УДК 004.056.5

МАРКОВСЬКИЙ О.П.,
ВИНОГРАДОВ Ю.М.,
САЛОХА О.Є.,
ТКАЧЕНКО І.М.

ЕФЕКТИВНЕ ОБЧИСЛЕННЯ КВАДРАТНОГО КОРЕНЯ НА ПОЛЯХ ГАЛУА GF(2^m)

У статті запропоновано спосіб прискореного обчислення кореня на полях Галуа GF (2^m). Показано, що задача обчислення коренів на полях Галуа може бути зведена до розв'язання системи лінійних бітових рівнянь. Запропоновано технологію реалізації цієї теоретичної ідеї. Доведено, що обчислювальна складність O*(m)* запропонованого способу істотно менша, ніж складність відомих способів, що становить O*(m^2)*.

In article, the method of accelerated calculation of square root on Galois fields GF (2^m) has been proposed. By the theoretical way, it has been shown that computing roots on Galois fields' calculation can be reduced to solving system of linear bits equations. New technology of this theoretical idea was proposed. It has been proved, that calculation complexity O*(m)* of proposed method is much smaller in comparing to known methods, which equals O*(m^2)*.

Вступ

Розв'язання алгебраїчних рівнянь на кінцевих полях і, зокрема, на полях Галуа GF GF(2^m), є однією з найважливіших задач обчислювальної алгебри і теорії чисел [1]. Перші роботи, присвячені вирішенню цієї задачі, були опубліковані більше ста років тому [2].У наш час, крім теоретичного інтересу, ця задача має велику практичну цінність, оскільки вона відіграє ключову роль в комп'ютерних технологіях корекції помилок, кодування і стиснення даних. Потужним імпульсом розвитку комп'ютерних технологій у розв'язанні таких рівнянь стало практичне використання криптографічних систем захисту інформації, що базуються на еліптичних кривих та інших різновидах Абелевих груп [1].

Найважливішою складовою частиною технологій вирішення алгебраїчних рівнянь на кінцевих полях є обчислення квадратного кореня. Типовим застосуванням операції добування кореня є стиснення і відновлення точки на еліптичній кривій [3]. Точка з координатами *(x, y)* на кривій стискається до виду *(x, β)* де $\beta \in \{0,1\}$. Для відновлення чисельного значення *y* по *(x,β),* необхідно вирішити квадратне рівняння $y^2 = P(x)$, тобто обчислити квадратний корінь $\sqrt{P(x)}$. Подібна ситуація виникає і при хешуванні на еліптичних кривих, яке застовується в ряді криптосистем [4,5].

Ефективність засобів корекції помилок, кодування і стиснення даних, а також систем криптографічного захисту інформації значною мірою визначається можливістю досягнення високої продуктивності при програмній та апаратній реалізації. Операція добування квадратного кореня на полях Галуа GF(2^m) є однією з найскладніших задач в обчислювальному плані, тому від швидкості її реалізації значною мірою залежить продуктивність зазначених вище засобів. При використанні існуючих методів добування квадратного кореня на полях GF(2^m) час виконання цієї операції пропорційний m^2.

З розвитком технологій розподілених обчислень істотно зросли можливості комп'ютерних систем, які потенційно можуть бути використані для порушення захисту. Найпростішим заходом підвищення криптостійкості систем, що ґрунтуються на використанні полів Галуа GF(2^m), є збільшення розрядності *m* чисел, що використовуються. Це значною мірою уповільнює продуктивність засобів криптографічного захисту. Тому, в сучасних умовах важливою і актуальною є проблема розробки нових підходів до прискорення програмної та апаратної реалізації операції обчислення квадратного кореня на кінцевих полях. Основним резервом зменшення обчислювальної складності цієї важливої для практичних застосувань операції є врахування особливостей її використання в реальних системах [6].

Аналіз відомих методів обчислення коренів на кінцевих полях

Практична значимість задачі обчислення квадратного кореня на кінцевих полях, особливо для систем криптографічного захисту інформації на основі еліптичних кривих, стимулює інтенсивні дослідження в області методів вирі-

шення цієї задачі. Як вже зазначалося, більше ста років тому були запропоновані два базових методи обчислення квадратного кореня на полях Галуа GF(2^m): Tonelli [2] та Cipolla [5]. Пізніше ці методи були розширені для випадку поля GF(q^m), де q – просте число і отримали відповідні назви: Tonelli-Shanks [4] і Cipolla-Lehmer [1].У 1977 році в роботі [5] метод Tonelli-Shanks був розширений для випадку добування кореня довільного ступеня. В роботі [6] був розроблений спеціалізований метод обчислення кубічного кореня, що відрізняється підвищеною швидкодією.

