УДК 004.056.5 DOI: 10.18372/2073-4751.84.20895

Козачек М.С.,
orcid.org/0009-0007-4126-8520,
kozachekns@gmail.com,
Верба О.А., к.т.н.,
orcid.org/0000-0001-5752-5121,
olverba@gmail.com,
Гуцуляк Н.А.,
orcid.org/0009-0009-0472-9695,
nyancatandme0@gmail.com.

МЕТОД ПРИСКОРЕНОГО ОБЧИСЛЕННЯ ЕКСПОНЕНТИ НА ПОЛІ ГАЛУА КОМБІНУВАННЯМ ПАРАЛЕЛЬНОЇ ОБРОБКИ ТА ПЕРЕДОБЧИСЛЕНЬ

НТУУ "Київський політехнічний інститут ім. Ігоря Сікорського"

Вступ

Досягнуте в останні роки, якісне зростання технічних характеристик Інтернету резонує розширення впровадження технології бездротової передачі даних в системи контролю та управління об'єктами реального світу. Це зумовлено малою вартістю передачі даних через Інтернет, високою швидкістю, а також простотою конфігурації чи реконфігурації систем за рахунок використання готових інфраструктурних рішень.

Разом з тим, використання потенційно відкритого середовища для віддаленого контролю об'єктів реального світу несе загрозу несанкціонованого втручання в процес управління такими об'єктами. Особливо гостро ця проблема стоїть в системах віддаленого управління критичною інфраструктурою, енергетикою, хімічною промисловістю, транспортом та об'єктами медичної чи військової сфери.

Один з можливих підходів до прискорення експоненціювання на малопотужних термінальних пристроях, полягає в переході до альтернативної алгебри кінцевих полів Галуа, виконання мультиплікативних операцій в якій, в порівнянні з класичною алгеброю, здійснюється значно простіше. Ще одна перевага експоненціювання на кінцевих полях Галуа, полягає в можливості реалізації паралельного обчислення експоненти на багатоядерних процесорах, за рахунок використання специфічних

Для протидії зовнішнього втручання в роботу таких систем використовується весь наявний арсенал захисту даних, зокрема, механізм цифрового підпису. В основі цього механізму лежить криптографія з відкритим ключем, базовою обчислювальною операцією якої є модулярне експоненціювання $A^E \bmod M$. Для більшості практичних застосувань NIST рекомендує використовувати розрядність n чисел не менше 4096 біт[1].

При реалізації модулярної експоненти над числами такої розрядності, обчислення потребує виконання сотень мільйонів процесорних операцій. Враховуючи що управління та контроль віддаленими об'єктами здійснюється режимі реального часу, висуваються жорсткі вимоги до швидкості реалізації механізмів криптографії з відкритим ключем.

особливостей цієї алгебри [2]. Враховуючи тенденцію динамічного збільшення кількості ядер в сучасних термінальних мікроконтролерах (ТМК), об'єктивно виникає потреба в створенні нових методів прискорення обчислення модулярної експоненти, спеціалізованих для такого виду мікроконтролерів.

Таким чином, наукова задача прискорення комп'ютерної реалізації експоненціювання на полях Галуа за рахунок організації її паралельного обчислення на багатоядерних термінальних платформах є актуальною для сучасного етапу розвитку інформаційних технологій.

Аналітичний огляд методів паралельного обчислення експоненти на полі Галуа

Кінцеві поля Галуа GF(2^n) знаходять широке застосування в багатьох сферах інформаційних технологій, особливо в криптографії[3]. Зокрема, на їх основі побудовано стандартизований алгоритм симетричного шифрування AES, генератори псевдовипадкових послідовностей для потокових шифрів та алгоритми на основі еліптичної криптографії [3].

Поле Галуа GF(2^n), утворене простим поліномом $P(x) = x^n+p_{n-1}\cdot x^{n-1}+p_{n-2}\cdot x^{n-2}+\ldots +p_1\cdot x+ p_0$, "$j$ $\in$ {0,1,…,n-1}: $p_j \in \{0,1\}$ двійковому представленню якого відповідає число $P = 2^n+p_{n-1}\cdot x^{n-1}+p_{n-2}\cdot x^{n-2}+\ldots+ p_1\cdot x+ p_0$, включає в себе скінченну множину 2^n-1 n-розрядних чисел [4].

Виконання базових математичних операцій на таких полях відрізняється від класичної алгебри. Так, якщо в модулярній алгебрі використовується арифметичне додавання, то на кінцевих полях аналогічна операція виконується без переносів та позначається символом '$\oplus$'. Аналогом класичної операції множення виступає поліноміальне множення, що позначається як '$\cdot$'. Редукція числа A на кінцевому полі Галуа утвореному поліномом P, що позначається як A rem P та зводиться до обчислення залишку від поліноміального ділення поліному числа $A(x)$ на утворюючий поліном поля $P(x)$.

