О схемной и программной реализации арифметики в конечных полях характеристики 7 для вычисления спариваний*

С. Б. ГАШКОВ
Московский государственный университет им. М. В. Ломоносова
e-mail: sbgashkov@gmail.com

А. А. БОЛОТОВ
Корпорация «Эл-Эс-Ай Лоджик», США
e-mail: anatoli.bolotov@gmail.com

А. А. БУРЦЕВ
Московский физико-технический институт
e-mail: a-burtsev@yandex.ru

С. Ю. ЖЕБЕТ
Московский энергетический институт (технический университет)
e-mail: sergia@list.ru

А. Б. ФРОЛОВ
Московский энергетический институт (технический университет)
e-mail: abfrolov@gmail.com

УДК 515.1

Ключевые слова: криптография, эллиптические кривые, сложность вычислений.

Аннотация

Изучаются схемные и программные методы умножения многочленов над полями характеристики 7 применительно к реализации криптографических протоколов на гиперэллиптических кривых третьего рода, основанных на спариваниях. Рассматриваются схемные и программные методы реализации арифметики в полях $GF(7)$, $GF(7^2)$, $GF(7^n)$, $GF(7^{7n})$ и $GF(7^{14n})$ и оценивается сложность соответствующих схем и программ.

Abstract

S. B. Gashkov, A. A. Bolotov, A. A. Burtsev, S. Yu. Zhebet, A. B. Frolov, On hardware and software implementation of arithmetic in finite fields of characteristic 7 for calculation of pairings, Fundamentalnaya i prikladnaya matematika, vol. 15 (2009), no. 3, pp. 75—111.

We study scheme (hardware) and program (software) methods of multiplication of polynomials over fields of characteristic 7 in order to apply them to parings based

*Исследование выполнено при поддержке РФФИ, проект 08-01-00632-а, аналитической ведомственной целевой программы «Развитие научного потенциала высшей школы», проект 2.1.1/1662, а также программы поддержки ведущих научных школ РФ, проект НШ-4470.2008.1.

Фундаментальная и прикладная математика, 2009, том 15, № 3, с. 75—111.

cryptographic protocols on hyperelliptic curves of genus three. We consider hardware and software implementations of arithmetic in $GF(7)$, $GF(7^2)$, $GF(7^n)$, $GF(7^{7n})$, and $GF(7^{14n})$ and estimate the complexity of corresponding schemes and programs.

Посвящается 70-летию академика В. Б. Кудрявцева

1. Введение

В криптографии обычно применяются эллиптические кривые или над простыми полями, или над полями характеристики 2. Последние наиболее удобны для реализации в виде электронных схем (более подробно об этом см., например, [3, 4, 13, 14]). С появлением идеи использования в криптографии так называемых билинейных спариваний (введённых в работах Вейля, Тейта и Лихтенбаума) для конструирования новых криптоалгоритмов стали применяться эллиптические и гиперэллиптические кривые над полями характеристики 3 (см., например, [10,11,26]). Как следствие, появился интерес к реализации арифметики в этих и других полях нечётной характеристики (см., например, [9,19,22,25]). В [17, 18] алгоритм из [10] для быстрого вычисления спаривания в эллиптических кривых $Y^2 = X^3 - X \pm 1$ над полем $GF(3^n)$ был усовершенствован и перенесён на гиперэллиптические кривые C_b: $Y^2 = X^p - X + b$, $b = \pm 1$, определённые над полем $k = GF(p^n)$, где n и $2p$ взаимно просты и $p \equiv 3 \pmod 4$. Эти кривые рассматриваются также над расширениями поля k, а именно над $F = GF(p^{pn})$ и $K = GF(p^{2pn})$. Рассматриваемые кривые являются обобщениями эллиптических кривых $E_{3,b}$: $Y^2 = X^3 - X + b$. В первоначальном варианте алгоритма [17, 18], кроме обычных арифметических операций, использовалась операция извлечения кубических корней в поле $GF(3^n)$. Вариант этого алгоритма без использования этой операции был предложен в [23] (там также был дан вариант этого алгоритма для суперсингулярных кривых над полем $GF(2^n)$). В [11] введено так называемое η-спаривание и показано, что для его вычисления можно уменьшить вдвое длину выполняемого цикла в алгоритме Дуурсма—Ли. В [12] показано, как в вычислении η-спаривания можно обойтись без извлечения кубических корней, а в [27] — как можно ускорить алгоритм финального потенцирования при вычислении η-спаривания. Во всех этих алгоритмах используется арифметика в поле $GF(3^{6n})$.

Опишем кратко обобщение алгоритма [23] на гиперэллиптические кривые C_b: $Y^2 = X^p - X + b$, $b = \pm 1$ (более подробное изложение см. в [4]).

Пусть $\rho \in GF(p^p) \subset F = GF(p^{pn})$ — корень многочлена $X^p - X + 2b$ над $GF(p)$, и пусть $\sigma \in GF(p^2) \subset K = GF(p^{2pn})$ — корень многочлена $X^2 + 1$. Многочлен $X^p - X + 2b$ неприводим над полем $GF(p)$, поэтому все его корни лежат в поле $GF(p^p)$. Одновременно многочлен $X^p - X + 2b$ над $GF(p^2)$ неприводим над полем $GF(p^2)$.

Пусть даны точки

$$P = (\alpha, \beta) \in C_b(k), \quad Q = (\rho - x, \sigma y) \in F \times K, \quad (x, y) \in C_b(k),$$

где

$$\sigma^2 = -1, \quad \rho^p - \rho + 2b = 0, \quad k = GF(p^n), \quad F = GF(p^{pn}), \quad K = GF(p^{2pn}).$$

Вначале выполняется инициализация

$$f = 1; \quad x = x^p; \quad y = y^p; \quad d = -(2n-1)b.$$

В цикле по i от 1 до n выполняем подстановки

$$\alpha = (\alpha^p)^p; \quad \beta = (\beta^p)^p; \quad g = (\beta y\sigma - (\alpha + x + d - \rho)^{(p+1)/2});$$
$$f = f^p g; \quad y = -y; d = d + 2b.$$

Полученное в конце цикла f возводим в степень $p^{pn} - 1$.

Для реализации этого алгоритма (ниже мы называем его алгоритмом Дуурсма, Ли и Квона или, сокращённо, алгоритмом ДЛК) необходимо реализовать арифметику в поле $GF(p^{2np})$ и его подполях $GF(p^n)$ и $GF(p^{2n})$, последняя используется для реализации арифметики в поле $GF(p^{2np})$. В случае $p = 3$ это сделано в [19, 22, 25, 26]. В [4] кратко описан метод построения арифметики в этих полях при любом простом $p \equiv 3 \pmod 4$. Из этого описания следует, что с ростом p сложность схемной реализации умножения в поле $GF(p^{2pn})$ уменьшается (при условии, что n изменятся так, что порядок поля существенно не изменится). Под схемной реализацией понимается реализация операций булевыми (не автоматными) схемами, а под сложностью — число базисных элементов, составляющих схему (базис состоит из двуместных булевых функций $\&$, $\vee$, $\oplus$ и их отрицаний). Понятие схемной сложности по существу совпадает с известным понятием битовой сложности.

В настоящей работе будет показано, что с ростом p битовая сложность алгоритма ДЛК тоже уменьшается (при сохранении того же уровня надёжности криптопротоколов, использующих этот алгоритм). Поэтому представляется более эффективным использовать этот алгоритм при $p = 7$. В [24] развиваются идеи из [17, 18] в случае $p = 7$. Предлагаемая ниже реализация арифметики в этом случае может найти применение и в алгоритмах из [24].

Работа состоит из данного введения, семи разделов и заключения. В разделе 2 приведены схемные реализации операций сложения и умножения в $GF(7)$ и $GF(7^2)$. В разделе 3 оценивается сложность умножения в полях $GF(7^{2n})$ над подполем $GF(7^2)$. Раздел 4 посвящён схемной реализации арифметики в поле $GF(p^{2np})$ над подполем $GF(p^{2n})$. В разделе 5 осуществлены вывод битовой сложности алгоритма ДЛК относительно $\mathrm{A}(p)$ — сложности аддитивных операций в $GF(p)$, $\mathrm{M}(k)$ и $\mathrm{M}(K)$ — сложностей умножения в полях k и K. В разделе 6 произведено сравнение битовой сложности алгоритма ДЛК при $p = 3$ и $p = 7$. Раздел 7 посвящён описанию и сравнительному анализу программной реализации схем арифметики в полях $GF(7^n)$ и $GF(7^{2n})$. В разделе 8 получены сравнительные оценки реализации умножения многочленов над $GF(p^n)$ и над $GF(p^2)$, $p < 2^8$, различными методами.

2. Схемы для арифметики по модулю 7

В [22] показано, что умножение в $GF(3)$ выполняется булевой схемой сложности 6 и глубины 2, сложение в $GF(3)$ выполняется схемой сложности 7 и глубины 3.

Лемма 1. *Умножение в $GF(7)$ выполняется схемой сложности* 25 *и глубины* 5.

Доказательство. Заметим, что умножение числа $(x_2x_1x_0)_2$ по модулю 7 на числа $4 = (100)_2$, $2 = (010)_2$ равносильно циклическим сдвигам битов x_i, умножение на $1 = (001)_2$ — это тождественное преобразование, умножение на числа $3 = (011)_2 = -4$, $5 = (101)_2 = -2$, $6 = (110)_2 = -1$ делается точно так же, но у результата меняется знак на противоположный, что равносильно прибавлению по модулю два ко всем битам результата знакового бита σ. Очевидно, этот бит можно вычислить по формуле $m(y_2, y_1, y_0) = (y_0 + y_1)(y_0 + y_2) + y_0$. Умножение на числа $(000)_2$ и $(111)_2$ даёт в результате $(000)_2$.

Рассмотрим линейный оператор

$$u_1 = y_0 \oplus y_1 \oplus 1, \quad u_0 = y_0 \oplus y_2 \oplus 1.$$

Очевидно, он принимает значение 11 на противоположных наборах 000, 111, значение 10 на противоположных наборах 011, 100, значение 01 на противоположных наборах 010, 101, значение 00 на противоположных наборах 001, 110.

Рассмотрим оператор

$$d_2 = \neg u_1 \,\&\, \neg u_0, \quad d_1 = u_1 \,\&\, \neg u_0, \quad d_0 = \neg u_1 \,\&\, u_0.$$

На наборах 000, 111 он принимает значение 000, на наборах 100, 011 — значение 010, на наборах 010, 101 — значение 001, на наборах 001, 110 — - значение 100.

Рассмотрим оператор

$$a_2 = d_2x_2 \vee d_1x_0 \vee d_0x_1, \quad a_1 = d_2x_1 \vee d_1x_2 \vee d_0x_0, \quad a_0 = d_2x_0 \vee d_1x_1 \vee d_0x_2.$$

На наборах $000x_2x_1x_0$, $111x_2x_1x_0$ он принимает значение 000, на наборах $001x_2x_1x_0$, $110x_2x_1x_0$ — значение $x_2x_1x_0$, на наборах $010x_2x_1x_0$, $101x_2x_1x_0$ — значение $x_1x_0x_2$, $(x_1x_0x_2)_2 = 2(x_2x_1x_0)_2 \bmod 7$, на наборах $100x_2x_1x_0$, $011x_2x_1x_0$ — значение $x_0x_2x_1$, $(x_0x_2x_1)_2 = 4(x_2x_1x_0)_2 \bmod 7$. Тогда умножение чисел $(x_2x_1x_0)_2$, $(y_2y_1y_0)_2$ по модулю 7 реализуется оператором

$$z_2 = a_2 \oplus m, \quad z_1 = a_1 \oplus m, \quad z_0 = a_0 \oplus m.$$

Действительно, если $(y_2y_1y_0)_2 = 0$, то $m = 0$ и он принимает значение $(000)_2 = (000)_2 \cdot (x_2x_1x_0)_2 \bmod 7$; если $(y_2y_1y_0)_2 = 7$, то $m = 1$ и он принимает значение $(111)_2 = 7 = 0 \bmod 7 = (111)_2 \cdot (x_2x_1x_0)_2 \bmod 7$; если $(y_2y_1y_0)_2 = a = 1, 2, 4$, то $m = 0$ и он принимает значение $a(x_2x_1x_0)_2 \bmod 7$; если $(y_2y_1y_0)_2 = a = 3, 5, 6$, то $-a \bmod 7 = b = 4, 2, 1$, $m = 1$ и он принимает значение $-b(x_2x_1x_0)_2 \bmod 7 = a(x_2x_1x_0)_2 \bmod 7$.