Базовими операціями на полях Галуа GF(2^m) є додавання та множення їх елементів. Операція додавання відповідає додаванню в поліноміальній математиці і далі позначена як '+'. Операція множення на полях GF *(2^m)* фактично складається з двох операцій: поліноміального множення (множення без переносів), позначеного далі символом '⊗' і редукції, тобто знаходження залишку від поліноміального ділення добутку на утворюючий поліном $P(x)$ поля. Операція редукції позначена далі як 'rem', на відміну від арифметичної редукції 'mod'.

Для кожного елементу поля GF(2^m), що утворюється нерозкладним поліномом *P(x)* ступеню *m,* якому відповідає число *p,* існує мультиплікативна циклічна група, порядок *n* якої не перевищує 2^m-1. Наприклад, для поля Галуа, утвореного нерозкладним поліномом $P(x) = x^4 + x^3 + 1$ ($p = 25_{10} = 11\ 001_2$) генеруються циклічні групи. Порядок циклічної групи дорівнює 2^m-1, якщо її генератор не має спільних дільників з 2^m-1.

У кожній циклічній групі поля GF(2^m) може бути виділена циклічна підгрупа, кожен елемент якої є квадратом попередньої. При цьому порядок кожної з квадратичної підгруп не перевищує $log_2 n$, тобто менше або дорівнює m.

Основна ідея знаходження квадратного кореня $\sqrt{A}$ на полях Галуа GF(2^m) методом Tonelli-Shanks полягає в проходженні квадратичної циклічної підгрупи до знаходження її елемента, що передує шуканому. У процедурному значенні прохід по квадратичній циклічній підгрупі еквівалентний операції експоненціювання[10].

$$B = A^{2^{m-1}} rem(p). \qquad (1)$$

Таким чином, ідея добування квадратного кореня на полях GF(2^m) теоретично є досить простою, однак її практична реалізація пов'язана зі значними витратами обчислювальних ресурсів, оскільки обчислення (1) передбачає виконання m-1 операцій піднесення до квадрату і редукції.

Операції піднесення до квадрату виконуються за правилами поліноміального множення, тобто без урахування переносів. Операція піднесення до квадрату може бути ефективно реалізована з використанням важливої властивості: в двійковій формі представлення квадрата числа A розряди, що знаходяться на парних позиціях, дорівнюють нулю, а непарні розряди дорівнюють відповідним розрядам числа A, тобто, якщо $A=a_0+a_1\cdot 2+a_2\cdot 2^2+...+a_{m-1}\cdot 2^{m-1}$, то:

$$A \otimes A = A^2 = a_0 + a_1 \cdot 2 + a_2 \cdot 2^2 + ... + a_{m-1} \cdot 2^{2m-2}. \qquad (2)$$

Найважливішим наслідком цієї властивості є той факт, що обчислення поліноміального квадрату не вимагає для своєї реалізації ніяких обчислювальних операцій, а зводиться лише до перестановки розрядів вихідного числа [7].

При оцінці обчислювальної складності операції добування квадратного кореня за методом Tonelli-Shanks слід враховувати те, що на практиці розрядність елементів поля m істотно перевищує розрядність процесора w, тому при виконанні операцій елементи поля розбиваються на s секцій *(s = m/w).*

Операція редукції, тобто приведення результату поліноміального множення в рамки поля Галуа виконується шляхом обчислення залишку від поліноміального ділення *(2·m-1)*-розрядного результату поліноміального піднесення до квадрату на *(m+1)*-розрядний код утворюючого поліному поля Галуа . Реалізація операції поліноміального ділення включає виконання *(m-1)* циклів, в кожному з яких здійснюється зсув на один розряд *(m+1)*- розрядного коду p і логічне додавання його з кодом поточного залишку в разі, якщо старший розряд останнього дорівнює одиниці. Для зсуву *(m+1)*- розрядного коду p на один розряд необхідно виконати *(s+1)* процесорних операцій зсуву. Так як ця операція виконується в кожному з *(m - 1)* циклів редукції, то сумарна кількість процесорних операцій зсуву становить *(s+1)·(m-1)*. Виходячи з того, що в процесі редукції операція додавання виконується, в середньому, в половині циклів поліноміального ділення, то середня кількість таких операцій становить *(m-1)/2*. Беручи до уваги, що для реалізації цієї операції на w- розрядному процесорі треба виконати *(s+1)* процесорних операцій логічного додавання для редукції результату множення складає: *(s+1)·(m-1)/2*. Відповідно, середній час виконання однієї ре-