Обчислення експоненти A^E rem P, де $E = e_{n-1}\cdot 2^{n-1}+ \ldots + e_2\cdot 2^2+ e_1\cdot 2+ e_0$, $\forall$ j $\in$ {0,1,..,n-1}: e_n Î{0,1} на кінцевому полі Галуа, як і в модулярній алгебрі, виконується за однією з двох варіацій класичного алгоритму [4], що відрізняються напрямком аналізу бітів експоненти.

В варіації алгоритму експоненціювання, з аналізом бітів від молодших до старших, виконується n циклів, лічильник j в яких змінюється від нуля до n. Ця варіація використовує дві змінні R та D для обчислення результату, початкове значення яких встановлюється в одиницю та A, відповідно: R=1, $D = A$. На кожному j-тому циклі, якщо поточний e_j розряд експоненти E дорівнює одиниці, в змінну R обчислюється добуток R та D на кінцевому полі Галуа: $R = R \cdot D$ rem P. В рамках того ж циклу, змінна D підноситься до квадрату на полі Галуа: $D = D^2$ rem P. Після виконання всіх n циклів, в змінній R формується результат експоненціювання A^E rem P.

В варіації алгоритму експоненціювання з аналізом біт від старших до молодших також виконується n циклів, проте лічильник j змінюється від n до нуля. Ще одна відмінність полягає в використанні однієї змінної R, яка на початку встановлюється в одиницю: R=1. В рамках кожного з n циклів, змінна R підноситься до квадрату на полі Галуа: $R = R^2$ rem P, та якщо поточний e_j біт експоненти E рівний одиниці – множиться на число A: $R = R \cdot A$ rem P. В кінці виконання n циклів алгоритму, результат обчислення A^E rem P сформовано в змінній R.

Аналіз обох варіацій алгоритму експоненціювання на полі Галуа показує, що вони носять строго послідовний характер і тому, до певного

часу, не могли бути розпаралелені. В зв'язку з цим, основні зусилля дослідників були покладені на прискорення складових експоненціювання на полі Галуа, а саме –множення та піднесення до квадрату.

Операція множення чи піднесення до квадрату на полі Галуа, як і в модулярній алгебрі, складається з виконання двох фаз: самої операції та редукції отриманого результату. При чому, ці фази можуть виконуватися як послідовно, так і суміщено в часі.

Серед відомих підходів до суміщення в часі виконання множення та редукції в модулярній алгебрі найбільш часто використовується метод [5] модулярного множення $A \cdot B \bmod M$ Пітера Монтгомері. Ідея цього методу полягає в виконанні рівно n циклів, в рамках кожного з яких, якщо поточний біт множника B рівний одиниці – до результату додається множене A, та якщо поточний результат непарний – додається модуль M. В кінці кожного з n циклів, результат зсувається праворуч на один біт. Сформований добуток $A \cdot B \bmod M$ потребує додаткової корекції, проте вона виконується один раз в кінці циклу експоненціювання та фактично не впливає на час обчислення експоненти. Така організація суміщення операцій дозволила виконувати множення та редукцію з використанням $2 \cdot n^2$ адитивних операцій над довгими, n розрядними числами, що робить метод множення Монтгомері одним з найбільш ефективних методів прискорення модулярного множення.

В роботі [6] запропоновано адаптацію методу модулярного множення Монтгомері для полів Галуа. Цей метод, як і класичний, дозволяє реалізувати суміщене в часі виконання множення та редукції за $2 \cdot n^2$ логічних операцій над довгими числами на полі Галуа. Проте, як і в класичному методі, результат потребує корекції.

Ще один можливий варіант суміщення мультиплікативних операцій та редукції на полі Галуа, розроблений в роботі [7]. В цьому методі обчислення добутку $A \cdot B \bmod P$ суміщеного з редукцією на полі Галуа організовано в вигляді n циклів, в кожному з яких лічильник j змінюється від n до 0. В рамках кожного з n циклів, виконується зсув поточного результату ліворуч, додавання до нього: множеного A, якщо поточний b_j розряд множника B рівний нулю та додавання числового представлення поліному P, якщо n+1 – біт результату рівний одиниці. Відповідно, обчислення добутку на полі Галуа, в середньому, потребує виконання $2 \cdot n^2$ логічних операцій над n розрядними числами.