Сложность схемы для оператора m, d_2, d_1, d_0 равна 7, глубина выходов d_i равна 2, глубина выхода m равна 3. Сложность схемы для оператора a_2, a_1, a_0 равна $15 + 7 = 22$, глубина равна 5. Сложность схемы для оператора z_2, z_1, z_0 равна $22 + 3 = 25$, глубина равна 6. Если в схеме для a_2, a_1, a_0 заменить $\vee$ на $\oplus$, то оператор z_2, z_1, z_0 можно реализовать формулами

$$z_2 = (d_2x_2 \oplus d_1x_0) \oplus (d_0x_1 \oplus m), \quad z_1 = (d_2x_1 \oplus d_1x_2) \oplus (d_0x_0 \oplus m),$$
$$z_0 = (d_2x_0 \oplus d_1x_1) \oplus (d_0x_2 \oplus m)$$

с той же сложностью 25 и глубиной 5. □

Отождествим набор $(111)_2$ с набором $(000)_2$, так как $7 = 0 \bmod 7$.

Лемма 2. *Сложение в $GF(7)$ выполняется схемой сложности* 17 *и глубины* 8. *Существует также схема для сложения сложности* 18 *и глубины* 7 *и схема для сложения сложности* 20 *и глубины* 6.

Доказательство. Действительно, складываем два трёхзначных двоичных числа при помощи стандартной схемы сложения таких чисел со сложностью 12 и глубиной 7, получаем сумму $(\sigma, \varepsilon_2, \varepsilon_1, \varepsilon_0)_2$. Приведение по модулю

$$(\sigma, \varepsilon_2, \varepsilon_1, \varepsilon_0)_2 = 2^3\sigma + (\varepsilon_2, \varepsilon_1, \varepsilon_0)_2 = (\varepsilon_2, \varepsilon_1, \varepsilon_0)_2 + \sigma = (z_2z_1z_0)_2 \bmod 7$$

имеет сложность 5 и глубину 3, так как при прибавлении однобитового числа

$$z_1 = x_1 + y_1, \quad z_2 = x_2 + (x_1y_1), \quad z_3 = x_3 + x_2(x_1y_1) \bmod 2.$$

Таким образом, схема сложения в $GF(7)$ имеет сложность 17 и глубину 8.

Если подсхему прибавления однобитового числа построить по формулам

$$z_1 = x_1 + y_1, \quad z_2 = x_2 + (x_1y_1), \quad z_3 = x_3 + (x_2x_1)y_1 \bmod 2,$$

немного изменив порядок действий, то сложность подсхемы будет 6, но глубина основной схемы увеличится лишь на 2, а не на 3, так как вычисление x_2x_1 можно произвести заранее. Эта схема сложения в $GF(7)$ имеет сложность 18 и глубину 7.

Если обычную схему сложения двух трёхзначных чисел переписать в виде

$$z_1 = x_1 + y_1, \quad z_2 = (x_2 + y_2) + (x_1y_1),$$
$$z_3 = \big((x_3 + y_3) + x_2y_2\big) + \big((x_2 + y_2)(x_1y_1)\big),$$
$$z_4 = \big((x_3y_3) + (x_3 + y_3)(x_2y_2)\big) + (x_3 + y_3)\big((x_2 + y_2)(x_1y_1)\big),$$

то полученная схема будет иметь сложность 14 и глубину 4. Таким образом, схема сложения в $GF(7)$ будет иметь сложность 20 и глубину 6. □

Лемма 3. *Сложение в $GF(7)$ может быть выполнено схемой сложности* 21 *и глубины* 4.

Доказательство. Действительно, можно непосредственно проверить, что указанная выше схема сложения по модулю 7 реализует на первом выходе функцию

$$z_0 = \big((x_0 + y_0 + x_2y_2) + x_1y_1(x_2 + y_2)\big) + (x_2 + y_2)(x_1 + y_1)x_0y_0.$$

Так как при умножении каждого из слагаемых $(x_2x_1x_0)$, $(y_2y_1y_0)$ на 2 по модулю 7 происходит циклический сдвиг $x_0 \to x_1 \to x_2 \to x_0$, то такой же сдвиг происходит и с суммой по модулю семь, поэтому формулы для z_1, z_2 получаются из формулы для z_0 циклическими сдвигами переменных. Очевидно, сложность схемы равна 21, так как со сложностью 6 вычисляются $x_i \oplus y_i$, x_iy_i, после чего со сложностью 3 вычисляются $x_1y_1(x_2 \oplus y_2)$, $x_2y_2(x_0 \oplus y_0)$, $x_0y_0(x_1 \oplus y_1)$, каждая формула вычисляется со сложностью 4. Глубина равна 4. □

Схемы для вычитания получаются путём навешивания отрицаний на биты, составляющие вычитаемое число. Поэтому сложность и глубина схемы для вычитания такие же, как у схемы сложения.

Лемма 4. *Сложность умножения на любую константу из* $GF(7)$ *в любом поле* $GF(7^n)$ *равна* 0.

Доказательство. Умножение на 2 произвольного элемента в $GF(7)$ сводится к циклической перестановке двоичной записи этого элемента. Действительно умножение на 2 представляется сдвигом двоичной записи влево, приписыванием нуля справа и приведением по модулю 7. Сложность смены знака также равна 0 в этих полях, так как вместо сложения можно делать в соответствующих местах вычитание и наоборот, а сложение и вычитание имеют одинаковую схемную сложность в $GF(7)$. Поэтому сложность умножения на любую константу из $GF(7)$ равна 0 в любом поле $GF(7^n)$. □

Для умножения в поле $GF(7^2)$ можно использовать две схемы.

Стандартная схема, основанная на формуле

$$(a + b\sigma)(c + d\sigma) = (ac - bd) + \sigma(ad + bc),$$

имеет сложность $4\,\mathrm{M}(7) + 2\,\mathrm{A}(7) = 134$ и глубину $\mathrm{D}\big(\mathrm{M}(7)\big) + \mathrm{D}\big(\mathrm{A}(7)\big) = 12$. Существуют схемы сложности 142 и глубины 9 и сложности 136 и глубины 11.

Схема в нормальном базисе, основанная на формуле

$$\big((a + b)(c + d) - 2bd\big)\gamma + \big((a + b)(c + d) - 2ac\big)\gamma^p$$

имеет сложность $3\,\mathrm{M}(7) + 4\,\mathrm{A}(7) = 146$ и глубину

$$\mathrm{D}\big(\mathrm{M}(7)\big) + 2\,\mathrm{D}\big(\mathrm{A}(7)\big).$$

Эта схема хуже. Но для больших p она имеет меньшую сложность, чем стандартная.

3. Умножение в поле $GF(7^{2n})$ над полем $GF(7^2)$

Рассмотрим случай $n \leqslant 31$ и покажем, что даже для таких малых степеней использование дискретного преобразования Фурье (ДПФ) имеет преимущество перед школьным методом и методом Карацубы.

3.1. Умножение многочленов степени 23 и 24

Для вычисления ДПФ порядка 48 используем стандартную конструкцию, описанную, например, в [8].

Пусть $\varepsilon = \omega_3 \neq 1$, $\varepsilon^3 = 1$. В поле $GF(7)$ $\varepsilon = 2$, $\varepsilon^2 = 4$.

Тогда F_3 определяется равенствами

$$y_0 = x_0 + x_1 + x_2, \quad y_1 = x_0 + \varepsilon x_1 + \varepsilon^2 x_2, \quad y_2 = x_0 + \varepsilon^2 x_1 + \varepsilon^4 x_2$$

и вычисляется по формулам

$$z_1 = x_1 + x_2, \quad z_2 = x_1 - x_2, \quad z_3 = \varepsilon z_2, \quad z_4 = z_3 - x_2, \quad z_5 = z_1 + z_4,$$
$$y_0 = x_0 + z_1, \quad y_1 = x_0 + z_4, \quad y_2 = x_0 - z_5.$$

Рассмотрим следующую схему для F_3, удобную для применения не к числам x_i, а к произвольным векторам:

$$z_1 = x_1 + x_2, \quad z_2 = x_1 - x_2, \quad y_0 = x_0 + z_1,$$
$$z_3 = \left(\frac{\varepsilon + \varepsilon^2}{2} - 1\right) z_1, \quad z_4 = \frac{\varepsilon - \varepsilon^2}{2} z_2,$$
$$z_5 = y_0 + z_3, \quad y_1 = z_5 + z_4, \quad y_2 = z_5 - z_4.$$

В поле $GF(7)$ $\frac{\varepsilon - \varepsilon^2}{2} = -1$, $\frac{\varepsilon + \varepsilon^2}{2} - 1 = 2$, поэтому в этой схеме $z_3 = 2z_1$, $z_4 = -z_2$. Она удобна для векторизации, т. е. для применения к векторам x_i, которые сами являются результатами применения одного линейного преобразования $x_i = F(X_i)$. Тогда для вычисления F_3 в указанной схеме можно использовать новые операции

$$Z_1 = X_1 + X_2, \quad Z_2 = X_1 - X_2, \quad y_0 = x_0 + z_1 = F(X_0 + Z_1),$$
$$z_3 = 2z_1 = 2F(Z_1), \quad z_4 = -z_2 = -F(Z_2),$$

оставив остальные без изменения. Если первоначальный вариант схемы при применении к векторам $x_i = F(X_i)$ требует трёхкратного вычисления преобразования F, 6 векторных сложений и 2 скалярных умножения, то указанный вариант требует те же 6 векторных сложений, но не требует скалярных умножений, зато вместо трёхкратного вычисления F вычисляется один раз в чистом виде и два раза в виде, умноженном на скаляры 2, -1.

Для вычисления ДПФ порядка 16 достаточно 17 скалярных умножений и 64 сложений. В поле $GF(7^2)$ сложность умножения на w^4 равна 0, поэтому пропадает $1 + 2 + 4$ скалярных умножений и остаётся только 10.

Для вычисления ДПФ порядка 48 в поле $GF(7^2)$ применяем теорему Гуда—Томаса и представляем $F_{48}(X)$ в виде

$$F_{48} = F_3\big(F_{16}(X_1), F_{16}(X_2), F_{16}(X_3)\big),$$

где X_i — векторы, составленные из компонент вектора X по следующему правилу:

$$X_1 = (x_0, x_3, \ldots, x_{45}), \quad X_2 = (x_{16}, x_{19}, \ldots, x_{46}, x_1, x_4, \ldots, x_{12}),$$
$$X_3 = (x_{32}, x_{35}, \ldots, x_{47}, x_2, x_5, \ldots, x_{29}),$$

а F_3 — это векторизация ДПФ третьего порядка, рассмотренная выше. Так как для его вычисления используются 6 векторных сложений (т. е. 96 обычных) и вычисляются преобразования F_{16}, $2F_{16}$, $-F_{16}$, то дополнительно нужно 192 сложения и 30 скалярных умножений. Окончательно нужно 288 сложений и 30 скалярных умножений.

Для умножения многочленов степени 23 достаточно сделать 48 умножений и три преобразования Фурье F_{48}. Значит, кроме умножений, достаточно не более 864 сложений и 90 скалярных умножений. На самом деле в первых двух ДПФ экономится по 48 сложений в каждом. Действительно, в векторе коэффициентов многочлена только первые 24 коэффициента могут быть отличны от нуля, поэтому в векторе X_1 последние 8 коэффициентов нулевые, в векторе X_2 имеется 8 подряд идущих (в циклическом смысле) нулевых коэффициентов и в векторе X_3 тоже. Деление многочленов степени 15 с такими наборами коэффициентов на двучлены $x^8 \pm 1$ с остатком выполняется со сложностью 0 (для нахождения остатка нужен только сдвиг коэффициентов, возможно со сменой знака у некоторых из них). Глубина F_{48} равна $8\,\mathrm{D}(\mathrm{A}(7))$. Поэтому сложность циклической свёртки многочленов степени 23 равна

$$858\,\mathrm{A}\big(GF(7^2)\big) + 48\,\mathrm{M}\big(GF(7^2)\big) = 35\,604.$$

Глубина равна

$$16\,\mathrm{D}\big(\mathrm{A}(7)\big) + D\Big(\mathrm{M}\big(GF(7^2)\big)\Big).$$

Можно проверить, что сложность и глубина циклической свёртки многочленов 24-й степени оценивается так же. Но для умножения многочленов 24-й степени, кроме вычисления этой свёртки, нужно ещё одно умножение и одно вычитание в $GF(7^2)$.

