

ПІДВИЩЕННЯ ШВИДКОДІЇ ДЕКОДЕРУ НЕРЕГУЛЯРНИХ LDPC-КОДІВ НА ОСНОВІ ОРГАНІЗАЦІЇ ПАРАЛЕЛЬНИХ ЧЕРГ ЗАПИСУ/ЗЧИТУВАННЯ

У статті проводиться дослідження частково паралельного LDPC-декодера нерегулярних кодів, який працює за алгоритмом мінімальної суми. Представлена модель організації запису/зчитування на основі паралельних черг, що дозволяє виконувати операції доступу до пам'яті з максимальним ступенем паралелізму відповідно до структури матриці

Ключові слова: LDPC, декодування, паралельні черги.

MAKSYM MUSIYENKO, YAROSLAV KRAINYK
Petro Mohyla Black Sea State University, Ukraine

ПІДВИЩЕННЯ ШВИДКОДІЇ ДЕКОДЕРУ НЕРЕГУЛЯРНИХ LDPC-КОДІВ НА ОСНОВІ ОРГАНІЗАЦІЇ ПАРАЛЕЛЬНИХ ЧЕРГ ЗАПИСУ/ЗЧИТУВАННЯ

This article provides research of irregular codes LDPC-decoder working with minimal sum. Model parallel queues of read/write organization that allows performing parallel access to memory with maximum parallelism according to matrix structure.

Key words: LDPC, decoding, parallel queues.

Вступ

Дослідження LDPC-кодів, а особливо нерегулярних LDPC-кодів (англ. Low Density Parity Check Codes – коди з низькою щільністю перевірки парності), є важливим науково-технічним питанням, оскільки поліпшення їх характеристик забезпечує кращі показники виправлення помилок при меншій потужності сигналу. Недоліком даного класу кодів є вища складність апаратної реалізації декодування, що пов'язана з характером розташування елементів матриці перевірки. Можливі колізії доступу до необхідних значень призводять до того, що операції зчитування/запису можна виконувати лише послідовно або з надмірним використанням ресурсів пам'яті, що зменшує пропускну здатність декодера. Паралельна реалізація даних операцій здатна збільшити швидкодію, проте вимагає додаткових витрат логічних ресурсів.

Важливим аспектом роботи LDPC-декодера є зміна конфігурації роботи, яка може проводитися з різною метою: завантаження матриці, що забезпечує кращі показники виправлення помилок, зменшення кількості операцій у процесі декодування і т.д. Робота декодера з різними матрицями можлива лише при накладанні певних обмежень на характеристики матриці. При цьому, робота з різними матрицями з послідовним виконанням запису/зчитування може бути єдиним варіантом, в той час як для інших послідовних реалізацій, за наявності можливостей для розпаралелювання, означатиме програш у швидкодії.

Саме тому організація роботи декодера для нерегулярних кодів з можливістю реконфігурації з забезпеченням максимальної швидкодії є важливою та актуальною для дослідження.

Метою даної роботи було збільшення пропускну здатності частково паралельного декодера LDPC-декодера з алгоритмом декодування мінімальної суми, що працює з нерегулярними LDPC-кодами, за рахунок розробки моделі паралельних черг запису/зчитування для використання можливостей паралельного виконання операцій в залежності від структури матриці перевірки парності. Також необхідно було максимально розширити можливості моделі для виконання реконфігурації – роботи з різними матрицями.

Огляд попередніх досліджень

Автори роботи [1] використовують у якості пристрою, що забезпечує паралельну обробку, пристрій швидкого зсуву (barrel shifter). Подібна реалізація паралельної обробки дозволяє досягти дуже високої швидкості. Проте, розглянуто лише варіанти матриць, структура розташування елементів яких спрощує реалізацію пристрою зсуву, в той час, як ускладнення структури матриці, значно підвищило б складність його реалізації.

У роботі [2] вирішення проблеми паралельної обробки пропонується на основі мережі Бенеша. У роботі розглянуто рішення для квазіциклічних LDPC-кодів. Проте, така мережа комутації вимагає декілька циклів обробки для передачі сигналу на потрібний вихід, що може зменшувати швидкодію.

Основна частина

Основними характеристиками матриці перевірки парності для нерегулярних LDPC-кодів є: M – кількість рядків матриці, N – кількість стовпців матриці, $w_{r\max}$ – максимальна кількість значущих елементів у рядку.