В роботі [8] запропоновано метод, який прискорює суміщене множення та редукцію на полі Галуа, за рахунок одночасної обробки ε розрядів множника, при використанні різновиду алгоритму експоненціювання з старших розрядів, де A – постійне, в рамках одного обчислення експоненти, число. Такий підхід дозволяє прискорити суміщене множення на полі Галуа у ε разів, за рахунок використання передобчислених значень A в ступенях $2,3,4,\ldots 2^{\varepsilon}-1$ на полі Галуа. При чому, оскільки, операція множення становить одну третину загальної кількості операцій алгоритму експоненціювання, реальне прискорення складає не більше ніж 1.5 раз.

Для обчислення квадрату числа на полі Галуа, в роботі [9] запропоновано використання специфічної властивості цієї алгебри. Ця властивість полягає в альтернативному обчислені квадрату, що зводиться до вставки нулів між розрядами числа. Використання цієї особливості в роботі [10] дозволило зменшити кількість обчислень до $0.86 \times n^2$ операцій над довгими числами.

В роботі [11] представлено метод суміщення в часі мультиплікативних операцій множення та піднесення до квадрату на полі Галуа. Завдяки використанню особливості

альтернативного обчислення квадрату на полях Галуа, в роботі [11] досягнуто прискорення обчислення експоненти в 4.5 раз, в порівнянні з класичним алгоритмом.

Поява і швидкий розвиток багатоядерних термінальних мікроконтролерів педалює активну розробку методів паралельного обчислення експоненти на таких платформах. Всі ці методи в своїй основі використовують властивість полів Галуа[10], яка дозволяє підносити число в будь-який ступінь двійки за однаковий час завдяки використанню передобчислень залежних тільки від утворюючого поліному.

В роботі [2] запропоновано метод паралельного обчислення експоненти на полі Галуа з використанням алгоритму з аналізом зі старших розрядів на h-ядерному процесорі. Ідея цього методу полягає в розподілі коду експоненти E на h адитивних складових $E = E_1+E_2+\ldots+E_h$, кожна з яких утворює часткову експоненту, до якої входить n/h значущий бітів коду E, відстань між якими складає h-1 біт. В рамках кожного з h незалежних обчислюваних процесів обробляється часткова експонента, порядковий номер якої відповідає номеру ядра. Обробка кожної часткової експоненти включає в себе n/h циклів, в рамках кожного з яких виконується піднесення поточного результату в ступінь 2^h з використанням передобчислень, та якщо поточний біт часткової експоненти рівний одиниці – множення на полях Галуа на постійне, в межах одного експоненціювання, число A. Результат обчислення A^E rem P формується на h-тому ядрі, як добуток на полі Галуа всіх результатів роботи ядер. Відомий метод [2] дозволяє прискорити обчислення експоненти в h разів, завдяки паралельному виконанню операцій множення та одночасному піднесенню в ступінь 2^h.

Ще один підхід до паралельного обчислення експоненти на полі Галуа полягає в груповому розподілі її n/h суміжний бітів на h адитивних часткових експонент. Такий метод, як і попередній, передбачає, що кожен з h незалежних потоків обчислює часткову експоненту, відповідну своєму порядковому номеру. Втім, обчислення цієї часткової експоненти виконується за n/h циклів класичного алгоритму експоненціювання з старших розрядів, після чого, отриманий результат підноситься в ступінь двійки, різний для кожного ядра, з використанням передобчислень. Переваги такого підходу полягають в можливості використання додаткових резервів передобчислень, за рахунок обробки групи суміжних бітів.

Таким чином, проведений аналітичний огляд методів показав, що використані ще не всі резерви прискорення паралельного обчислення експоненти на полях Галуа.

Мета досліджень

Мета дослідження полягає в прискоренні комп'ютерної реалізації експоненціювання на полях Галуа – базової операції криптографії з відкритим ключем, за рахунок організації її паралельного обчислення на багатоядерних термінальних платформах з використанням комплексу передобчислень.

Організація паралельного обчислення експоненти на полі Галуа з використанням передобчислень

Для досягнення поставленої мети пропонується організація фрагментарного обчислення експоненти A^E rem P на полі Галуа GF(2^n) в вигляді h незалежних потоків на h ядерному процесорі. При такій організації n-розрядний код експоненти E умовно розділяється на n/k фрагментів, по k біт в кожному. При чому, кожен з фрагментів ділиться на h зон, по k/h біт в кожній.

Наприклад, якщо код експоненти E = 241591, двійковому представленню якого відповідає E = 0011 1010 1111

1011 0111_2, кількість ядер h = 2, а кількість біт в фрагменті k = 6, то розподіл бітів експоненти можна представити у вигляді, зображеному на рисунку 1.

Такий поділ бітів експоненти E на фрагменти дозволяє рівномірно розподілити обчислення між всіма ядрами, шляхом обробки на кожному ядрі зон, що відповідають його номеру. Так, в рамках наведеного прикладу, перше ядро обробляє перші зони, що рівноцінне обчисленню експоненти $A^{E1} remP$, де E_1 = 111 000 111 000 110 000.