3.2. Умножение многочленов степени 27 и 31

Для умножения многочленов степени 27 достаточно вычислить их циклическую свёртку по модулю $X^{48} - 1$, которая равна

$$h_0 + h_{48} + (h_1 + h_{49})X + (h_2 + h_{50})X^2 + (h_3 + h_{51})X^3 + (h_4 + h_{52})X^4 + \\ + (h_5 + h_{53})x^5 + (h_6 + h_{54})X^6 + h_7X^7 + \ldots + h_{47}X^{47},$$

вычислить

$$h_0 = f_0g_0, \quad h_1 = f_1g_0 + f_0g_1, \quad h_2 = f_2g_0 + f_0g_2 + f_1g_1,$$
$$h_3 = f_3g_0 + f_0g_3 + f_2g_1 + f_1g_2,$$
$$h_{52} = f_{27}g_{25} + f_{25}g_{27} + f_{26}g_{26}, \quad h_{53} = f_{27}g_{26} + f_{26}g_{27}, \quad h_{54} = f_{27}g_{27}$$

со сложностью $14\,\mathrm{M}\big(GF(7^2)\big) + 15\,\mathrm{A}\big(GF(7^2)\big)$, а потом сделать ещё 7 вычитаний со сложностью $7\,\mathrm{A}\big(GF(7^2)\big)$.

Для того чтобы вычислить циклическую свёртку, достаточно сделать 48 умножений и 3 ДПФ F_{48}. В первых двух ДПФ экономится по 40 сложений

в каждом. Действительно, в векторе коэффициентов многочлена только первые 28 коэффициента могут быть отличны от нуля, поэтому в векторе X_1 последние 6 коэффициентов нулевые, в векторе X_2 есть 7 идущих подряд (в циклическом смысле) нулевых коэффициентов и в векторе X_3 тоже. Деление многочлена 15-й степени с вектором коэффициентов X_1 на двучлены $X^8 \pm 1$ с остатком выполняется с помощью 4 аддитивных операций, а в случае векторов X_2, X_3 — с помощью 2 аддитивных операций.

Сложность умножения многочленов степени 27 равна

$$874\,\mathrm{A}\big(GF(7^2)\big) + 62\,\mathrm{M}\big(GF(7^2)\big) = 38\,024.$$

Глубина равна

$$17\,\mathrm{D}\big(\mathrm{A}(7)\big) + D\Big(\mathrm{M}\big(GF(7^2)\big)\Big).$$

Аналогично для умножения многочленов степени 31 достаточно вычислить их циклическую свёртку по модулю $x^{48} - 1$, которая равна

$$h_0 + h_{48} + (h_1 + h_{49})X + \ldots + (h_{14} + h_{62})X^{14} + h_{15}X^{15} + \ldots + h_{47}X^{47},$$

вычислить

$$h_0 = f_0 g_0, \quad h_1 = f_1 g_0 + f_0 g_1, \ldots, \quad h_7 = f_7 g_0 + f_6 g_1 + \ldots + f_0 g_7,$$
$$h_{62} = f_{31} g_{31}, \quad h_{61} = f_{31} g_{30} + f_{30} g_{31}, \ldots, \quad h_{56} = f_{31} g_{25} + \ldots + f_{25} g_{31}.$$

Для того чтобы вычислить циклическую свёртку порядка 48 достаточно сделать 48 умножений и 3 ДПФ F_{48}. В первых двух ДПФ экономится по 32 сложений в каждом. Действительно, в векторе коэффициентов многочлена только первые 32 коэффициента могут быть отличны от нуля, поэтому в векторе X_1 последние 5 коэффициентов нулевые, в векторе X_2 есть 5 идущих подряд (в циклическом смысле) нулевых коэффициентов, а в векторе X_3 первые 6 коэффициентов нулевые. Деление многочленов 15-й степени с векторами коэффициентов X_1, X_2 на двучлены $x^8 \pm 1$ с остатком выполняется с помощью 6 аддитивных операций, а в случае вектора X_3 — с помощью 4 аддитивных операций. Поэтому свёртка выполняется со сложностью $890\,\mathrm{A}\big(GF(7^2)\big) + 48\,\mathrm{M}\big(GF(7^2)\big)$.

Вычисление свёртки

$$h_0 = f_0 g_0, \quad h_1 = f_1 g_0 + f_0 g_1, \ldots, \quad h_7 = f_7 g_0 + f_6 g_1 + \ldots + f_0 g_7$$

равносильно вычислению произведения многочленов степени 7 по модулю x^8.

Многочлен седьмой степени можно представить в виде

$$(a_0 + a_1 X + a_2 X^2 + a_3 x^3) + X^4(a_4 + a_5 X + a_6 X^2 + a_7 X^3) = f_0 + f_1 Y, \quad Y = X^4,$$

где

$$f_0 = a_0 + a_1 X + a_2 X^2 + a_3 X^3, \quad f_1 = a_4 + a_5 X + a_6 X^2 + a_7 X^3.$$

Перемножение двух многочленов f и g по модулю X^8 согласно тождеству

$$(f_0 + f_1 Y)(g_0 + g_1 Y) \bmod Y^2 = (f_0 g_1 + f_1 g_0)Y + f_0 g_0$$

сводится к умножению двух четырёхчленов, к двукратному умножению двух четырёхчленов по модулю X^4 и 7 сложениям. Умножение четырёхчленов по модулю X^4 аналогично сводится к умножению двух двучленов, к двукратному умножению двух двучленов по модулю X^2 и 3 сложениям. Умножение двучленов методом Карацубы имеет сложность $3\,\mathrm{M}\big(GF(7^2)\big)+4\,\mathrm{A}\big(GF(7^2)\big)$. Поэтому умножение четырёхчленов по модулю X^4 имеет сложность $9\,\mathrm{M}\big(GF(7^2)\big)+9\,\mathrm{A}\big(GF(7^2)\big)$.

Оценим сложность умножения четырёхчленов. Их произведением является многочлен $a_0 + a_1X + \ldots + a_6X^6$. Его коэффициенты

$$a_0 = f_0g_0, \quad a_6 = f_3g_3, \quad a_1 = (f_0g_1 + f_1g_0)$$

находятся со сложностью $4\,\mathrm{M}\big(GF(7^2)\big) + \mathrm{A}\big(GF(7^2)\big)$, где f_i и g_i — соответствующие коэффициенты исходных сомножителей. После этого для вычисления $a_0 + a_1X + \ldots + a_6X^6$ достаточно найти циклическую свёртку четырёхчленов

$$a_0+a_1X+\ldots+a_6X^6 = (a_0+a_4)+(a_1+a_5)X+(a_2+a_6)X^2+a_3X^3 \bmod (X^4-1)$$

и выполнить 3 вычитания. Для вычисления свёртки достаточно вычислить три раза ДПФ четвёртого порядка по схеме

$$\begin{array}{lll} h \bmod (X^4-1), & h \bmod (X^2-1), & h \bmod (X^2+1), \\ h \bmod (X^2-1), & h \bmod (X+1), & h \bmod (X-1), \\ h \bmod (X^2+1), & h \bmod (X-\omega^4), & h \bmod (X+\omega^4) \end{array}$$

со сложностью $24\,\mathrm{A}\big(GF(7^2)\big)$ и выполнить ещё 4 умножения в $GF(7^2)$. Суммируя оценки, находим, что умножение двух четырёхчленов выполняется со сложностью $8\,\mathrm{M}\big(GF(7^2)\big)+28\,\mathrm{A}\big(GF(7^2)\big)$. Поэтому умножение многочленов степени 7 по модулю X^8 выполняется со сложностью

$$8\,\mathrm{M}\big(GF(7^2)\big)+28\,\mathrm{A}\big(GF(7^2)\big)+2\Big(9\,\mathrm{M}\big(GF(7^2)\big)+9\,\mathrm{A}\big(GF(7^2)\big)\Big)+7\,\mathrm{A}\big(GF(7^2)\big) = \\ = 26\,\mathrm{M}\big(GF(7^2)\big) + 53\,\mathrm{A}\big(GF(7^2)\big).$$

Вычисление свёртки $h_{62}, \ldots, h_{56}$ равносильно вычислению произведения многочленов степени 6 по модулю X^7. Многочлен 6-й степени можно представить в виде

$$(a_0 + a_1X + a_2X^2 + a_3X^3) + X^4(a_4 + a_5X + a_6X^2) = f_0 + f_1Y, \quad Y = X^4.$$

Перемножение двух многочленов f и g по модулю X^7 согласно тождеству

$$(f_0 + f_1Y)(g_0 + g_1Y) \bmod Y^2 = (f_0g_1 + f_1g_0)Y + f_0g_0$$

сводится к умножению двух четырёхчленов, к двукратному умножению двух трёхчленов по модулю X^3 и 6 сложениям. Умножение трёхчленов по модулю X^3 аналогично сводится к умножению двух двучленов, к двукратному умножению одночленов и 2 сложениям. Поэтому умножение двух трёхчленов по модулю X^3 можно сделать со сложностью $5\,\mathrm{M}\big(GF(7^2)\big) + 6\,\mathrm{A}\big(GF(7^2)\big)$. Сле-

довательно, умножение многочленов степени 6 по модулю X^7 выполняется со сложностью

$$8\,\mathrm{M}\big(GF(7^2)\big) + 28\,\mathrm{A}\big(GF(7^2)\big) + 10\,\mathrm{M}\big(GF(7^2)\big) + 12\,\mathrm{A}\big(GF(7^2)\big) + 6\,\mathrm{A}\big(GF(7^2)\big) = \\ = 18\,\mathrm{M}\big(GF(7^2)\big) + 46\,\mathrm{A}\big(GF(7^2)\big).$$

Значит, сложность вычисления

$$h_0 = f_0 g_0, \quad h_1 = f_1 g_0 + f_0 g_1, \ldots, \quad h_7 = f_7 g_0 + f_6 g_1 + \ldots + f_0 g_7,$$
$$h_{62} = f_{31} g_{31}, \quad h_{61} = f_{31} g_{30} + f_{30} g_{31}, \ldots, \quad h_{56} = f_{31} g_{25} + \ldots + f_{25} g_{31}$$

равна $44\,\mathrm{M}\big(GF(7^2)\big) + 99\,\mathrm{A}\big(GF(7^2)\big)$.

Поэтому сложность умножения многочленов 31-й степени равна

$$890\,\mathrm{A}\big(GF(7^2)\big) + 48\,\mathrm{M}\big(GF(7^2)\big) + 15\,\mathrm{A}\big(GF(7^2)\big) + 44\,\mathrm{M}\big(GF(7^2)\big) + 99\,\mathrm{A}\big(GF(7^2)\big) = \\ = 1004\,\mathrm{A}\big(GF(7^2)\big) + 92\,\mathrm{M}\big(GF(7^2)\big) = 46\,464.$$

Глубина равна

$$18\,\mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\big(\mathrm{M}(7)\big) = 131.$$

Если взять схемы сложности и глубины

$$\mathrm{D}\big(\mathrm{A}(7)\big) = 4, \quad \mathrm{A}(7) = 21, \quad \mathrm{D}\big(\mathrm{M}(7)\big) = 5, \quad \mathrm{M}\big(GF(7^2)\big) = 142,$$

то получим оценки сложности и глубины схемы умножения многочленов 31-й степени:

$$\mathrm{M}(GF(7^2), 31) = 55232, \quad \mathrm{D}\big(\mathrm{M}(GF(7^2), 31)\big) = 77.$$

Для того чтобы превратить схему умножения многочленов 30-й степени в схему умножения в поле $GF(7^{62})$, надо её присоединить к схеме, вычисляющей остаток от деления многочлена 60-степени на данный неприводимый над полем $GF(7^2)$ многочлен степени 31. Если этот многочлен является трёхчленом, то можно его выбрать так, что средний член имеет степень не больше 15 (взяв в случае необходимости вместо $f(X) = X^n + aX^k + b$ многочлен $b^{-1} f^*(X) = b^{-1} X^n f(1/X) = b^{-1} + ab^{-1} X^{n-k} + X^n$). Тогда деление в столбик заканчивается на втором шаге, и полученная таким образом схема имеет сложность не больше $4 \cdot 31\,\mathrm{A}\big(GF(7^2)\big)$ и глубину не больше $4D\Big(\mathrm{A}\big(GF(7^2)\big)\Big)$. В поле $GF(7^{2\cdot 27})$ в качестве неприводимого многочлена можно выбрать двучлен. В этом случае сложность и глубина схемы приведения по модулю уменьшаются вчетверо. В любом случае сложность и глубина схемы умножения в поле $GF(7^{62})$ по сравнению со сложностью и глубиной схемы умножения многочленов 30-й степени возрастает незначительно.