Причому, хоча б для одного рядка з індексом i виконується умова

$$w_{ri} \neq w_{r\max} \quad (1)$$

Для паралельного обчислення значень повідомлень перевірки стовпці матриці розділяють на інтервали, які характеризуються мінімальним та максимальним індексами, що входять до інтервалу:

$$I_k = [j_{k\min}; j_{k\max}]; 0 \leq j_{k\min} < N; j_{k\min} < j_{k\max} \quad (2)$$

Кількість інтервалів k обирається відповідно до характеристик матриці та на основі подальших досліджень відносно розташування значущих елементів. При цьому значення мінімального індексу

наступного інтервалу має бути на одиницю більше за максимальний індекс попереднього:

$$j_{k+1\min} = j_{k\max} + 1. \quad (3)$$

Кожному значущому елементу в рядку можна співставити його порядковий номер:

$$n_{iv} = v; v = 1 \dots w_{ri}. \quad (4)$$

Добре піддаються розпаралелюванню LDPC-коди, для матриці яких виконується умова, що одному індексу значущого елементу j завжди відповідає одне значення порядкового номеру n_{ij} для всіх рядків:

$$n_{ij} = const; i = 1 \dots M. \quad (5)$$

У такому випадку розбиття на інтервали I_k значно спрощується.

В той же час, коли індексу значущого елементу j може відповідати декілька різних значень порядкових номерів, організація паралельних обчислень стає проблематичною, оскільки можуть виникати колізії доступу до елементів, які можна вирішити лише послідовною організацією операцій. Виникнення колізій пов'язано з тим, що у такому випадку неможливо знайти таке розбиття на інтервали, при якому вдасться уникнути перетину індексів інтервалів, що розташовуються поряд та не вдасться зберегти виконання рівності (3). Це означає, що не має можливості відобразити розподіл на інтервали в апаратну реалізацію без уникнення колізій доступу елементів, наприклад, у пам'яті.

Матриці перевірки парності, що мають такі характеристики, найчастіше генеруються випадковим чином без накладання обмежень на характер розташування значущих елементів. Це означає, що для частини рядків можлива паралельна реалізація, а для іншої частини це призводить до колізій. Таким чином, подібні матриці мають неоднорідну структуру з точки зору паралельного виконання обчислень.

З точки зору паралельної організації обчислень для алгоритму декодування мінімальної суми, така неоднорідність означає необхідність очікування даних перед обробкою. Наприклад, для стадії обчислення повідомлень перевірки алгоритму необхідна наявність всіх даних про елементи в рядку для обчислення мінімуму та знаку:

$$\alpha_{ij} = \prod_{j' \in I' \setminus j} \text{sign}(\beta_{ij'}) \cdot \min(|\beta_{ij'}|). \quad (6)$$

Послідовна організація запису/зчитування означає, що значно збільшується час отримання елементів у порівнянні з можливою паралельною реалізацією, відповідно до їх кількості в рядку w_{ri} .

Для реалізації обробки нерегулярних LDPC-кодів для алгоритму мінімальної суми, матриця перевірки парності характеризується подібною неоднорідністю пропонується модель організації паралельних черг запису/зчитування. Вона передбачає розбиття на $w_{r\max}$ інтервалів I_k , яке в найпростішому випадку виконується наступним чином:

$$\begin{cases} step = N / w_{r\max}; \\ j_{k\min} = \text{round}(step * (k - 1) + 1); \\ j_{k\max} = \text{round}(j_{k\min} + step); \\ k = 1 \dots r_{r\max}, \end{cases} \quad (7)$$

де $step$ – крок збільшення індексів інтервалу;

$round$ – функція округлення до найближчого цілого.

Вхідними даними виступають адреси, за якими слід виконати необхідні операції, дані для запису, за необхідності та вхід вибору операції. Виходом моделі є вихід, що сигналізує про завершення виконання операції, а також зчитані дані при виконанні операції зчитування.

За обслуговування елементів інтервалу відповідає співставлений йому модуль виконання запису/зчитування. Кожен такий модуль має доступ до окремого елементу пам'яті, в якому міститься інформація про результати обчислень. Приналежність елементу до певного блоку обслуговування визначається на основі входження індексу елементу до інтервалу I_k . Для представлення приналежності вхідних даних для блоку використовується шаблон приналежності – бітовий рядок, в якому приналежність даних до інтервалу визначається як «1», а відсутність приналежності – як «0»:

$$\text{pattern}_{ij}(u) = \begin{cases} 1, j \in I_k; \\ 0, j \notin I_k. \end{cases} \quad (8)$$

Довжина кожного бітового рядка становить $w_{r\max}$. Такий бітовий рядок формується для кожного значущого елемента рядку, а у випадку, якщо кількість значущих елементів у рядку менша за $w_{r\max}$, то бракуюча кількість елементів заповнюється лише нулями. Шаблони приналежності зберігаються в пам'яті; кількість блоків пам'яті для збереження шаблонів – $w_{r\max}$.

Одночасно з подачею вхідних даних на блок подається шаблон відповідності, на основі якого він визначає дані призначені для його обробки. На основі отриманого шаблону формується черга обробки. Всі

елементи черги мають отримати обслуговування, а відмови відсутні. Час роботи моделі – дискретний. Таким чином, модель паралельних черг представляє собою систему масового обслуговування (СМО) без відмов з дискретним часом.