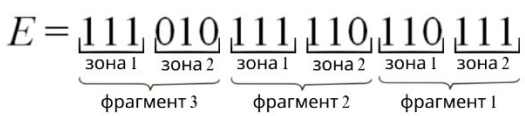

Рисунок 1. Умовний розподіл бітів експоненти E = 241591, при k= 6 та h = 2.

Паралельно з цим, друге ядро обробляє всі другі зони, тобто обчислює $A^{E2} remP$, де E_2 = 010 000 110 000 111.

Таким чином, кожне s-те ядро, де $s \in \{1,2,\ldots,h\}$ обчислює свою $A^{Es} remP$ експоненту, після чого, результат роботи ядер об'єднується на h-тому ядрі в вигляді добутку обчислених результатів на кінцевому полі Галуа.

$$A^{E} rem P = (\bigotimes_{i=1}^{h} A^{Ei} rem P) rem P \quad (1)$$

Запропонована організація обчислення експонент $A^{E1} remP$, $A^{E2} remP$,…, $A^{Eh} remP$ передбачає використання модифікованої варіації алгоритму експоненціювання зі старших розрядів. Така модифікація полягає в виконанні n/k циклів, в кожному з яких операція піднесення до квадрату замінена піднесенням в ступінь 2^k, а множення на значення постійного, в рамках обчислення експоненти числа A, замінене на одночасну обробку k розрядів фрагменту.

Для ефективного піднесення в ступінь 2^k використовується таблиця T передобчислень, що дозволяє підносити будь-яке число в ступінь двійки за однаковий час. Процедура побудови та використання такої таблиці описана в [12]. Формування таблиці T проводиться один раз на стадії вибору утворюючого поліному $P(x)$, що на практиці є компонентою відкритого ключа криптосистеми, а значить змінюється відносно рідко. З цього випливає, що час обрахунку таблиці T практично не вливає на часову ефективність паралельного експоненціювання на кінцевому полі Галуа запропонованимго методом. Побудована при зміні відкритого ключа криптосистеми таблиці передобчислень T зберігається в програмованій енергонезалежній пам'яті термінального мікроконтролера.

В якості вхідних параметрів процедура побудови таблиці T передобчислень використовує значення P та ступінь m двійки в яку підноситься число за допомогою сформованої таблиці передобчислень. Зокрема, в запропонованій організації експоненціювання в якості параметру m використовується значення k, а саме розмір фрагменту.

При покроковому виконанні процедури формування таблиці T передобчислень для швидкого піднесення в ступінь двійки з вхідним параметром $m = k$ = 6 та значенням утворюючого поліному $P(x)=x^{18}+x^{17}+x^{14}+x^{13}+x^{12}+x^{9}+x^{8}+x^{7}+x^{6}+x^{3}+x^{2}+x+1$, десятковому представленню якого відповідає число P = 422863, сформована таблиця T набуває вигляду, зафіксованого в таблиці.1.

Таблиця 1. Передобчислення для швидкого піднесення числа в ступінь 64 $=2^6$ на кінцевому полі Галуа з значенням

j	*T*[*j*]	*j*	*T*[*j*]	*j*	*T*[*j*]
0	1	6	202002	12	85214
1	21143	7	160858	13	190911
2	109570	8	830	14	120491
3	232139	9	11533	15	176004
4	225195	10	179617	16	226971
5	2986	11	13264	17	259172

P = 422863 та m = 6.

Одночасна обробка k розрядів коду фрагменту передбачає використання кожним s-тим ядром таблиці T_s передобчислень, ідея якої полягає в попередньому обчисленні експонент A в ступенях всіх можливих $2^{k/h}$ -1 комбінацій коду s-тої зони на кінцевому полі Галуа.

Формування таких таблиць T_s передобчислень проводиться на кожному ядрі перед початком обчислення експоненти A^E rem P та передбачає виконання приведеної нижче розробленої процедури з вхідними параметрами A, E та P:

1. Перший елемент таблиці $T_s[1]$ обчислюється як значення числа A піднесеного в ступінь $= 2^{k-(h-s)/h}$ на кінцевому полі Галуа GF(2^n): T[1]=$A^{\cdot}$ rem P. .
2. Індексу j поточного номеру елементу таблиці T_s присвоюється значення двійки: j = 2.
3. Елемент таблиці T_s передобчислень під номером j формується як добуток першого $T_s[1]$ та попереднього $T_s[j\text{-}1]$ елементу цієї таблиці на кінцевому полі Галуа GF(2^n) наступним чином:
$T_s[j] = (T_s[1] \otimes T_s[j-1]) rem P$.
4. Індекс j збільшується на одиницю: $j = j+1$. Якщо значення j менше $2^{k/h}$ ($j < 2^{k/h}$), то здійснюється перехід на повторне виконання п.3.