3.3. Умножение многочленов степени 49

Для умножения многочленов степени 49 представляем их в виде

$$a_0 + a_1 Y + \ldots + a_{24} Y^{24}, \quad Y = X^2, \quad \deg a_i = 1,$$

и вычисляем циклическую свёртку 24-го порядка. Для её выполнения требуется $2 \cdot 240 = 480$ сложений двучленов, 20 умножений двучленов на скаляр из поля $GF(7^2)$, 48 умножений двучленов, 288 сложений трёхчленов и 10 умножений двучленов на скаляр из поля $GF(7^2)$. Умножение двучленов требует 3 умножения и 4 сложения. Для вычисления произведения скалярных многочленов степени 49 по известной свёртке 24-го порядка нужно ещё одно умножение двучленов и одно вычитание трёхчленов. Потом для получения двучленных коэффициентов нужно ещё 48 сложений в поле $GF(7^2)$. Окончательно требуется $3 \cdot 49 = 147$ умножений, $2 \cdot 480 + 3 \cdot 289 + 4 \cdot 49 + 48 = 2071$ сложений и $40 + 30 = 70$ скалярных умножений в поле $GF(7^2)$. Поэтому схемная сложность равна 90 112. Глубина равна $20\,\mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\big(\mathrm{M}(7)\big)$.

3.4. Сравнение с методом Карацубы в случае выбора полей характеристики 3

В случае выбора полей характеристики 3 при реализации алгоритма ДЛК для достижения того же уровня надёжности надо выбрать поле $GF(3^{2n})$ с $n \geqslant 127$. Использование метода Карацубы для реализации умножения многочленов 127-й степени над полем $GF(3^2)$ приводит к схеме сложности не меньше $38 \cdot 3^7 + 14(6 \cdot 3^7 - 2^{10} + 2) = 252\,508$ и глубины не меньше $3 \cdot 7 \cdot 3 + 5 = 61$. Школьный алгоритм имеет глубину $7 \cdot 3 + 5 = 26$ и сложность $38 \cdot 2^{14} + 14 \cdot 127^2 = 848\,398$. Использование метода Тоома для умножения многочленов степени $4n - 1$ с разбиением на четыре равные части и использованием для умножения четырёхчленов ДПФ четвёртого порядка приводит к схеме с рекуррентной оценкой сложности

$$\mathrm{M}(GF(3^2), 4n-1) = 8\,\mathrm{M}(GF(3^2), n-1) + 16n\,\mathrm{A}\big(GF(3^2)\big) + 12(2n-1)\,\mathrm{A}\big(GF(3^2)\big) = \\ = 8\,\mathrm{M}(GF(3^2), n-1) + 14(40n - 12).$$

Отсюда получаем, что

$$\mathrm{M}(GF(3^2), 64n - 1) = 512\,\mathrm{M}(GF(3^2), n-1) + 168 \cdot 320n - 96 \cdot 126 + 14(640n - 12),$$

поэтому $\mathrm{M}(GF(3^2), 127) = 200\,216$.

4. Схемы для арифметики в поле $GF(p^{2pn})$ над полем $GF(p^{2n})$

4.1. Оценка сложности

Умножение в базисе $\{1, \rho, \ldots, \rho^{p-1}\}$ над подполем G сводится к обычному умножению многочленов степени не выше $p-1$ над полем G и приведению полученного многочлена $d_0 + d_1 X + \ldots + d_{2p-2} X^{2p-2}$ по модулю $X^p - X + 2b$.

Приведение по модулю $X^p - X + 2b$ выполняется по формулам

$$c_{p-1} = d_{p-1} + d_{2p-2}, \quad c_{p-2} = d_{p-2} - 2bd_{2p-2} + d_{2p-3},$$
$$c_{p-3} = d_{p-3} - 2bd_{2p-3} + d_{2p-4}, \ldots, \quad c_1 = d_1 - 2bd_{p+1} + d_p, \quad c_0 = d_0 - 2bd_p.$$

Так как $b = \pm 1$, эти формулы содержат $2p - 2$ сложений/вычитаний и $p - 2$ удвоений.

При вычислении fg, где $g = (\beta y\sigma - (\alpha + x + d - \rho)^{(p+1)/2})$, $d = -(2n-1)b$ и $\beta y \in k$ в алгоритме ДЛК, можно заметить, что в записи элемента $\beta y\sigma$ в базисе B может быть не более одной ненулевой координаты x_p, а в записи элемента $(\alpha + x + d - \rho)^{(p+1)/2}$ ненулевыми могут быть только координаты $x_0, \ldots, x_{(p+1)/2} \in k$, причём последняя координата равна ± 1. Поэтому

$$g = g_0 + g_1\rho + \ldots + g_{(p+1)/2}\rho^{(p+1)/2}, \quad g_0 \in G, \quad g_i \in k, \quad i > 0, \quad g_{(p+1)/2} = \pm 1.$$

Поскольку коэффициенты g имеют указанный вид, при применении школьного алгоритма число умножений в поле G уменьшается до $p\big((p+1)/2\big)$ (причём $p\big((p-1)/2\big)$ из них являются умножениями вида $k \times G$), сложений в том же поле также становится меньше. Отсюда получаем оценку сложности умножения f на g в виде $p(p+2)\,\mathrm{M}(k) + p(p+12)\,\mathrm{A}(k)$, где $\mathrm{M}(k)$, $\mathrm{A}(k)$ — сложность умножения и сложения/вычитания в поле $k = GF(p^n)$.

Использование метода Карацубы позволяет уменьшить число умножений в поле k. В [22] указаны формулы для умножения при $p = 3$ и $\rho^3 - \rho - 1 = 0$, с помощью которых можно оценить сложность умножения fg как $13\,\mathrm{M}(k) + 33\,\mathrm{A}(k)$.

В [24, 26] предлагаются алгоритмы для спариваний на гиперэллиптических кривых, в которых используется умножение произвольных элементов в поле K. Поэтому далее будут предложены схемы для умножения в этом поле элементов общего вида.

Покажем, что в общем случае число умножений в поле K можно уменьшить до $2p - 1$ с сохранением числа сложений $O(p^2)$. Для этого используем идею Тоома [7]: вычисление коэффициентов произведения многочленов

$$f_1(X)f_2(X) = (a_0 + a_1X + \ldots + a_{p-1}X^{p-1})(b_0 + b_1X + \ldots + b_{p-1}X^{p-1}) = \\ = c_0 + c_1X + \ldots + c_{2p-2}X^{2p-2} = f(X)$$

можно выполнить, вычислив $f_1(a)f_2(a) = f(a)$ в $2p - 1$ различных элементах поля $GF(p^2) \subset G$ с помощью интерполяционной формулы. В качестве узлов интерполяции удобно выбрать все элементы поля $GF(p)$ и элементы $a\sigma$, $a \in GF(p)^*$.

Поле $GF(p^2)$ есть квадратичное расширение поля $GF(p)$, оно представляет собой множество упорядоченных пар элементов из $GF(p)$, сложение и умножение которых определяются аналогично операциям в поле комплексных чисел. Тогда значения $f_j(a_i)$ для данного i, $i = 1, \ldots, 2p-1$, $a_i \in GF(p^2)$, и данного j, $j = 1, 2$, можно вычислить с помощью $p - 1$ сложений в поле G и $p - 1$ умноже-

ний в этом поле на элементы вида a или σa, где $a \in GF(p)$, по схеме Горнера. Общая сложность этих вычислений оценивается как

$$(2p-1)\cdot 2\cdot\big((p-1)\cdot 2n\,\mathrm{A}(p)+(p-1)\cdot 2n\,\mathrm{M}(p)\big)= \\ =4n(2p^2-3p+1)\big(\mathrm{A}(p)+\mathrm{M}(p)\big).$$

Значения $f_1(a)f_2(a)=f(a)$ вычисляются со сложностью $(2p-1)\,\mathrm{M}(G)=$ $=(2p-1)\big(3\,\mathrm{M}(k)+4\,\mathrm{A}(k)\big)$.

Фундаментальные многочлены интерполяции $l_i(X)$, $i=1,\dots,2p-1$, вычисляются заранее и имеют степень $2p-2$ и число коэффициентов $2p-1$ каждый. При этом коэффициенты многочленов $l_i(X)$, $i=1,\dots,p$, принадлежат подполю $GF(p)$, а у остальных многочленов — подполю $GF(p^2)$ в силу выбора узлов интерполяции. Действительно, многочлен, корни которого — узлы интерполяции, есть

$$(X-a_1)(X-a_2)\cdots(X-a_p)(X-\sigma a_{p+1})\cdots(X-\sigma a_{2p-1}),$$

где $a_j\in GF(p)$, $j\in\overline{1,2p-1}$. Произведение $(X-\sigma a_{p+1})\cdots(X-\sigma a_{2p-1})$ «вещественно», так как в нём участвует чётное число попарно «комплексно-сопряжённых» скобок. Поэтому первые p многочленов l_i, $i=\overline{1,p}$, в интерполяционной формуле «вещественны», т. е. имеют коэффициенты из $GF(p)$. Оставшиеся $p-1$ многочленов имеют коэффициенты из $GF(p^2)$ (надо ещё учесть, что каждый фундаментальный многочлен l_j получается из указанного произведения путём удаления j-й скобки и умножения на некоторый коэффициент C, как это видно из интерполяционной формулы Лагранжа; этот коэффициент также будет «вещественным» для всех $p=1,\dots,2p-1$).

Многочлен f восстанавливается по формуле

$$f(X)=\sum_{i=1}^{2p-1} f(a_i)l_i(X).$$

Для этого используется $(2p-1)p$ умножений в поле G на элементы подполя $GF(p)$ (значения коэффициентов $l_i(X)$ для $i=1,\dots,p$). Каждое умножение имеет сложность $2n\,\mathrm{M}(p)$. Также выполняются $(2p-1)(p-1)$ умножений элементов поля G на элементы подполя $GF(p^2)$ — значения коэффициентов $l_i(x)$ для $i=p+1,\dots,2p-1$. Каждое умножение имеет сложность $n\big(3\,\mathrm{M}(p)+4\,\mathrm{M}(p)\big)$. Кроме этого, согласно формуле Лагранжа требуется $(2p-2)(2p-1)$ сложений сложности $2n\,\mathrm{A}(p)$ каждое.

Общая сложность этих операций оценивается как

$$(10p^2-11p+3)n\,\mathrm{M}(p)+(16p^2-24p+8)n\,\mathrm{A}(p).$$

Суммируя полученные оценки, окончательно имеем следующую оценку сложности умножения в поле K:

$$(6p-3)\,\mathrm{M}(k)+(8p-4)\,\mathrm{A}(k)+(18p^2-23p+7)n\,\mathrm{M}(p)+(24p^2-36p+12)n\,\mathrm{A}(p).$$

Так как $\mathrm{A}(k)=n\,\mathrm{A}(p)$, то полученная оценка переписывается в виде

$$\mathrm{M}\big(GF(p^{2pn})\big)=(6p-3)\,\mathrm{M}(k)+(18p^2-23p+7)n\,\mathrm{M}(p)+(24p^2-32p+8)n\,\mathrm{A}(p).$$

Её можно переписать в виде

$$(6p-3)\,\mathrm{M}(k) + O(p^2 n \log p \,\log\log p \,\log\log\log p),$$

так как для оценки $\mathrm{M}(p)$ можно использовать метод Шёнхаге—Штрассена.