Результатом операції запису є подача службового сигналу про завершення виконання операції. В той же час, у результаті виконання зчитування необхідно подати отримані дані для обробки в основному циклі. Для цього слугує мережа комутації вихідних сигналів блоків обробки. Мережа має $w_{r,max}^2$ входів та $w_{r,max}$ виходів даних. Також на вхід мережі подаються значення сигналів шаблонів, які для кожної ітерації i групуються наступним чином:

$$pattern_{netwj}(d) = pattern_{id}(j); d = 1 \dots w_{r,max}. \quad (9)$$

Таким чином, у згрупованому сигналі $pattern_{netwj}$ зустрічається максимум одна одиниця, порядковий номер якої в бітовому рядку визначає, як саме виконати з'єднання між входами та виходами. Номер одиниці в рядку позначає, який сигнал з блоків обробки необхідно зчитати для подачі на конкретний вихід. У випадку роботи лише з однією матрицею, мережа комутації може бути спрощена шляхом дослідження інтервалів та відкиданням зв'язків, що не використовуються.

Обробка черги для кожного блоку відбувається послідовно. За один елементарний дискретний інтервал часу можливе обслуговування лише одного блоку даних. Обробка даних всією системою вважається завершеною лише тоді, коли завершена обробка для всіх модулів. Загальний час обробки окремого блоку залежить від кількості елементів, що необхідно обробити. Тому степінь паралельності може змінюватися від повністю паралельної до послідовної, в залежності від розташування значущих елементів. Тому вибір на користь використання моделі паралельних черг слід робити з урахуванням того, чи дозволить це підвищити пропускну здатність декодера. Час обробки для моделі паралельних черг для одного циклу ітерації визначиться як

$$T_{queuei} = T_{queue_aux} + T_{queue_servicei} \quad (10)$$

де T_{queuei} – загальний час обробки;

T_{queue_aux} – час на виконання допоміжних дій (подача сигналів тощо), постійне значення для кожного циклу;

$T_{queue_servicei}$ – час безпосереднього обслуговування, який визначається як максимальна кількість елементів, призначена для обслуговування одним блоком:

$$T_{queue_servicei} = \max_j(count_ones(pattern_{ij})), \quad (11)$$

де $count_ones$ – функція, що повертає кількість одиниць у бітовому рядку. Середній час для повної ітерації декодування на основі моделі паралельних черг визначить як

$$T_{queue_servicei_mean} = T_{queue_aux} + M(T_{queue_servicei}), \quad (12)$$

де $M(T_{queue_servicei})$ – математичне сподівання кількості елементів на одному циклі. У тому випадку, якщо час обробки для моделі паралельних черг виявиться меншим за час послідовної обробки

$$T_{queue_servicei_mean} < T_{sequential} \quad (13)$$

то доцільним є використання запропонованої моделі.

Перевагою даної моделі з точки зору проведення реконфігурації декодера є те, що зміна матриці перевірки парності може проводитися без внесення додаткових модифікацій до структури блоку, завдяки тому, що принцип обробки елементів перенесений на обробку шаблонів приналежності. Це означає, що можливості паралельного виконання операцій використовуються відповідно до розташування елементів. Основною умовою до матриць є те, щоб кількість значущих елементів у рядках матриці не перевищувала $w_{r,max}$.

Для перевірки розробленої моделі в роботі виконана реалізація декодера з використанням моделі паралельних черг запису/зчитування на основі мови схемотехнічного опису VHDL у середовищі розробки Altera Quartus II 13.1 Web Edition. Основою для декодера обрана мікросхема ПЛІС Altera Cyclone IV EP4CGX150DF31I7AD. Реалізація декодера проводилась для матриці перевірки парності з параметрами $M=500$, $N=1000$, $w_{r,max}=8$. Візуалізація значущих елементів матриці представлена на рис. 1, де точкам відповідають значення 1.

Використання основних ресурсів ПЛІС для реалізації наведено у табл. 1.

Таблиця 1

Використання основних ресурсів для реалізації декодера

Ресурс	Використано, од.	Використано, %
Логічних елементів	4582	3
Комбінаційних елементів	2761	2
Регістрів	3937	3
Бітів пам'яті	86016	1

У табл. 2 наведені значення тактових частот, на яких може працювати мікросхема за граничних температурних умов.

