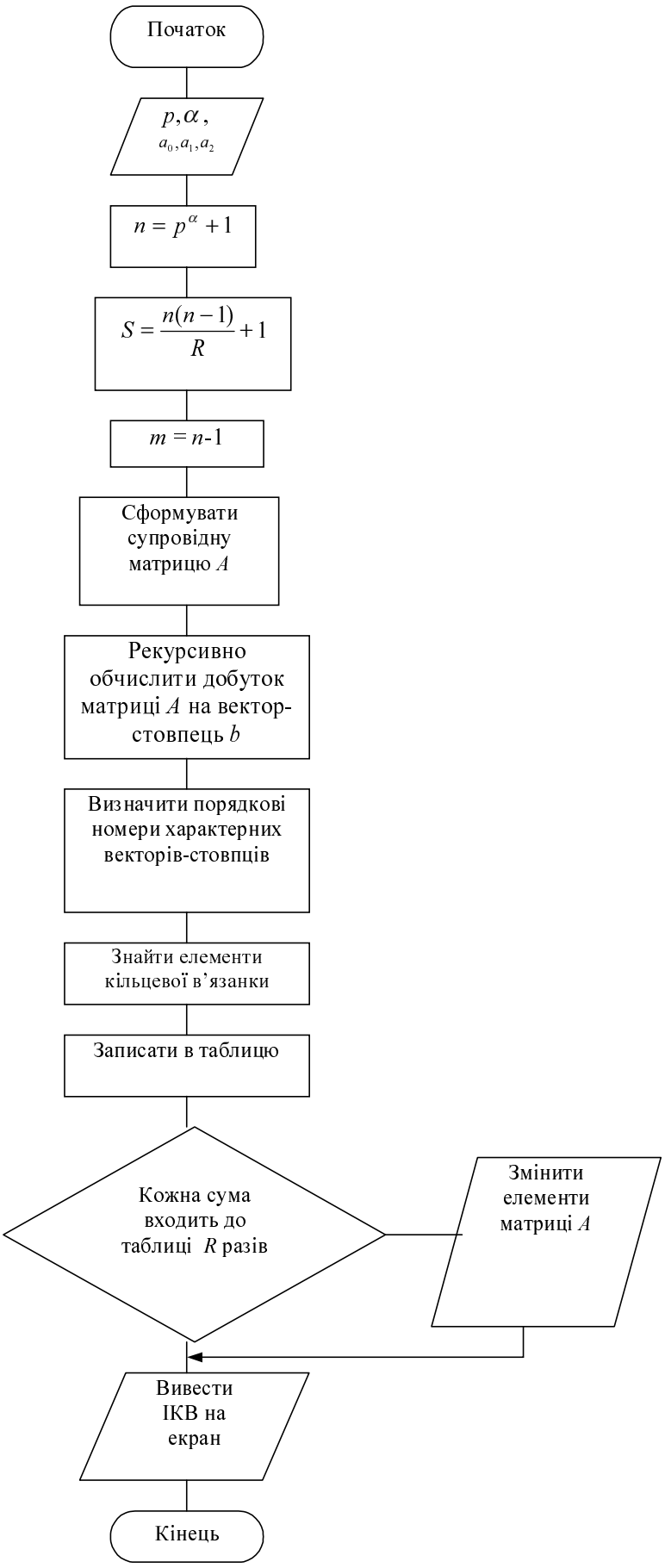

Рис. 1. Блок-схема алгоритму синтезу ідеальних кільцевих в'язанок високих порядків

Огляд мови логічного програмування Prolog

Турбо-Пролог – це здійснена компанією Borland International реалізація мови програмування високого рівня Пролог компіляторного типу. Її вирізняє висока швидкість компіляції і обчислень. Турбо-Пролог призначений для видачі відповідей, які він логічно виводить за допомогою своїх могутніх внутрішніх процедур. Відомо, що програма мовою Турбо-Пролог в кілька рядків може замінити кілька сторінок тексту при програмуванні будь-якою іншою мовою [5].

Завдяки наявності потужних засобів зіставлення Турбо-Пролог придатний не тільки для використання в програмах, що належать до сфери штучного інтелекту й обробки природно-мовних конструкцій, але також застовується в таких традиційних галузях, як управління базами даних. Будь-яка програма, написана мовою Турбо-Пролог, складається з п'яти розділів. Такими є: розділ опису доменів, розділ бази даних, розділ опису предикатів, розділ опису мети і розділ опису тверджень. Ключові слова domains, database, predicates, goal і clauses позначають початок відповідних розділів.

Призначення цих розділів таке:

– розділ domains містить визначення доменів, що описують різні класи об'єктів, використовуваних у програмі;

– розділ database містить твердження бази даних, що є предикатами динамічної бази даних. Якщо програма такої бази даних не вимагає, то цей розділ може бути опущений;

– розділ predicates служить для опису використовуваних програмою предикатів;

– у розділі goal мовою Турбо-Пролог формулюється призначення створюваної програми. Складовими частинами при цьому можуть бути якісь підцілі, з яких формується єдина мета програми;

– у розділ clauses заносяться факти і правила, відомі апріорі. Про вміст цього розділу можна говорити як про дані, необхідні для роботи програми.