При $p = 3$ имеем оценку

$$\mathrm{M}\big(GF(3^{6n})\big) = 15\,\mathrm{M}\big(GF(3^n)\big) + 100n\,\mathrm{M}(3) + 128n\,\mathrm{A}(3).$$

Для $p = 7$ полученная выше оценка переписывается в виде

$$\mathrm{M}\big(GF(7^{14n})\big) = 39\,\mathrm{M}\big(GF(7^n)\big) + 728n\,\mathrm{M}(7) + 1079n\,\mathrm{A}(7).$$

4.2. Схемы для умножения в поле $GF(7^{14n})$ над полем $GF(7^{2n})$

4.2.1. Умножение двух многочленов шестой степени

Построим схему для умножения двух многочленов шестой степени

$$f_0 + f_1 X + \ldots + f_6 X^6, \quad g_0 + g_1 X + \ldots + g_6 X^6$$

с коэффициентами из $GF(7^{2n})$ методом Тоома. Выберем узлы интерполяции: 0, ±1, ±2, ±3, $\pm\sigma$, $\pm2\sigma$, $\pm3\sigma$. Значения в узлах многочлена f представим следующим образом:

$$\begin{aligned}
&f(0) = f_0,\\
&f(1) = \big((f_0 +_1 f_4) +_3 (f_2 +_2 f_6)\big) +_6 \big((f_1 +_4 f_5) +_5 f_3\big),\\
&f(-1) = \big((f_0 + f_4) + (f_2 + f_6)\big) -_7 \big((f_1 + f_5) + f_3\big),\\
&f(\sigma) = \big((f_0 + f_4) -_8 (f_2 + f_6)\big) +_{10} \big((f_1 + f_5) -_9 f_3\big)\sigma,\\
&f(-\sigma) = \big((f_0 + f_4) - (f_2 + f_6)\big) -_{11} \big((f_1 + f_5) - f_3\big)\sigma,\\
&f(2) = \big((f_0 +_{12} 2f_4) +_{14} (4f_2 +_{13} f_6)\big) +_{17} \big((2f_1 +_{15} 4f_5) +_{16} f_3\big),\\
&f(-2) = \big((f_0 + 2f_4) + (4f_2 + f_6)\big) -_{18} \big((2f_1 + 4f_5) + f_3\big),\\
&f(2\sigma) = \big((f_0 + 2f_4) -_{19} (4f_2 + f_6)\big) +_{21} \big((2f_1 + 4f_5) -_{20} f_3\big)\sigma,\\
&f(-2\sigma) = \big((f_0 + 2f_4) - (4f_2 + f_6)\big) -_{22} \big((2f_1 + 4f_5) - f_3\big)\sigma,\\
&f(3) = f(-4) = \big((f_0 +_{23} 4f_4) +_{25} (2f_2 +_{24} f_6)\big) +_{28} \big((4f_1 +_{26} 2f_5) -_{27} f_3\big),\\
&f(-3) = f(4) = \big((f_0 + 4f_4) + (2f_2 + f_6)\big) -_{29} \big((4f_1 + 2f_5) - f_3\big),\\
&f(3\sigma) = f(-4\sigma) = \big((f_0 + 4f_4) -_{30} (2f_2 + f_6)\big) +_{32} \big((4f_1 + 2f_5) +_{31} f_3\big)\sigma,\\
&f(-3\sigma) = f(4\sigma) = \big((f_0 + 4f_4) - (2f_2 + f_6)\big) -_{33} \big((4f_1 + 2f_5) + f_3\big)\sigma.
\end{aligned}$$

Сложность этих вычислений с учётом указанных скобками разбиений равна $33\,\mathrm{A}\big(GF(7^{2n})\big) = 66n\,\mathrm{A}(7)$. В этом равенстве учтено, что $\mathrm{A}\big(GF(7^{2n})\big) = 2n\,\mathrm{A}(7)$. Глубина этой схемы равна $3\,\mathrm{D}\big(\mathrm{A}(7)\big)$. Те же вычисления необходимо проделать и для многочлена g. Суммарная сложность равна $132n\,\mathrm{A}(7)$.

Найдём значения многочлена

$$h(X) = f(X) \cdot g(X), \quad h(X) = h_0 + h_1 X + \ldots + h_{12} X^{12},$$

в выбранных узлах a_j, $j = 0, \ldots, 12$. Сложность вычисления $h_j = h(a_j) = f(a_j)g(a_j)$, $j = 0, \ldots, 12$, оценивается как $(2p-1)\big(3\,\mathrm{M}(k) + 4\,\mathrm{A}(k)\big)$ и при $p = 7$ составляет

$$13\Big(3\,\mathrm{M}\big(GF(7^n)\big) + 4\,\mathrm{A}\big(GF(7^n)\big)\Big) = 39\,\mathrm{M}\big(GF(7^n)\big) + 52n\,\mathrm{A}(7).$$

Оценим сложность схемы для интерполяции. Пусть

$$d_0 = -h(0),\ d_1 = -4h(1),\ d_2 = -4h(-1), \ldots,\ d_{11} = -4h(3\sigma),\ d_{12} = -4h(-3\sigma).$$

Фундаментальные многочлены (после некоторой перестановки) таковы:

$$\begin{aligned}
&- 4X(X^4-4)(X^4-2)(X^2-1)(X+i)h(i),\\
&- 4X(X^4-4)(X^4-2)(X^2-1)(X-i)h(-i),\\
&- (X^4-4)(X^4-2)(X^4-1)h(0) = -(X^8+X^4+1)(X^4-1)h(0),\\
&- 4X(X^8+X^4+1)(X^2+1)(X+1)h(1),\\
&- 4X(X^8+X^4+1)(X^2+1)(X-1)h(-1),\\
&- 4X(X^4-1)(X^4-4)(X^2+4)(X+2)h(2),\\
&- 4X(X^4-1)(X^4-4)(X^2+4)(X-2)h(-2),\\
&- 4X(X^4-1)(X^4-4)(X^2-4)(X+2i)h(2i),\\
&- 4X(X^4-1)(X^4-4)(X^2-4)(X-2i)h(-2i),\\
&- 4X(X^4-1)(X^4-2)(X^2-2)(X+4i)h(3i),\\
&- 4X(X^4-1)(X^4-2)(X^2-2)(X-4i)h(-3i),\\
&- 4X(X^4-1)(X^4-2)(X^2+2)(X+4)h(-3),\\
&- 4X(X^4-1)(X^4-2)(X^2+2)(X-4)h(3).
\end{aligned}$$

Действительно, для получения k-го фундаментального многочлена нужно из произведения

$$\begin{aligned}
X \times (X-1) \times (X+1) \times (X-2) \times (X+2) \times (X-3) \times (X+3) \times\\
\times (X-i) \times (X+i) \times (X-2i) \times (X+2i) \times (X-3i) \times (X+3i)
\end{aligned}$$

удалить $(X-k)$, $k = \pm1, \pm2, \pm3, \pm i, \pm2i, \pm3i$, перемножить оставшиеся скобки и результат домножить на $h(k)$ и разделить на значение от k полученного многочлена; оно равно -2, за исключением случая $k = 0$, где оно равно -1.

Вычисление коэффициентов многочлена $h(X)$ по его значениям в 13 узлах можно представить в следующем виде:

$$\begin{aligned}
h(X) = (X^8+X^4+1)^2\big[d_0(X^4-1) +^1 X(X^2-1)\big(d_7(X+i) +^2 d_8(X-i)\big) +^4\\
+ X(X^2+1)\big(d_1(X+1) +^2 d_2(X-1)\big)\big] +^{12}
\end{aligned}$$

$$+ X(X^4-1)^4\Big[(X^4-2)\Big((X^2-2)\big(d_{11}(X+4i) +^2 d_{12}(X-4i)\big) +^4$$
$$+ (X^2+2)\big(d_5(X+4) +^2 d_6(X-4)\big)\Big) +^8$$
$$+ (X^4-4)\Big((X^2+4)\big(d_4(X-2) +^2 d_3(X+2)\big) +^4$$
$$+ (X^2-4)\big(d_9(X-2i) +^2 d_{10}(X+2i)\big)\Big)\Big].$$

Индексы k сверху над каждой операцией относится к её сложности, записанной в виде $k\,\mathrm{A}\big(GF(7^{2n})\big)$. Отсутствие индекса означает, что сложность равна нулю. Поэтому сложность этих вычислений равна

$$\mathrm{A}\big(GF(7^{2n})\big)\big[\big((2+2+4+1)+1+1\big)+12+4+\big((2+2+4)\cdot 2+8\big)\big] = \\ = 2n\,\mathrm{A}(7)\cdot 51 = 102n\,\mathrm{A}(7).$$

Глубина этой схемы равна $6\,\mathrm{D}\big(\mathrm{A}(7)\big)$.

Суммируя полученные оценки, находим, что сложность умножения многочленов степени не выше 6 над $GF(7^{2n})$ равна

$$L = 132n\,\mathrm{A}(7) + 13\,\mathrm{M}\big(GF(7^{2n})\big) + 102n\,\mathrm{A}(7) = 13\,\mathrm{M}\big(GF(7^{2n})\big) + 234n\,\mathrm{A}(7).$$

Глубина схемы равна $9\,\mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\Big(\mathrm{M}\big(GF(7^{2n})\big)\Big)$.

Приведение по модулю $X^7 - X + 2$ многочлена 12-й степени

$$d_0 + d_1X + \ldots + d_{12}X^{12} = c_0 + c_1X + \ldots + c_6X^6 \pmod{(X^7 - X + 2)}$$

выполняется по формулам

$$c_6 = d_6 + d_{12}, \quad c_5 = d_5 - 2d_{12} + d_{11},$$
$$c_4 = d_4 - 2d_{11} + d_{10}, \ldots, \quad c_1 = d_1 - 2d_8 + d_7, \quad c_0 = d_0 - 2d_7.$$

Эти формулы содержат 12 сложений/вычитаний (в $GF(7^{2n})$) и 6 удвоений (их сложность при схемной реализации равна нулю). Таким образом, приведение по модулю $X^7 - X + 2$ имеет сложность $12\,\mathrm{A}\big(GF(7^{2n})\big) = 24n\,\mathrm{A}(7)$ и глубину $2\,\mathrm{D}\big(\mathrm{A}(7)\big)$. Учитывая полученную оценку сложности умножения многочленов 6-й степени с коэффициентами из $GF(7^{2n})$, имеем

$$\mathrm{M}\big(GF(7^{14n})\big) = 13\,\mathrm{M}\big(GF(7^{2n})\big) + 258n\,\mathrm{A}(7).$$

Для глубины справедливо неравенство

$$\mathrm{D}\Big(\mathrm{M}\big(GF(7^{14n})\big)\Big) \leqslant 11\,\mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\Big(\mathrm{M}\big(GF(7^{2n})\big)\Big).$$

4.2.2. Умножение многочленов степени 6 на многочлены степени 3

В алгоритме ДЛК на самом деле используется не умножение произвольных многочленов степени $p-1$ над полем $GF(p^{2n})$, а умножение многочлена степени $p-1$ на многочлен степени $(p+1)/2$, у которого старший коэффициент равен 1.

Представляя последний в виде суммы $X^{(p+1)/2}$ и многочлена степени не выше $(p-1)/2$, получаем, что такое умножение сводится к умножению многочлена степени $p-1$ на многочлен вдвое меньшей степени и $p-1$ сложениям в поле $GF(p^{2n})$. При $p=7$ умножается многочлен степени 6 на многочлен степени 3.

Для умножения используем ДПФ порядка 16 в поле $GF(7^2)$ (о ДПФ см., например, [8]).

В качестве первообразного корня 16-й степени из 1 можно взять $w = 2+4\sigma$. Действительно,

$$
\begin{aligned}
&w_1 = w^2 = -12 + 16\sigma = 2 + 2\sigma,\\
&w_1^2 = w^4 = 4 \cdot 2\sigma = \sigma,\\
&w_1^6 = w^{12} = -w_1^2 = -\sigma,\\
&w^6 = w^2 w^4 = \sigma w^2 = \sigma w_1 = 2\sigma - 2,\\
&w_1^8 = w^{16} = 1,\\
&w^{10} = -w^2,\\
&w^{14} = -w^6,\\
&w^3 = w \cdot w_1 = -2\sigma - 4,\\
&w^5 = w \cdot w^4 = \sigma w = 2\sigma - 4,\\
&w^7 = w^3 w^4 = \sigma w^3 = 2 - 4\sigma,\\
&w^9 = -w = -2 - 4\sigma,\\
&w^{11} = -w^3 = 2\sigma + 4,\\
&w^{13} = -w^5 = 4 - 2\sigma,\\
&w^{15} = -w^7 = -2 + 4\sigma,\\
&w^8 = -1.
\end{aligned}
$$

Из этой таблицы видно, что умножение на w^4, w^8, w^{12} имеет сложность 0, так как умножение на σ в $GF(7^2)$ имеет сложность 0. Умножение на w^2, w^6, w^{10}, w^{14} имеет сложность $2\,\mathrm{A}(7)$ (умножение на степени двойки и смена знака, как отмечалось выше, имеют нулевую сложность). Так как $w^{2k+1} = \pm 2\sigma \pm 4$ или $\pm 4\sigma \pm 2$, как видно из таблицы, то умножение на w^{2k+1}, $k = \overline{0,7}$, элементов из $GF(7^2)$ имеет сложность $2\,\mathrm{A}(7)$.