Розроблена процедура формування таблиці T_s передобчислень на першому (s = 1) з двох ядер (h = 2) може бути проілюстрована в вигляді наступного прикладу. Нехай, розмір фрагменту k = 6, P = 422863 та експонента E = 241591 залишаються незмінними.

В рамках виконання п.1 запропонованої процедури формування таблиці T_s, на першому ядрі перший елемент таблиці $T_1[1]$ обчислюється як число A = 192336 піднесене до квадрату 6/2·1 = 3 раз: $T_1[1] = 192336^{2^3} rem 422863 = 153161$. Згідно п.2, лічильник j встановлюється в двійку: j = 2. Відповідно до виконання п.3, другий (j = 2) елемент таблиці T_1 обчислюється наступним чином: $T_1[2] = T_1[1] \otimes T_1[2\text{-}1]$ rem 422863=182781. На п.4, запропонованої процедури, лічильник поточного елементу таблиці збільшується на одиницю: j = 2+1=3, оскільки, значення j менше 2^3 (3 < 8), то здійснюється перехід на повторне виконання п.3.

На п.3 третій елемент таблиці обчислюється як поліноміальний добуток першого на другий елемент тієї ж таблиці: $T_1[3] = T_1[1] \otimes T_1[3\text{-}1]$ rem 422863 = 86909. Відповідно до п.4, лічильник j інкрементується: j = 3+1 = 4, оскільки, значення 4 < 8, то здійснюється перехід на повторне виконання п.3.

При подальшому послідовному виконанні запропонованої процедури побудови таблиці T_s передобчислень на першому ядрі, а також аналогічному виконанні п.1-4 на другому ядрі, сформовані таблиці набувають вигляду, зафіксованого в таблиці 2.

Таблиця 2. Результати передобчислень T_s для першого і другого ядра

j	$T_1[j]$	$T_2[j]$
1	153161	192336
2	182781	34063
3	86909	152395
4	109146	109146
5	213576	68430
6	252496	37524
7	124980	32347

Паралельне обчислення поліноміальної експоненти A^E rem P на h ядрах, з використанням сформованих таблиць T та T_s передобчислень, на кожному s-тому ядрі, зводиться до виконання наступної процедури:

1. Лічильнику j присвоюється номер старшого фрагменту: $j = n/k$. Змінна для збереження результату R_s встановлюється в одиницю: $R_s = 1$.
2. Значення змінної R_s підноситься в ступінь 2^k на кінцевому полі Галуа GF(2^n), згідно процедури використання таблиці T, описаної в []: $R_s^{2^k} remP$.
3. Змінна R_s поліноміально множиться на елемент таблиці T_s номер якого відповідає коду s-тої зони j-того фрагменту; позиція старшого біта цієї зони визначається як $f = k \cdot j - (k/h \cdot (s-1))$: $R_s = (R_s \urcorner T_s[2^{k/h-1} \cdot e_{f-1} + 2^{k/h-2} \cdot e_{f-2} + \ldots + e_{k/h}])remP$.
4. Лічильник j зменшується на одиницю: $j = j-1$. Якщо, значення j більше нуля ($j > 0$), то здійснюється перехід на повторне виконання п.2.
5. Якщо номер s поточного ядра не дорівнює h ($s \neq h$), то обчислений результат надсилається h-тому ядру. В іншому випадку ($s = h$), h-те ядро обчислює кінцевий результат поліноміального експоненціювання, як добуток обчисленого значення R_h та отриманих значень R_s на полі Галуа GF(2^n):

$$R_h = (R_1 \urcorner R_2 \urcorner \ldots \urcorner R_h) \text{ rem } P.$$

Роботу розробленої процедури паралельного обчислення поліноміальної експоненти можна проілюструвати на наступному прикладі. Нехай, незмінними залишаються число A = 192336, ступінь E = 241591, поліном P = 422863 та кількість ядер h = 2.

В рамках п.1, процедури паралельного експоненціювання, на першому ядрі змінна R_1 встановлюється в одиницю: $R_1 = 1$, разом з тим лічильнику j присвоюється значення старшого фрагменту: $j = 3$. На п.2 змінна R_1 підноситься в 64-тий ступінь на кінцевому полі Галуа з використанням таблиці T_1, сформованої вище: $R_1 = 1$. Відповідно до п.3 змінна R_1 множиться на елемент таблиці T_1 номер якого відповідає коду першої зони (s = 1) третього фрагменту (j=3): $R_1 = (R_1 \urcorner T_1[111_2])$ rem$P$. На п.4 лічильник j зменшується на одиницю: j = 3-1 = 2, оскільки, j>0, то здійснюється перехід на повторне виконання п.2.