Більшість програм, однак, не містить усіх п'яти названих розділів.

Турбо-Пролог забезпечує можливість введення до програми коментарів, що виділяються символами /* і */. Коментарі можна поміщати в будь-якому місці програми, причому щодо їх довжини немає практично жодних обмежень. Для того, щоб використовуватися за призначенням, коментарі повинні містити інформацію про саму програму, ім'я програмного файла, компілятор, базу даних, а також про призначення кожного з предикатів і правил, що не є достатньо очевидними.

Опис контрольного прикладу

Нехай потрібно синтезувати просту в’язанку 48-го порядку. Для цього на першій сторінці вікна програми вказати значення параметра p=47, позначити і натиснути кнопку Запуск. Через деякий період часу, залежно від потужності комп’ютера, програма видасть результат, який відповідає обраним вхідним даним (рис. 2).

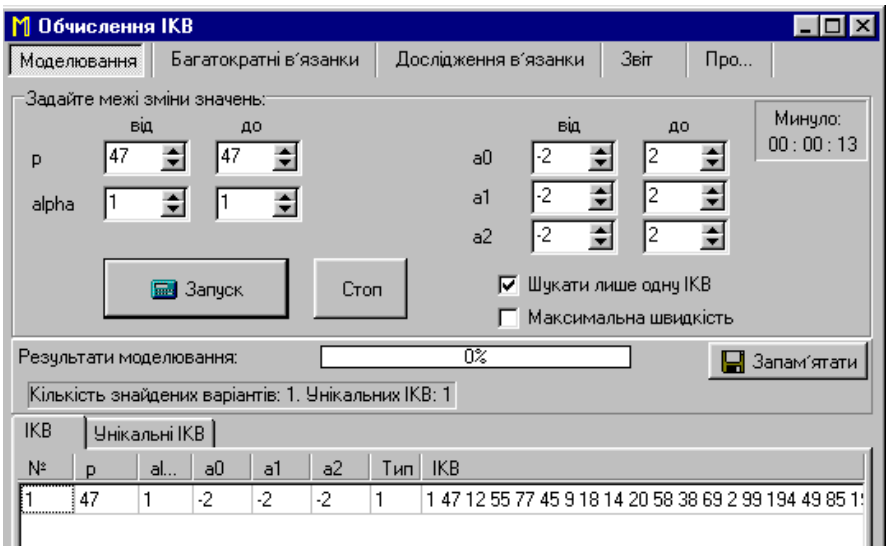

Рис. 2. Заставка обчислення ІКВ з вікнами обрання меж зміни значень вхідних даних і висвітленням таблиці результатів синтезу ІКВ

У програмі реалізовано захист від помилкового введення початкових даних.

Результати роботи програми можна запам’ятати у текстовому форматі.

Для заданого полінома знаходяться всі ІКВ, які він породжує при певних межах зміни значення параметра *p*. Наприклад, якщо обрано поліном $x^3 - x^2 - 2x - 2$, а параметр *p* змінюється в діапазоні від 1 до 239, заставка обчислення ІКВ набуває такого вигляду (рис. 3):

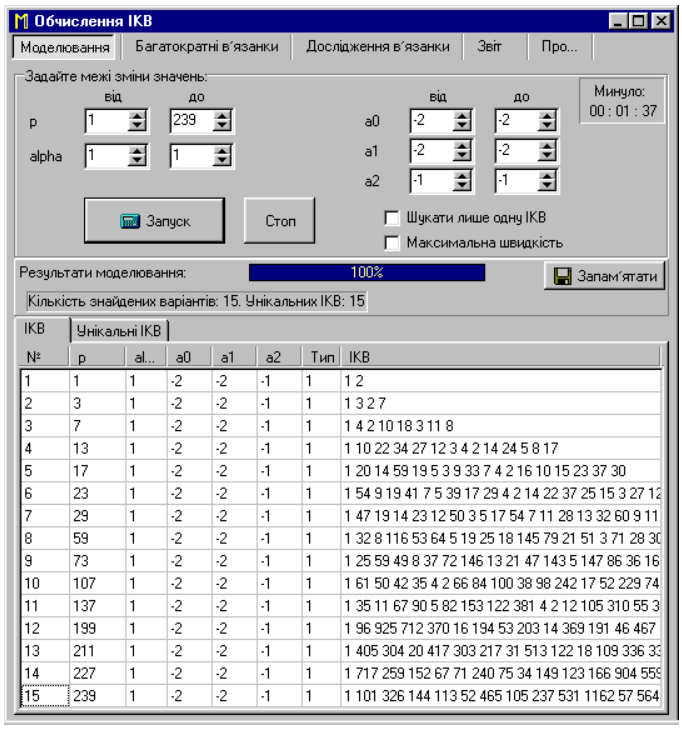

Рис. 3. Результати обчислення ІКВ за поліномом $x^3 - x^2 - 2x - 2$, *якщо параметр* p *змінюється в діапазоні від 1 до 239*

За результатами обчислення знайдено сім′ю ІКВ із 15 варіантів, які породжуються поліномом $x^3 - x^2 - 2x - 2$.