Вычисляем $f(\pm 1)$, $f(\pm w^2)$, $f(\pm w^4)$, $f(\pm w^6)$ по полиномиальной схеме алгоритма ДПФ [8]:

$$
\begin{array}{lll}
f, & f \bmod (X^4-1), & f \bmod (X^4+1),\\
f \bmod (X^4-1), & f \bmod (X^2-1), & f \bmod (X^2+1),\\
f \bmod (X^4+1), & f \bmod (X^2-w^4), & f \bmod (X^2+w^4),\\
f \bmod (X^2-1), & f \bmod (X-1), & f \bmod (X+1),\\
f \bmod (X^2+1), & f \bmod (X-w^4), & f \bmod (X+w^4),
\end{array}
$$

$$f \bmod (X^2 - w^4), \qquad f \bmod (X - w^2), \qquad f \bmod (X + w^2),$$
$$f \bmod (X^2 + w^4), \qquad f \bmod (X - w^6), \qquad f \bmod (X + w^6).$$

Сложность этих вычислений составляет $48n\,\mathrm{A}(7)$ (мы учли, что $\deg f \leqslant 6$). В самом деле,

$$\begin{aligned} f_0 + f_1X + \ldots + f_6X^6 &= \\ &= (f_0 + f_1X + f_2X^2 + f_3X^3) + (f_4 + f_5X + f_6X^2) \bmod (X^4 - 1) = \\ &= (f_0 + f_4) + (f_1 + f_5)X + (f_2 + f_6)X^2 + f_3X^3, \\ f_0 + f_1X + \ldots + f_6X^6 &= \\ &= (f_0 + f_1X + f_2X^2 + f_3X^3) - (f_4 + f_5X + f_6X^2) \bmod (X^4 + 1) = \\ &= (f_0 - f_4) + (f_1 - f_5)X + (f_2 - f_6)X^2 + f_3X^3. \end{aligned}$$

Эти приведения по модулю имеют схемную сложность $6n\,\mathrm{A}\big(GF(7^2)\big)$. Вычисление второй и четвёртой строк требует $(4 + 4) + (2 + 2 + 3 + 3) = 18$ сложений/вычитаний в $GF(7^{2n})$. Суммарная сложность равна $24n\,\mathrm{A}\big(GF(7^2)\big) = 48n\,\mathrm{A}(7)$. Глубина равна $4\,\mathrm{D}\big(\mathrm{A}(7)\big)$. Сложность вычисления $g(\pm 1)$, $g(\pm w^2)$, $g(\pm w^4)$, $g(\pm w^6)$ при $\deg g \leqslant 3$ равна $36n\,\mathrm{A}(7)$. Глубина равна $3\,\mathrm{D}\big(\mathrm{A}(7)\big)$. Суммарная сложность вычисления значений многочленов f, g в выбранных узлах составляет $48n\,\mathrm{A}(7) + 36n\,\mathrm{A}(7) = 84n\,\mathrm{A}(7)$. Глубина равна $4\,\mathrm{D}\big(\mathrm{A}(7)\big)$.

Найдём значения многочлена

$$h(X) = f(X) \cdot g(X), \quad h(X) = h_0 + h_1X + \ldots + h_9X^9,$$

в выбранных узлах a_j, $j = 1, \ldots, 8$. Сложность вычисления $h_j = h(a_j) = f(a_j)g(a_j)$, $j = 0, \ldots, 8$, составляет $8\,\mathrm{M}\big(GF(7^{2n})\big)$. Применяя к вектору h_j обратное преобразование Фурье, получаем вектор коэффициентов многочлена $h_0 + h_1X + \ldots + h_9X^9 \bmod (X^8 - 1)$ — циклической свёртки. Сложность его вычисления равна $52n\,\mathrm{A}(7)$. Глубина равна $5\,\mathrm{D}\big(\mathrm{A}(7)\big)$. Очевидно,

$$h_0 + h_1X + \ldots + h_9X^9 \bmod (X^8 - 1) = (h_0 + h_8) + (h_1 + h_9)X + h_2X^8 + \ldots + h_7X^7.$$

Так как h_0 и h_9 вычисляются с помощью двух умножений коэффициентов многочленов f, g, то восстановить все коэффициенты $h(X)$ по $h(X) \bmod (X^8 - 1)$ можно с помощью двух вычитаний. Окончательная сложность умножения равна

$$10\,\mathrm{M}\big(GF(7^{2n})\big) + 84n\,\mathrm{A}(7) + 52n\,\mathrm{A}(7) + 4n\,\mathrm{A}(7) = 10\,\mathrm{M}\big(GF(7^{2n})\big) + 140n\,\mathrm{A}(7),$$

а глубина равна

$$5\,\mathrm{D}\big(\mathrm{A}(7)\big) + 4\,\mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\big(\mathrm{M}(GF(7^{2n})\big) = 10\,\mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\big(\mathrm{M}(GF(7^{2n})\big).$$

Значит, умножение в поле $GF(7^{14n})$ элемента f, представимого многочленом степени 6, на элемент g, представимый многочленом степени 4 с единичным старшим коэффициентом имеет сложность не выше

$$10\,\mathrm{M}\big(GF(7^{2n})\big) + 140n\,\mathrm{A}(7) + 24n\,\mathrm{A}(7) + 12n\,\mathrm{A}(7) = 10\,\mathrm{M}\big(GF(7^{2n})\big) + 176n\,\mathrm{A}(7).$$

Глубина схемы равна

$$13\,\mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\Big(\mathrm{M}\big(GF(7^{2n})\big)\Big).$$

Учитывая полученные оценки, при $n = 31$ получаем оценку

$$\mathrm{M}\big(GF(7^{14n})\big) \leqslant 10\,\mathrm{M}\big(GF(7^{2n})\big) + 176n\,\mathrm{A}(7) = 557\,392.$$

Глубина равна

$$31\,\mathrm{D}\big(\mathrm{A}(7)\big) + \mathrm{D}\big(\mathrm{M}(7)\big).$$

Сравним со школьным методом. Для умножения многочлена 6-й степени с коэффициентами из поля $GF(7^{2n})$ на многочлен 3-й степени, у которого все старшие коэффициенты принадлежат полю $GF(7^n)$, требуются 7 умножений в поле $GF(7^{2n})$ и 21 умножение на элемент подполя $GF(7^n)$. Общая сложность этих операций равна

$$42\,\mathrm{M}\big(GF(7^n)\big) + 21\,\mathrm{M}\big(GF(7^n)\big) + \\ + 28\,\mathrm{A}\big(GF(7^{2n})\big) + 18\,\mathrm{A}\big(GF(7^{2n})\big) + 24n\,\mathrm{A}(7) + 12n\,\mathrm{A}(7),$$

что в два раза больше предыдущей оценки.

Сведение умножения многочлена 6-й степени на многочлен 3-й степени к умножению многочленов 3-й степени и многочлена 3-й степени на многочлен 4-й степени и применению к ним метода Карацубы требует 18 умножений на элемент подполя $GF(7^n)$ и 4 умножения в поле $GF(7^{2n})$, поэтому сложность такой схемы не менее

$$36\,\mathrm{M}\big(GF(7^n)\big) + 12\,\mathrm{M}\big(GF(7^n)\big) + 16\,\mathrm{A}\big(GF(7^{2n})\big) + 24n\,\mathrm{A}(7) + 12n\,\mathrm{A}(7) + 6n\,\mathrm{A}(7).$$

Метод Карацубы

$$\begin{aligned}
&a_1b_0 + a_0b_1 = (a_0 + a_1)(b_0 + b_1) - a_0b_0 - a_1b_1,\\
&a_2b_0 + a_1b_1 + a_0b_2 = a_1b_1 + (a_0 + a_2)(b_0 + b_2) - a_0b_0 - a_2b_2,\\
&a_3b_0 + a_2b_1 + a_1b_2 + a_0b_3 =\\
&\quad = (a_0 + a_3)(b_0 + b_3) - a_0b_0 - a_3b_3 + (a_1 + a_2)(b_1 + b_2) - a_1b_1 - a_2b_2
\end{aligned}$$

и т. д. приводит к сравнимым результатам.

5. Оценка битовой сложности алгоритма ДЛК

В алгоритме ДЛК в поле K выполняются операции

$$g = \beta y\sigma - (\alpha + x + d - \rho)^{(p+1)/2}, \quad f = f^p g$$

и заключительная операция возведения в степень $p^{pn} - 1$. Как и в [22] при $p = 3$, естественно выбрать в поле K над подполем k базис, являющийся произведением базиса $\{1, \sigma\}$ поля $GF(p^2)$ над подполем $GF(p)$ и базиса $\{1, \rho, \ldots, \rho^{p-1}\}$ поля $GF(p^p)$ над подполем $GF(p)$. Минимальный многочлен первого базиса

есть X^2+1, а второго — X^p-X+2b. Произведением этих базисов является базис

$$B=\{1,\sigma\}\times\{1,\rho,\dots,\rho^{p-1}\}=\{1,\rho,\dots,\rho^{p-1},\sigma,\sigma\rho,\dots,\sigma\rho^{p-1}\},$$

являющийся как базисом поля $GF(p^{2p})$ над подполем $GF(p)$, так и базисом поля $K=GF(p^{2pn})$ над подполем $k=GF(p^n)$ при $(n,2p)=1$. Произвольный элемент поля K разлагается по этому базису следующим образом:

$$x=\sum_{i=0}^{p-1}x_i\rho^i+\sigma\sum_{i=0}^{p-1}x_{p+i}\rho^i,$$

где $x_i\in k$. Элементы

$$X_0=\sum_{i=0}^{p-1}x_i\rho^i,\quad X_1=\sum_{i=0}^{p-1}x_{p+i}\rho^i$$

принадлежат подполю $F=GF(p^{pn})$,

$$x=X_0+\sigma X_1.$$

Можно представить элемент x и в виде

$$x=\sum_{i=0}^{p-1}(x_i+x_{p+i}\sigma)\rho^i,$$

где $x_i+x_{p+i}\sigma\in G=GF(p^{2n})$.

Разлагая указанным способом элемент

$$g=\beta y\sigma-(\alpha+x+d-\rho)^{(p+1)/2},$$

получаем, что

$$g=g_1+g_2\sigma,$$

где

$$g_1=-(\alpha+x+d-\rho)^{(p+1)/2}\in F,\quad g_2=\beta y\in k\subset F.$$

У элемента g_2 в базисе $\{1,\rho,\dots,\rho^{p-1}\}$ поля F максимум одна ненулевая координата. Так как $\alpha+x+d\in k$, то элемент g_1 представи́м в виде

$$g_1=\sum_{i=0}^{(p+1)/2}u_i\rho^i,\quad u_{(p+1)/2}=1,$$

так как $(p+1)/2$ чётно, и вообще

$$(\alpha+x+d-\rho)^s=\sum_{i=0}^{s}u_i\rho^i,\quad u_s=\pm1,\quad s\leqslant p-1.$$

Для вычисления g нужно вначале вычислить $a=\alpha+x+d$, что требует одной операции сложения в поле k и одной операции сложения в поле $GF(p)$, потом

надо вычислить $a^2, \ldots, a^{(p+1)/2}$, для чего нужно $(p-1)/2$ умножений в поле k, и потом для вычисления u_i, $i = 0, \ldots, (p-1)/2$, нужно ещё не больше $(p+1)\log_2 p$ сложений в поле k (при $p = 3$ сложений вообще не нужно, при $p = 7$ тоже, так как они сводятся к удвоениям по модулю $p = 7$).

Умножение fg можно рассматривать как умножение в базисе $\{1, \rho, \ldots, \rho^{p-1}\}$ над подполем G, а умножение в поле G можно проводить в базисе $\{1, \sigma\}$, и оно сводится к трём умножениям в подполе k и пяти сложениям:

$$(a + b\sigma)(c + d\sigma) = (ac - bd) + \sigma(ad + bc) = (ac - bd) + \sigma\big((a + b)(c + d) - ac - bd\big).$$

В алгоритме на самом деле используется не умножение произвольных многочленов степени $p-1$ над полем $GF(p^{2n})$, а многочлена степени $p-1$ на многочлен степени $(p+1)/2$, у которого старший коэффициент равен 1. Представляя последний в виде суммы $x^{(p+1)/2}$ и многочлена степени $(p-1)/2$, получаем, что такое умножение сводится к умножению многочлена степени $p-1$ на многочлен вдвое меньшей степени и $p-1$ сложениям в поле $GF(p^{2n})$. Можно также уточнить, что во втором сомножителе все коэффициенты, кроме, может быть, свободного члена, принадлежат не полю $GF(p^{2n})$, а его подполю $GF(p^n)$.