При послідовному-аналогічному виконанні наступних пунктів запропонованої процедури експоненціювання, перше ядро надсилає сформований в змінній R_1 = 83721 результат роботи другому ядру. Покрокове формування цього результату, в залежності від поточного фрагменту, наведено в таблиці 3.

Паралельно з першим ядром, друге ядро аналогічно виконує п.1-4, описаної вище процедури експоненціювання, але оброблює не першу зону, а другу. Змінна результату R_2 при обробці другої зони, в залежності від поточного фрагменту наведена в таблиці 3.

В рамках п.5, процедури паралельного експоненціювання, на другому ядрі обчислюється остаточний результат, як добуток роботи двох ядер на полі Галуа: R_1 = 83721 $\urcorner$ 192921 rem 422863 = 136785.

Таблиця 3. Динаміка зміни результатів R_s в залежності від номеру j поточного фрагмента

Перше ядро		
j	піднесення в ступінь	множення
3	1	124980
2	7640	179400
1	170867	83721
Друге ядро		
j	піднесення в ступінь	множення
3	1	34063
2	156996	119435
1	49760	192921
Формування кінцевого результату		
R_1 = 83721 $\urcorner$ 192921 rem 422863 = 136785		

Таким чином, після виконання запропонованої процедури паралельного обчислення експоненти з використанням комплексних передобчислень, на першому ядрі зафіксовано результат обчислення 192336^{241591}rem 422863 = 136785.

Аналіз ефективності

З огляду на поставлену мету, в якості оцінки ефективності розробленого методу доцільно використовувати коефіцієнт прискорення ψ, що визначається як співвідношення часу обчислення експоненти на полі Галуа класичним методом до часу виконання аналогічної операції запропонованим:

$$\psi = \frac{\tau_{кл}}{\tau_з}, \quad (2)$$

де $\tau_{кл}$ – час обчислення експоненти на полі Галуа з використанням класичного методу, а $\tau_з$ – час виконання аналогічної операції запропонованим методом.

Класичне обчислення експоненти A^E rem P на кінцевому полі Галуа, передбачає виконання, в середньому: n операцій піднесення до квадрату та $0.5 \cdot n$ операцій множення, що загалом складає $1.5 \cdot n$ мультиплікативних операцій над довгими числами. Час виконання однієї такої операцій на кінцевому полі Галуа складає $2 \cdot n \cdot \mu$, де μ – тривалість виконання однієї логічної операції над n розрядними числами.

Таким чином, середній час $\tau_{кл}$ одного обчислення експоненти A^E rem P на кінцевому полі Галуа з використанням класичного методу складає $\tau_{кл} = 3 \cdot n^2 \cdot \mu$.

Обчислення аналогічної операції в запропонованому методі складається з двох етапів: формування таблиць T_s передобчислень на кожному ядрі та, власне, обчислення експоненти A^E rem P з використанням сформованих таблиць.

Виконання першого етапу передбачає побудову таблиць T_s паралельно на h ядрах, відповідно, в час $\tau_з$ слід врахувати найбільший час побудови однієї такої таблиці. В найгіршому випадку, формування таблиці T_s передбачає $(h\text{-}1)\cdot(k/h)$ піднесень до квадрату та $2^{k/h}$-1 множень, що разом складає $((h\text{-}1)\cdot(k/h)+2^{k/h}\text{-}1)\cdot 2\cdot n$ – логічних операцій над довгими числами.

На другому етапі, паралельно обчислюється h часткових експонент з використанням побудованих таблиць T_s. Формування кожної з таких експонент передбачає виконання n/k циклів, в кожному з яких: попередній результат підноситься в ступінь 2^k, з використанням таблиці T передобчислень, в середньому, за $0.5 \cdot n$ логічних операцій, та обчислюється добуток отриманого результату та елементу таблиці T_s на полі Галуа, що загально складає $(2.5 \cdot n^2)/k$ логічних операцій над довгими числами.

Формування кінцевого результату обчислення A^E rem P передбачає виконання h множень над довгими числами, проте, оскільки на практиці кількість ядер в термінальних мікроконтролерах не перевищує восьми ($h <= 8$), ці множення можна не враховувати в загальний час $\tau_з$.