У Пролог-програмі поставлено захист від введення нечислового значення при запиті на введення параметра *n*.

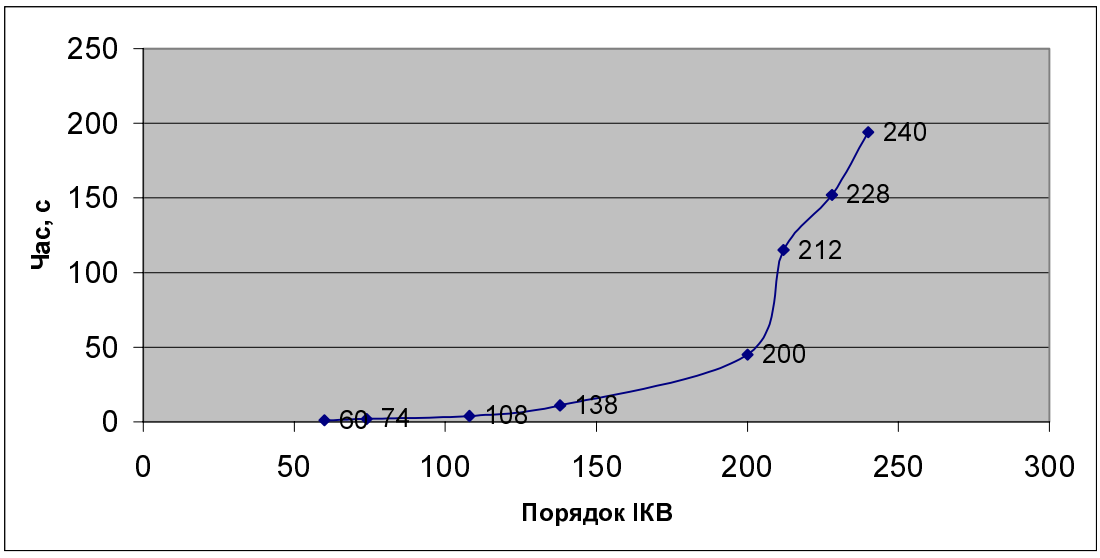

Рис. 4. Залежність тривалості обчислення ІКВ від їх порядку

У табл. 2 наведено залежність тривалості обчислення ІКВ від його порядку, а графік (рис. 4) ілюструє цю залежність. Тестування проводилося на комп'ютері з процесором Celeron 500 МГц фірми Intel.

Таблиця 2

Залежність тривалості обчислення ІКВ від його порядку

Порядок ІКВ	60	74	108	138	200	212	228	240
Час, с	1	2	4	11	45	115	152	194

Залежність тривалості обчислення ІКВ від їх порядку (рис. 4) дає підстави стверджувати, що розроблений на основі методу супровідних матриць алгоритм синтезу ідеальних кільцевих в′язанок дозволяє генерувати за лічені секунди сім′ї ІКВ нижче 50-го і декілька десятків секунд – нижче 200-го порядків. Побудова ІКВ вище 200-го порядку вимагає більшого часу обчислень, який надалі починає швидко зростати.

Висновки

Розроблений на основі методу супровідних матриць удосконалений алгоритм синтезу ідеальних кільцевих в′язанок дає змогу генерувати за лічені секунди сім′ї ІКВ високих порядків – до кількох сотень. За складеною програмою побудовано сім′ї ІКВ до 401-го порядку. Спроба синтезувати прості (однократні) і багатократні ІКВ високих порядків прямим перебиранням усіх варіантів для вказаного порядку кільцевої в'язанки в межах зміни їх елементів виявилася незадовільною через швидке зростання тривалості обчислення.

Результати комп′ютерного синтезу узгоджуються з теоретичними висновками щодо існування нескінченно великих сімей досконалих структур, якими є ідеальні кільцеві в′язанки. Ці алгебричні конструкції можуть застосовуватися для проектування потужних систем кодування інформації зі самоконтролем, до яких належить монолітний код. Останній має важливе значення не лише з погляду надійного захисту даних від завад, але й для забезпечення високої швидкості виявлення та виправлення помилок.

1. Різник В.В. Комбінаторні моделі і методи оптимізації в задачах інформатики: Навч.посібник. – К.: НМК ВО, 1991. – 72 с. 2. Цымбал В.П. Теория информации и кодирования. –К., 1982. 3. Холл М. Комбинаторика. – М. 1970.4. Свердлик М.Б. Оптимальные дискретные сигналы. – М., 1975.5. Братко И. Программирование на языке Пролог для искусственного интеллекта: Пер. с англ. — М.: Мир, 1990. – 560с., ил.