дукції результату піднесення до квадрату становить $1.5\cdot(s+1)\cdot(m-1)\cdot\tau$, де τ-час виконання на процесорі логічної операції. Враховуючи, що вилучення квадратного кореня вимагає $(m-1)$ операції зведення в квадрат, середнє число N_T логічних операцій, потрібних для добування квадратного кореня на полі GF(2^m) становить:

$$N_T = 1.5\cdot(s+1)\cdot(m-1)^2. \qquad (3).$$

Аналогічна оцінка обчислювальної складності $O(m^2)$ наведена в [1] і для методу Cipolla-Lehmer. У сучасних умовах збільшення продуктивності розподілених комп'ютерних систем, які потенційно можуть бути використані для порушення захисту, найбільш простим способом підвищення крипостійкості є збільшення розрядності хисел. При цьому, як випливає з (3) квадратично зростає обчислювальна складність реалізації операції отримання квадратного кореня на полях GF(2^m).

Відомі методи обчислення квадратного кореня на полях Галуа GF(2^m) розглядають цю важливу для практики задачу незалежно від особливостей її практичного використання в реальних криптосистемах. Разом з тим, особливості практичного використання операції дозволяють істотно зменшити її обчислювальну складність.

Метою досліджень є розробка способів прискорення вилучення квадратного кореня на полях Галуа, орієнтованих на апаратну реалізацію та широке розпаралелювання.

Обчислення кореня рішенням системи булевих рівнянь

Обчислення квадратного кореня на полях Галуа GF(2^m) може бути зведене до розв'язання системи лінійних бітових рівнянь. Нехай задано значення $A=a_0+a_1\cdot 2+a_2\cdot 2^2+...+a_{m-1}\cdot 2^{m-1}$, $a_0,a_1,...,a_{m-1}\in\{0,1\}$. Необхідно визначити $B=b_0+b_1\cdot 2+b_2\cdot 2^2+...+b_{m-1}\cdot 2^{m-1}$, $b_0,b_1,...,b_{m-1}\in\{0,1\}$ таке, що $(B\otimes B) rem\ P = A$ чи $B\otimes B = P\otimes D + A$, де $D=d_0+d_1\cdot 2+d_2\cdot 2^2+...+d_{m-2}\cdot 2^{m-2}$. Оскільки двійкові розряди поліноміального квадрата $B\otimes B$ числа B стоять на парних позиціях рівні нулю, а розряди з непарним номером рівні двійковим розрядам числа B, тобто $B\otimes B = b_0+b_1\cdot 2^2+b_2\cdot 2^4+...+b_{m-1}\cdot 2^{m-2}$, $b_0 = p_0\cdot d_0 + a_0$ оскільки наступний розряд $B\otimes B$ рівний нулю, то $p_0\cdot d_1 + p_1\cdot d_0 + a_1 = 0$. Аналогічним чином, може бути отримана система бітових рівнянь:

$$\begin{cases} b_0 = p_0\cdot d_0 + a_0 \\ 0 = p_0\cdot d_1 + p_1\cdot d_0 + a_1 \\ b_1 = p_0\cdot d_2 + p_1\cdot d_1 + p_2\cdot d_0 + a_2 \\ 0 = p_0\cdot d_3 + p_1\cdot d_2 + p_2\cdot d_1 + p_3\cdot d_0 + a_3 \\ \dots \\ b_{m/2-1} = p_0\cdot d_{m-2} + \dots + p_{m-2}\cdot d_0 + a_{m-2} \\ 0 = p_1\cdot d_{m-2} + p_2\cdot d_{m-3} + \dots + p_{m-1}\cdot d_0 + a_{m-1} \\ b_{m/2} = p_2\cdot d_{m-2} + p_3\cdot d_{m-3} + \dots + p_m\cdot d_0 \\ \dots \\ 0 = p_{m-1}\cdot d_{m-2} + p_m\cdot d_{m-3} \\ b_{m-1} = p_m\cdot d_{m-2} \end{cases} . \qquad (4)$$

При постійному утворюючому поліномі $P(x)$ поля Галуа GF(2^m) система (4) може бути приведена до вигляду:

$$\begin{cases} b_0 = \lambda_0(a_0, a_1, ..., a_{m-1}) \\ b_1 = \lambda_1(a_0, a_1, ..., a_{m-1}) \\ \dots \\ b_{m-1} = \lambda_{m-1}(a_0, a_1, ..., a_{m-1}) \end{cases} \qquad (5)$$

де $\lambda_0, \lambda_1, \dots, \lambda_{m-1}$ – лінійні булеві функції. Із системи (5) безпосередньо обчислюються значення бітів коду квадратного кореня на полі Галуа.