Таким чином, середній час $\tau_з$ обчислення експоненти на кінцевому полі Галуа з використанням запропонованого методу оцінюється за наступною формулою:

$$\tau_з = \left(\frac{2.5 \cdot n^2}{k} + \left((h - 1) \cdot (k / h) + 2^{k/h} - 1\right) \cdot 2 \cdot n \right) \cdot \mu . \quad (3)$$

Відповідно, коефіцієнт прискорення ψ обчислюється в наступному вигляді:

$$\psi = \frac{\tau_{кл}}{\tau_з} = \frac{3 \cdot n^2 \cdot \mu}{\left(\frac{2.5 \cdot n^2}{k} + \left((h - 1) \cdot (k / h) + 2^{k/h} - 1\right) \cdot 2 \cdot n \right) \cdot \mu} . \quad (4)$$

На практиці, коефіцієнт ψ прискорення залежить від кількості h ядер та розміру k/h зони, який обмежується обсягом пам'яті, потрібним для зберігання таблиць T_s передобчислень.

Оцінка обсягу пам'яті γ, потрібного для передобчислень визначається наступним чином. Одна таблиця складається з $2^{k/h}$-1 рядків, в кожному з яких зберігається n розрядне число, тобто загальний обсяг пам'яті для такої таблиці складає $(2^{k/h}-1)\cdot n$ біт. Враховуючи, що така таблиця будується на кожному ядрі, обсяг пам'яті оцінюється в $\gamma = (2^{k/h}-1)\cdot n\cdot h$ біт.

Для практичних застосувань, значення коефіцієнту ψ, в залежності від кількості ядер та розміру зони наведено в таблиці 4.

Таблиця 4. Значення коефіцієнту прискорення ψ та обсягу пам'яті γ, в залежності від значення h та k/h, при n = 4096

двох-ядерний процесор					
k/h	2	3	4	5	6
ψ	4.78	7.12	9.32	11.21	12.4
γ	2^{15}	2^{16}	2^{17}	2^{18}	2^{19}
чотирьох-ядерний процесор					
k/h	2	3	4	5	6
ψ	9.47	13.88	17.71	20.34	20.87
γ	2^{16}	2^{17}	2^{18}	2^{19}	2^{20}
восьми-ядерний процесор					
k/h	2	3	4	5	6
ψ	18.23	25.46	30.27	31.67	29.03
γ	2^{17}	2^{18}	2^{19}	2^{20}	2^{21}

З таблиці 4 видно, що оптимальне значення розміру зони k/h, в середньому, дорівнює п'яти. При обчисленні експоненти на чотирьох ядерному (h = 4) процесорі з використанням запропонованого методу, коефіцієнт прискорення ψ дорівнює 20.34, а об'єм передобчислень T_s складає 2^{16} байт.

Висновки

В результаті проведених досліджень, спрямованих на прискорення обчислення експоненти на полях Галуа, шляхом розпаралелення цієї операції на h-ядерних процесорах, отримано наступні результати.

Теоретично обґрунтовано, розроблено та досліджено метод паралельного обчислення експоненти A^E rem P на полі Галуа з аналізом розрядів ступеня від старших до молодших на h-ядерних процесорах, який відрізняється тим, що в рамках кожного з h незалежних обчислюваних процесів пофрагментно оброблюється по k розрядів коду експоненти, причому, ця обробка включає в себе формування добутку результату табличного піднесення поточного коду експоненти в ступінь 2^k та попередньо обчисленої експоненти A в ступені, яка дорівнює коду k/h розрядів фрагменту, позиція яких, визначається номером процесу, що дозволяє підвищити швидкодію, в порівнянні з відомими методами, за рахунок максимального використання передобчислень.

Експериментально підтверджено, що на чотирьох ядерному процесорі та розміру фрагменту в двадцять розрядів досягається прискорення в 20.34 рази, в порівнянні з класичним методом експоненціювання на полях Галуа.

Розроблений метод орієнтовано на використання в системах контролю та управлінні віддаленими об'єктами, обмін даними в яких здійснюється за допомогою технології Інтернету Речей на багатоядерних термінальних мікроконтролерах.

Література

1. Ruiz A.L. Two Galois Fields Cryptographic Applications / A.L. Ruiz, E. Castillo, L. Parrilla, A. Garsia // Aritmetic and Algebraic Circuits.- 2021.- P.551-565. DOI: 10. 1007/978-3-030-67266-9_12

2. Марковський О.П. Метод розподіленого обчислення на багатоядерних процесорах експоненти на полях Галуа для криптографічних застосувань/ О.П. Марковський, С.С.Нікольський // Проблеми

управління та інформатизації.- 2025.- № 1 (81).- С.59-71. DOI: 10.18372/2073-4751.81.20130

3. Hachenbergen D., Topic in Galois Fields. / D. Hachenbergen, D. Jungnickel // AACIM.-Vol.29.- Springer International Publishing.- 2020–P.785. DOI: 10.1007/978-3-030-60806-4.