Число сложений/вычитаний по модулю p можно уменьшить до четырёх аддитивных операций и двух удвоений, если воспользоваться вместо базиса $\{1, \sigma\}$ нормальным базисом $\{\gamma, \gamma^p\}$, где $\gamma = 1 + \sigma$.

Непосредственно проверяется, что

$$(a\gamma + b\gamma^p)(c\gamma + d\gamma^p) = \big((a + b)(c + d) - 2bd\big)\gamma + \big((a + b)(c + d) - 2ac\big)\gamma^p.$$

Умножение на два по модулю p сводится к приведению по этому модулю, так как удвоение в схемной реализации имеет нулевую сложность. При $p = 2^k - 1$ приведение по модулю сводится к циклическому сдвигу координат, поэтому и в этом случае требуется дополнительно только четыре аддитивных операции.

Возведение в степень p в указанном базисе сводится к перестановке координат.

Умножение произвольного элемента на элемент подполя k сводится к двум умножениям в поле k, как и в случае базиса $\{1, \sigma\}$. Умножение произвольного элемента на элемент вида $a(\gamma - \gamma^p) = 2a\sigma$ сводится к двум умножениям в поле k, как и в случае базиса $\{1, \sigma\}$ (при условии выполнения предварительных удвоений со сменой знака).

Возведение в степень p в поле K сводится к возведению в степень p элемента

$$x = d_0 + d_1 x + \ldots + d_{p-1} x^{p-1}, \quad x_i \in G, \quad i = 0, \ldots, n-1,$$

по модулю $x^p - x + 2b$. Очевидно,

$$x^p = d_0^p + d_1^p x^p + \ldots + d_{p-1}^p x^{(p-1)p}.$$

Для приведения по модулю $x^p - x + 2b$ выполняем деление в столбик. Можно проверить, что в результате получится многочлен

$$c_0 + c_1 x + \ldots + c_{p-1} x^{p-1},$$

где

$$c_i = (-2b)d_{i+1}^p\binom{i+1}{1} + (-2b)^2 d_{i+2}^p\binom{i+2}{2} + \ldots + \\ + (-2b)^{p-1-i} d_{p-1}^p\binom{p-1}{p-1-i}, \quad i = 0, \ldots, p-1.$$

Как известно, $\binom{p-1}{k} = (-1)^k \bmod p$, поэтому

$$c_i = (-2b)d_{i+1}^p\binom{i+1}{1} + (-2b)^2 d_{i+2}^p\binom{i+2}{2} + \ldots + (2b)^{p-1-i} d_{p-1}^p, \quad i = 0, \ldots, p-1.$$

Систему всех биномиальных коэффициентов по модулю p можно вычислить заранее с помощью тождества Паскаля за $p(p-1)/2$ сложений по модулю p. Количество сложений в поле G, требующихся для вычисления всех c_i по данным d_i^p, можно оценить как $p^2 \log_2 p$ (при $p = 3$ и при $p = 7$ достаточно $p(p-1)/2$ сложений, так как остальные сводятся к удвоениям по модулю p). Поэтому для вычисления возведения в степень p в поле K достаточно p раз возвести в степень p в поле k и выполнить не более $p^2 \log p$ сложений в этом же поле (при $p = 3, 7$ достаточно $p(p-1)$ сложений).

Возведение в степень p в поле k можно выполнить за не более чем $2(p-1)n$ сложений в поле $GF(p)$ при выборе стандартного базиса с неприводимым многочленом, имеющим не более трёх членов. В случае использования нормального базиса сложность этой операция практически нулевая. Поэтому битовая сложность возведения в степень p в поле K оценивается как $(4(p-1)p + p^2 \log p)n\,\mathrm{A}(p)$, где $\mathrm{A}(p)$ — сложность сложения по модулю p (при $p = 3$ и при $p = 7$ оценка имеет вид $5(p-1)pn\,\mathrm{A}(p)$).

Заключительная операция алгоритма — возведение в степень $p^{pn} - 1$ в поле K. Она сводится к pn-кратному возведению в степень p и последующему делению на основание. Битовая сложность возведения в степень p^{pn} в поле K оценивается как $p(4(p-1)p + 2p^2 \log p)n^2\,\mathrm{A}(p)$ (при $p = 3$ и при $p = 7$ оценка имеет вид $5(p-1)(pn)^2\,\mathrm{A}(p)$).

Деление сводится к инвертированию и умножению. В [23] было показано, что для проверки цифровой подписи не надо вычислять полностью указанную степень, так как для сравнения двух значений спаривания Тейта $x^{p^{pn}-1}$ и $y^{p^{pn}-1}$ достаточно сравнить $x^{p^{pn}}/x$ и $y^{p^{pn}}/y$ (или, что то же самое, $x^{p^{pn}}x$ и $y^{p^{pn}}y$) и тем самым избежать деления.

Для инвертирования предста́вим элемент $x \in K$ в виде $a + b\sigma$, где $a, b \in F$ (сложность нулевая), и вычислим обратный элемент по формулам

$$x^{-1} = \frac{a - b\sigma}{(a + b\sigma)(a - b\sigma)} = \frac{a - b\sigma}{(a^2 + b^2)} = \frac{a}{(a^2 + b^2)} - \frac{b}{(a^2 + b^2)}\sigma.$$

Инвертирование согласно этим формулам сводится к двум возведениям в квадрат, двум умножениям, одному инвертированию и двум вычитаниям в поле F.

Инвертирование в поле F можно свести к инвертированию в поле k и зависящему от p количеству сложений/вычитаний и умножений в поле k. Например,

при $p = 3$, $b = -1$, согласно [22], инвертирование в поле $F = GF(3^{3n})$ требует 3 возведений в квадрат, 12 умножений, 12 сложений/вычитаний и одного инвертирования в поле $k = GF(3^n)$.

Учитывая сделанные выше замечания, можно оценить сложность алгоритма ДЛК как

$$(4(p-1)+2p\log p)n\,\mathrm{A}(p)+n\Bigg(n\big((2p+4)(2(p-1)+p\,\log p)+$$
$$+(p+1)\log_2 p+2+p(4(p-1)p+2p^2\,\log p)\big)\,\mathrm{A}(p)+(p+1)\frac{\mathrm{M}(k)}{2}+\mathrm{M}(K)\Bigg),$$

где $\mathrm{M}(k)$ — сложность умножения в произвольном поле k, $\mathrm{A}(p)$ — сложность аддитивных операций в поле $GF(p)$. При $p = 3$ и при $p = 7$ оценка имеет вид

$$4(p-1)n\,\mathrm{A}(p)+n\Bigg(n\big((p-1)(5p(p+1)+8)+2\big)\,\mathrm{A}(p)+(p+1)\frac{\mathrm{M}(k)}{2}+\mathrm{M}(K)\Bigg).$$

При использовании нормального базиса в поле $k = GF(p^n)$ последняя оценка приобретает вид

$$n\left(n\big((p-1)p+2\big)\,\mathrm{A}(p)+(p+1)\frac{\mathrm{M}(k)}{2}+\mathrm{M}(K)\right).$$

Но главный член в этих оценках всегда $n\,\mathrm{M}(K)$.

6. Сравнение битовой сложности алгоритма ДЛК при $p = 3$ и $p = 7$

Вычисление fg в алгоритме ДЛК в случае $p = 3$ выполняется со сложностью $13\,\mathrm{M}(k) + 33\,\mathrm{A}(k)$ (см. [22]), поэтому оценка сложности булевой схемы, реализующей алгоритм ДЛК, в этом случае равна $15n\,\mathrm{M}\big(GF(3^n)\big) + 56n + 1197n^2$. Асимптотически эта оценка равна $15n\,\mathrm{M}\big(GF(3^n)\big)$. Минимальное значение n, рекомендуемое в [10], равно 79. При использовании школьного алгоритма умножения многочленов имеем оценку

$$\mathrm{M}\big(GF(3^n)\big) \geqslant n^2\,\mathrm{M}(3) + (n-1)^2\,\mathrm{A}(3) = 6n^2 + 7(n-1)^2 = 13n^2 - 14n + 7,$$

откуда следует, что

$$\frac{15n\,\mathrm{M}(GF(3^n))}{56n + 1197n^2} \geqslant \frac{15(13n^2 - 14n + 7)}{1197n + 56} > 10.$$

С ростом n эта величина растёт с линейной скоростью. При использовании метода Карацубы [6] для умножения многочленов степени n имеем при $n = 2^k$ оценку

$$\mathrm{M}\big(GF(3^n)\big) \geqslant n^{\log_2 3}\,\mathrm{M}(3) + (6n^{\log_2 3} - 8n + 2)\,\mathrm{A}(3),$$

откуда при, например, $n = 251$ (значение взято из [19]) имеем

$$\frac{15n\,\mathrm{M}(GF(3^n))}{1197n^2 + 56n} \geqslant \frac{15(48n^{\log_2 3} - 56n + 14)}{1197n + 56} > 11.$$

Таким образом, и в реальном случае сложность алгоритма ДЛК с погрешностью в несколько процентов оценивается как $15n\,\mathrm{M}(GF(3^n))$.

Рассмотрим случай $p = 7$. Как показано выше, схемная сложность умножения в поле $GF(7^{14m})$ оценивается сверху как $\mathrm{M}\big(GF(7^{14n})\big) = 39\,\mathrm{M}\big(GF(7^n)\big) + 310n\,\mathrm{A}(7)$. Поэтому сложность булевой схемы, построенной с использованием описанной выше схемы для умножения в поле $GF(7^{14n})$ и реализующей алгоритм ДЛК в этом случае, оценивается как

$$\begin{aligned} &24n\,\mathrm{A}(7) + 1730\,\mathrm{A}(7)n^2 + n\big(4\,\mathrm{M}(k) + \mathrm{M}(K)\big) = \\ &\quad = 43n\,\mathrm{M}\big(GF(7^n)\big) + 2040n^2\,\mathrm{A}(7) + 24n\,\mathrm{A}(7) = \\ &\quad = 43n\,\mathrm{M}\big(GF(7^n)\big) + 34\,580n^2 + 408n. \end{aligned}$$

Асимптотически эта оценка равна $43n\,\mathrm{M}\big(GF(7^n)\big)$.

Степень надёжности криптосистем, основанных на алгоритмах спаривания, определяется двоичным логарифмом порядка поля $GF(p^{2pn}$. Если он равен приблизительно 1000, то считается, что уровень надёжности такой же, как у криптосистем, основанных на потенцировании в мультипликативной группе поля $GF(2^{1024})$ и рекомендуемых соответствующими стандартами. При $p = 7$ это означает, что $n \geqslant 1024/(2p\log_2 p) > 26$.

При использовании школьного алгоритма умножения многочленов имеем оценку

$$\mathrm{M}\big(GF(7^n)\big) \geqslant n^2\,\mathrm{M}(7) + (n-1)^2\,\mathrm{A}(7) = 33n^2 + 17(n-1)^2 = 50n^2 - 34n + 17,$$

откуда следует, что

$$\frac{43n\,\mathrm{M}(GF(7^n))}{408n + 34\,580n^2} \geqslant \frac{43(50n^2 - 34n + 17)}{34\,580n + 408} > 1{,}6.$$

С ростом n эта величина растёт с линейной скоростью. Использование метода Карацубы для умножения такой малой степени существенного уменьшения сложности не даёт. Поэтому при $p = 7$ сложность алгоритма ДЛК уже начиная с уровня надёжности 1024 оценивается как $43n\,\mathrm{M}\big(GF(7^n)\big)$ с погрешностью около 6 процентов (с ростом уровня надёжности погрешность уменьшается с линейной скоростью).