Наприклад, якщо $m = 4$, $P(x) = x^4 + x^3 + 1$, $p_0 = 1$, $p_1 = 0$, $p_2 = 0$, $p_3 = 1$, $p_4 = 1$, система (5) приймає вигляд:

$$\begin{cases} b_0 = a_0 + a_3 \\ b_1 = a_0 + a_1 \\ b_2 = a_1 + a_3 \\ b_3 = a_1 \end{cases} \qquad (6)$$

Наприклад, якщо $A = 15$ $(a_0 = 1, a_1 = 1, a_2 = 1, a_3 = 1)$, з системи (6) відповідно: $b_0 = 0$, $b_1 = 0$, $b_2 = 0$ і $b_3 = 1$, тобто шуканий корінь $B = 8$.

При програмній реалізації рішення системи (5) можуть бути заздалегідь заготовлені по u бітових масок $M_{j1}, M_{j2}, \dots, M_{js}$ для виділення розрядів коду A, які входять в функцію λ_j для кожного із значень b_j, $j\in\{0, .., m-1\}$. Відповідно, обчислення значення b_j здійснюється у вигляді логічної суми бітів парності побітових добутків фрагментів коду A на відповідні маски:

$$b_j = (a_0, a_1, ..., a_{w-1})\cdot M_{j1} + (a_w, a_{w+1}, ..., a_{2w-2})\cdot M_{j2} + .. \\ .+ (a_{m-w-1}, a_{m-w}, ..., a_{m-1})\cdot M_{js}$$

Очевидно, що загальна кількість N_L логічних операцій необхідних для обчислення значень m бітів $b_0, b_1, \ldots, b_{m-1}$ визначається як

$$N_L = s \cdot m \qquad (7)$$

Порівняння отриманого виразу з оцінкою (3) числа операцій, необхідних для обчислення квадратного кореня на GF(2^m) для відомих способів, показує, що запропонований спосіб виконання цієї операції забезпечує зменшення обчислення обчислювальної складності приблизно в $1.5 \cdot m$ разів:

$$\beta = \frac{N_T}{N_L} = \frac{1.5 \cdot (s+1) \cdot (m-1)}{s \cdot m} \approx 1.5 \cdot m$$

Враховуючи, що в алгоритмах захисту інформації значення m складає сотні і тисячі біт, виграш в обчислювальній складності і, відповідно, у часі обчислення квадратного кореня на GF(2^m) досягається використанням запропонованого способу, складає 2-3 порядки. Ще більший виграш як у часі обчислення квадратного кореня на полях Галуа, так і по складності схеми, застосування запропонованого способу забезпечується при апаратній реалізації, оскільки вирази (5) представляють собою гранично просту форму обчислення бітів кореня, кожен з яких може обчислюватися паралельно.

Висновки

Запропоновано спосіб прискореного обчислення квадратного кореня на полях Галуа GF(2^m), який базується на зведенні цієї задачі до розв'язання системи лінійних бітових рівнянь. Доведено, що запропонований спосіб має обчислювальну складність O(m), істотно меншу, ніж відомі методи - O(m^2). Запропонований спосіб орієнтований на апаратну реалізацію та паралельні обчислення бітів кореня. Проведені дослідження показали, що використання запропонованого способу забезпечує виграш у часі на 2-3 порядки.

Література

1. Menezes A. Elliptic Curve Public Key Cryptosystems. / Menezes A. - Kluwer Academic Published.-1993 - 422 p.
2. Tonelli A. Bemerkung über die Auflösung quadratischer Congruenzen / Tonelli A. // Göttinger Nachrichten.- 1891.- PP.344-346.
3. Boneh D. Identity-based encryption from the Well pairing / Boneh D., Franclin M. // SIAM Journal of Computing.- Vol.23.- 2003.- № 3.- PP. 354-368.
4. Barreto P.S.L.M. Efficient Computation of Root in Finite Fields / Barreto P.S.L.M., Voloch J.F. // Designs, Codes and cryptography.- 2006.- № 39.- pp. 275-280.
5. Cipolla M. Un metodo per la risolutione della congruenza di secondo grado / Cipolla M. // Rendiconto dell'Accademia Scienze Fisiche e Matematiche.- Napoli.-1903.- Ser.3- Vol.IX.- PP.154-163.
6. Aldeman L.M. On taking root in finite fields / Aldeman L.M., Manders K. Miller G. // Proc. 18-th IEEE Symposium on Foundations of Computer Science.-1977.-PP.175-177.
7. Марковський О.П. Спосіб прискореного обчислення коренів на полях Галуа GF(2^m) / Марковський О.П., Виноградов Ю.М., Косейкіна Г.С. // Вісник Національного технічного університету України "КПІ" Інформатика, управління та обчислювальна техніка, – Київ: ВЕК+ – 2012 – № 56. с.165-168.