4. Schneier B. Appllied Cryptography. Protocols, Algorithms and Source Code in C / B. Schneier// Wiley.-2015.-P.784.

5. Montgomery P. Modular multiplication without trial division / P. Montgomery // Mathematics of Computation. – Vol. 44.- №170. – 1985. – P. 519–521.

6. Марковський О.П. Метод швидкого експоненціювання на полях Галуа для систем криптографічного захисту інформації / О.П. Марковський, І.В. Дайко // Проблеми управління та інформатизації.- 2024.- № 1 (77).- С.80-88. DOI: 10.18372/2073-4751.77.18660

7. Al-Mrayt Ghassan Abdel Jalil Halil. Organization of fast exponentiation on galois fields for cryptographic data protection systems / Al-Mrayt Ghassan Abdel Jalil Halil , Markovskyi O., Stupak A. // Information, Computing and Intelligent systems. – 2022. – № 3.- P.17-25. DOI: 10.20535/ 2708-4930.3. 2022. 265480.

8. Dychka I. Method of Performing Operations on the Elements of GF(2^m) Using a Sparse Table. / Ivan Dychka, Mykola Onai, Andrii Severin, Cennuo Hu // International Journal of Computer Network and Information Security(IJCNIS),-2024.- Vol.16, - № 1,- P.61-72. DOI:10.5815/ijcnis.2024. 01. 05

9. Wu K.Optimized Design of ECC Point Multiplication Algorithm Over GF(2^m) / K. Wu, G. Wei // Proceeding of International Conference on Electronic Engineering and Informatics (EEI), Nanjing, China,- 2019.- P. 420–425. DOI: 10.1109/EEI48997.2019.00097.

10. Марковський О.П. Метод швидкого обчислення експоненти на полях Галуа GF(2^n) для криптографічних застосувань / О.П. Марковський, С.С. Нікольський // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. Луцьк, 2025.- Вип. 58. – С. 188-196. DOI: 10.36910/6775-2524-0560-2925-58-23/

11. Верба О.А. Метод експоненціювання на полях Галуа для швидкої реалізації криптографічного захисту в IoT / О.А. Верба, С.С.Нікольський // Проблеми управління та інформатизації.- 2024.- № 4 (80).- С.21-31. DOI: 10.18372/2073-4751.80.19767.

Козачек М.С., Верба О.А., Гуцуляк Н.А.

МЕТОД ПРИСКОРЕНОГО ОБЧИСЛЕННЯ ЕКСПОНЕНТИ НА ПОЛІ ГАЛУА КОМБІНУВАННЯМ ПАРАЛЕЛЬНОЇ ОБРОБКИ ТА ПЕРЕДОБЧИСЛЕНЬ

Розроблено та досліджено метод обчислення експоненти на полі Галуа, при її паралельній реалізації на h-ядерному термінальному мікроконтролері для криптографічних застосувань. Прискорення досягнуто за рахунок наступних чинників: розподілення обчислень між h-ядрами, швидкого піднесення в ступінь 2^k та одночасної обробки k розрядів коду експоненти, шляхом комплексному використанню передобчислень. Наведено математичне обґрунтування запропонованого методу, процедури його функціонування, які ілюстровані числовими прикладами.

Теоретично доведено та експериментально підтверджено, що на чотирьох ядерному процесорі та розміру фрагменту в двадцять розрядів досягається прискорення в 20.34 рази, в порівнянні з класичним методом експоненціювання на кінцевих полях Галуа.

Ключові слова: мультиплікативні операції на полях Галуа, криптографічні алгоритми на основі алгебри полів Галуа, експоненціювання на полях Галуа, паралельні обчислення, багатоядерні мікроконтролери.

Kozachek M.S., Verba O.A., Hutsuliak N.A.

METHOD FOR ACCELERATING THE CALCULATIONS OF EXPONENT ON GALOIS FIELDS BY COMBINING PARALLEL PROCESSING AND PRE-CALCULATIONS

Method for exponent calculation on a Galois Field has been developed and researched, with its parallel implementation on an h-core terminal microcontroller for cryptographic applications. The acceleration was achieved due to the following factors: distribution of calculations between h-cores, fast exponentiation to the power of 2^k, and simultaneous processing of k digits of the exponent code through the complex use of precomputations. A mathematical proof of the proposed method is given, as well as procedure its operation whose are illustrated be numerical examples.

It has been theoretically proven and experimentally confirmed that on a four-core processor and a fragment size of twenty bits, a 20.34-fold acceleration is achieved compared to the classical method of exponentiation on finite Galois fields.

Key words: *multiplication operation on Galois fields, cryptographic algorithms based on Galois Fields algebra, Galois Fields exponentiation, parallel computing, multi-core microcontrollers.*

Стаття подана до редакції: 14/11/2025
Стаття прийнята до опублікування: 28/11/2025
Стаття опублікована: 30/12/2025