Таблиця 2

Максимальні тактові частоти схеми декодеру за граничних температурних умов

Температура, °C	Тактова частота, МГц
100	198,37
-40	220,22

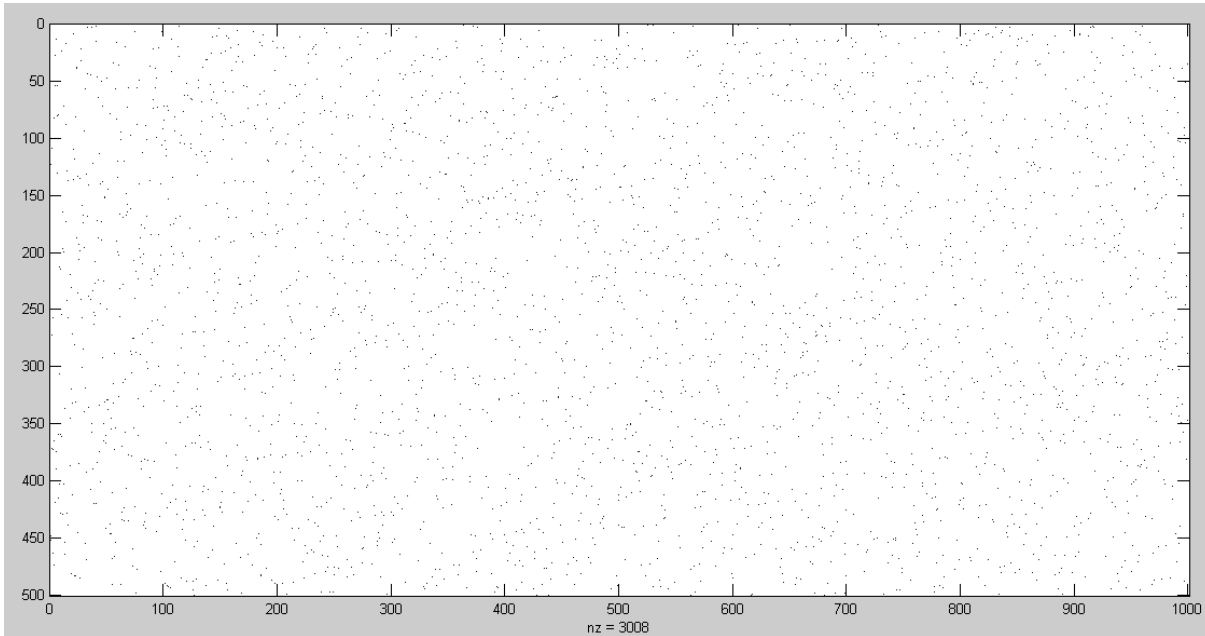


Рис. 1. Візуалізація значущих елементів матриці

Для обраної матриці реалізовано декодер з послідовним записом/зчитуванням. Реалізація показала, що на зчитування всіх необхідних даних для обробки рядка необхідно 11 тактів, в той час як за (12) середня кількість тактів на обробку становила 11,178. Такий показник пояснюється необхідністю подачі додаткових сигналів. В той же час показник $M(T_{queue_service})$ становив лише 2,178, що значно менше 8, які відповідають максимальній кількості елементів у рядку. Тому подальшим ресурсом для підвищення продуктивності декодеру є максимальне зменшення службових сигналів для розробленого блоку.

Висновки

В роботі запропонована модель паралельних черг запису/зчитування для реалізації операцій у частково паралельному LDPC-декодері, що працює за алгоритмом мінімальної суми та дозволяє використати можливості для паралельної організації операцій. Практична реалізація для обраної матриці перевірки парності показала суттєвий ресурс для підвищення швидкодії декодеру. Запропонована модель фактично представляє собою СМО, особливістю якої є використання шаблонів приналежності, що визначають, як саме необхідно проводити обробку вхідних даних. Використання шаблонів значно спрощує реалізацію реконфігурації декодеру з точки зору використання різних матриць перевірки парності при використанні моделі.

Література

1. Gomes, M., Falcao, G., Silva, V., Ferreira, V., Sengo, A., Falcao, M. Flexible Parallel Architecture for DVB-S2 LDPC-Decoders / IEEE GLOBECOM 2007 proceedings. – 2007. – P. 3265 – 3269.
2. Tang, J., Bhatt, T., Sundaramurthy, V. Reconfigurable shuffle network design in LDPC Decoders / IEEE Application-specific Systems, Architectures, and Processors. – 2006.

References

1. Gomes, M., Falcao, G., Silva, V., Ferreira, V., Sengo, A., Falcao, M. Flexible Parallel Architecture for DVB-S2 LDPC-Decoders / IEEE GLOBECOM 2007 proceedings. – 2007. – P. 3265 – 3269.
2. Tang, J., Bhatt, T., Sundaramurthy, V. Reconfigurable shuffle network design in LDPC Decoders / IEEE Application-specific Systems, Architectures, and Processors. – 2006.

Рецензія/Peer review : 6.10.2014 р.

Надрукована/Printed : 16.11.2014 р.