Для того чтобы оба варианта алгоритма имели одинаковый уровень надёжности, равный v, нужно, чтобы порядки полей $GF(3^{6n})$ и $GF(7^{14m})$ были близки к 2^v. Поэтому $n \approx v/(6\log_2 3) = 0{,}105\,v$, $m \approx v/(14\log_2 7) = 0{,}025\,v$, $m = 0{,}242\,n$. Сравним при таком соотношении полученные оценки сложности алгоритма ДЛК при $p = 3$ и $p = 7$. Первая оценка приблизительно равна $15n\,\mathrm{M}\big(GF(3^n)\big)$, а вторая — $43m\,\mathrm{M}\big(GF(7^m)\big)$. В случае реализации умножения в этих полях в стандартном базисе с использованием стандартного алгоритма

умножения многочленов отношение первой оценки ко второй приблизительно равно

$$\frac{15n(n^2\,\mathrm{M}(3)+(n-1)^2\,\mathrm{A}(3))}{43m(m^2\,\mathrm{M}(7)+(m-1)^2\,\mathrm{A}(7))}=\frac{15n(13n^2-14n+7)}{43m(50m^2-34m+17))}$$

(при выборе неприводимых многочленов в обоих базисах с числом одночленов не больше трёх сложность приведения результата по модулю базисных многочленов мала по сравнению со сложностью умножения многочленов). Асимптотически эта дробь равна

$$\frac{195}{2150}\left(\frac{n}{m}\right)^3>6{,}7.$$

Значит, при $p=7$ и том же уровне надёжности алгоритм ДЛК работает быстрее асимптотически в 6 раз. При выборе реальных значений m, n, $m>26$, это соотношение уменьшается, но всегда остаётся больше 3.

При использовании для умножения многочленов метода Карацубы указанная дробь уменьшается и будет приблизительно равна

$$\frac{15n(n^{\log_2 3}\,\mathrm{M}(3)+(6n^{\log_2 3}-8n+2)\,\mathrm{A}(3))}{43m(m^{\log_2 3}\,\mathrm{M}(7)+(6m^{\log_2 3}-8m+2)\,\mathrm{A}(7))}\approx$$
$$\approx\frac{15\cdot(\mathrm{M}(3)+6\,\mathrm{A}(3))}{43\cdot(\mathrm{M}(7)+6\,\mathrm{A}(7))}\left(\frac{n}{m}\right)^{\log_2 6}=\frac{15\cdot 48}{43\cdot 135}\left(\frac{n}{m}\right)^{\log_2 6}=\frac{720}{5805}\left(\frac{n}{m}\right)^{\log_2 6}>4{,}8.$$

Поэтому и при использовании метода Карацубы умножения многочленов при $p=7$ алгоритм ДЛК работает быстрее.

7. Программная реализация схем умножения многочленов над полем $GF(7^2)$

Сделаем несколько замечаний о сложности программного вычисления произведения многочленов.

7.1. Векторная реализация арифметических операций в полях $GF(7)$ и $GF(7^2)$

Векторную реализацию арифметических операций в полях $GF(7)$ и $GF(7^2)$ можно выполнить так же, как и для полей характеристики 3. Вектор длины n с компонентами из поля $GF(7^2)$ можно представить как пару векторов длины n с компонентами из поля $GF(7)$, а каждый из этих векторов можно представить как тройку булевых векторов x_2, x_1, x_0, при этом у вектора $2^2x_2+2x_1+x_0=x$ компоненты принадлежат множеству $\{0,1,2,3,4,5,6,7\}$.

Каждая из операций любой из схем может выполняться не над парой булевых значений, а над парой булевых векторов. Тогда, например, схема для сложения векторов длины n в поле $GF(7^2)$ превращается в программу, в которой все операции выполняются над булевыми векторами длины n и число этих

операций равно сложности схемы сложения в поле $GF(7^2)$. Если n равно длине машинного слова k, то каждая из таких операций делается с помощью одной машинной команды. Если длина вектора больше k, то в случае его представления как вектора длины не более $\lceil n/k \rceil$, состоящего из машинных чисел, каждая из операций с векторами выполняется с помощью $\lceil n/k \rceil$ машинных операций.

То же самое справедливо для покомпонентного умножения векторов в поле $GF(7^2)$. Поэтому операция покомпонентного умножения векторов длины n над полем $GF(7^2)$ выполняется с помощью $134\lceil n/k \rceil$ машинных команд, а операция покомпонентного сложения векторов длины n над полем $GF(7^2)$ выполняется с помощью $34\lceil n/k \rceil$ машинных команд.

Операция обнуления части компонент вектора длины n выполняется с помощью $6\lceil n/k \rceil$ машинных команд (так как каждый из шести булевых векторов нужно побитово умножить на подходящую битовую маску). Операция смены знака имеет такую же сложность (так как сводится к побитовому сложению по модулю 2 каждого из шести векторов с вектором, состоящим из одних единиц). Операция скалярного умножения на 2 и 4 сводится к циклической перестановке каждой тройки булевых векторов, которая делается с помощью $9\lceil n/k \rceil$ пересылок. Операция скалярного умножения на i сводится к перестановке троек булевых векторов и смене знака у одной тройки, поэтому она выполняется с помощью $12\lceil n/k \rceil$ машинных команд. Операция скалярного умножения на нетривиальную константу из поля $GF(7^2)$ сводится к двум умножениям на степени двойки, умножению на i, возможно двум сменам знака, и сложению. Поэтому она выполняется с помощью $67\lceil n/k \rceil$ машинных команд.

Векторные операции сложения и умножения над полем $GF(7^2)$ можно ускорить, если воспользоваться хранимой в памяти таблицей сложения и умножения в этом поле. Тогда сложность каждой векторной операции будет равна n — длине вектора, элементарной операцией является извлечение результата из таблицы (двумерного массива). Можно воспользоваться таблицей большего размера 2^{25} Кб $= 32$ Мб для покомпонентного сложения и умножения двумерных векторов над полем $GF(7^2)$. Тогда число элементарных операций уменьшится в два раза (но возрастёт время доступа к таблице). Размеры таблиц можно уменьшить в два раза, если пользоваться коммутативностью. Таблицу умножения можно заменить на существенно меньшие таблицу логарифмирования по первообразному корню и таблицу потенцирования, тогда операция умножения заменится на трёхкратное использование этих таблиц, операцию сложения по модулю 48 чисел (или двумерных векторов) и две операции сравнения с нулём (нужные в случае, когда один из сомножителей нулевой). Сложность выполнения такой операции повышается, но остаётся меньшей, чем сложность векторного умножения без использования таблиц.

Использование таблиц при умножении можно вообще устранить, заменив при сложении однократным использованием таблицы гауссовых логарифмов $\mathrm{LG}(x) = \log_g(1 + g^x)$, но при этом для выполнения сложения используются операции сложения и вычитания по модулю 48 согласно формуле $\log_g(g^x + g^y) = x + \mathrm{LG}(x - y)$, а также две операции сравнения с «логарифмом нуля».

Сложность выполнения операции умножения при этом уменьшается, сложность сложения увеличивается, но остаётся сравнимой со сложностью векторного сложения, зато используется только одна таблица гауссовых логарифмов размера 48 байт, таблица логарифмирования по первообразному корню g и таблица потенцирования, причём последние используются только в начале работы программы, когда все данные логарифмируются, и в конце работы, когда полученные результаты потенцируются. При использовании двумерных векторов операции их сложения/вычитания по модулю 48 усложняются вдвое, их использовать становится невыгодно. Тогда операцию сложения можно выполнять с помощью таблицы, а операцию вычитания свести к сложению с помощью смены знака по модулю 48, которую можно выполнять с помощью таблицы малого размера. Поэтому указанный метод, несколько уменьшая скорость вычисления, уменьшает почти в два раза объём используемых таблиц, так как таблица для сложения двумерных векторов по модулю 48 используется и для сложения, и для покомпонентного умножения.

7.2. Стандартный алгоритм умножения многочленов

Для умножения многочленов степени n представляем их в виде

$$a_0 + a_1 Y + \ldots + a_{\lfloor n/2 \rfloor} Y^{\lfloor n/2 \rfloor}, \quad Y = X^2, \quad \deg a_i \leqslant 1,$$

тогда их умножение выполняется за $(\lfloor n/2 \rfloor + 1)^2$ умножений двучленов и $(\lfloor n/2 \rfloor)^2$ сложений трёхчленов, после чего необходимо ещё $2\lfloor n/2 \rfloor$ сложений двучлена с одночленом, чтобы получить окончательный результат. Умножения двучленов можно выполнять с помощью таблицы. Так как поле $GF(7^2)$ можно задавать шестибитным вектором, то размер таблицы умножения будет 2^{25} Кб $= 32$ Мб. Её надо будет заранее загрузить в память машины. Выполнять с помощью таблицы сложение трёхчленов уже невозможно, так как объём таблицы будет 2^{39} Кб. Однако можно обойтись таблицей для сложения двучленов, если после вычисления каждой строки в алгоритме умножения прибавлять старший коэффициент каждого полученного после очередного умножения трёхчлена к младшему коэффициенту следующего трёхчлена, при этом потребуется дополнительно $(\lfloor n/2 \rfloor + 1)\lfloor n/2 \rfloor$ таких сложений. Общее число операций извлечения из таблицы будет равно $(2\lfloor n/2 \rfloor + 1)\lfloor n/2 \rfloor + (\lfloor n/2 \rfloor + 1)^2$. Этот метод будет быстрее метода непосредственного использования векторных сложений и умножений, сложность которого легко снизу оценить как $34n^2/k + 134n^2/k$, где k — длина машинного слова (не больше 64).

8. Умножение многочленов над $GF(p)$ и $GF(p^2)$

Обозначим через l количество бинарных разрядов, которые необходимы для представления одного элемента поля $GF(p)$, а через m количество бинарных разрядов в машинном слове; $k = \lfloor m/l \rfloor$. Например, в случае восьмиразрядного

машинного слова при реализации операций в $GF(3)$ m равно 4, а при реализации операций в $GF(7)$ m равно 2. При этом неиспользуемые биты машинного слова игнорируются. Для реализации арифметических операций в поле $GF(p^n)$, где $2 < p < 256$, многочлен можно представить в виде

$$\begin{aligned}&(a_0X^0 + a_1X^1 + \ldots + a_2X^{m-1})X^0 + \\ &\quad + (a_mX^0 + a_{m+1}X^1 + \ldots + a_{2m-1}X^{m-1})X^m + \ldots + \\ &\quad + (a_{n-m}X^0 + \ldots + a_{n-1}X^{m-1})X^{\lfloor \frac{n-1}{m} \rfloor},\end{aligned}$$

где $a_i \in GF(p)$.

При выполнении операций сложения, вычитания и умножения многочленов для выполнения соответствующих операций над их коэффициентами, т. е. операций в поле $GF(p)$ или $GF(p^2)$, используются предварительно построенные таблицы. Для ускорения вычислений используются не таблицы выполнения этих операций в полях $GF(p)$ и $GF(p^2)$, а таблицы, выполняющие операции с машинными словами, применение которых соответствует одновременному выполнению k операций в $GF(p)$ или $GF(p^2)$. Это ускорение накладывает дополнительные ограничения на размер m машинного слова. Так, для выполнения операции сложения или вычитания двух восьмиразрядных машинных слов необходима таблица размером 65 Кб, а для их умножения таблица размером 128 Кб. Дальнейшее увеличение размера машинного слова не имеет смысла, так как для выполнения операции сложения двух шестнадцатиразрядных машинных слов требуется таблица размером в 4 Гб, что неприемлемо. Однако для некоторых элементарных операций, таких как обнуление и присваивание, возможно использование машинных слов большего размера. Поэтому в реализации используется тридцатидвухразрядное базовое слово, а для выполнения табулируемых операций в каждом тридцатидвухразрядном слове выделяются восьмиразрядные слова. Данный подход позволяет использовать одни и те же алгоритмы для любых полей $GF(p)$, где $p < 256$. Классический алгоритм умножения приведён ниже.

Вход: векторы A, B длины n коэффициентов многочлена, m — количество элементов поля $GF(p)$ в машинном слове. Для простоты полагаем, что n кратно m, в противном случае к вектору справа приписывается необходимое количество нулевых старших разрядов.

Выход: вектор res длины $2n$, задающий коэффициенты произведения полиномов.

$\mathrm{res} := 0$
$i := 0$
пока $i < n/m$
    $j := 0$;
    пока $j < n/m$
        $Z := \mathrm{Tm}[A_j, B_i]$
        $\mathrm{res}_{i+j} := \mathrm{Ta}[\mathrm{res}_{i+j}, Z_0]$
        $\mathrm{res}_{i+j+1} := \mathrm{Ta}[\mathrm{res}_{i+j+1}, Z_1]$

$j := j + 1$
$i := i + 1$
вернуть res

В данном алгоритме $\mathrm{Ta}[X, Y]$ — операция адресации по таблице сложения в поле $GF(p)$, $\mathrm{Tm}[X, Y]$ — операция адресации по таблице умножения в поле $GF(p)$.

Аналогичным образом можно рассматривать и другие алгоритмы, в частности метод Карацубы, комбинированный метод и их схемную реализацию, описанные в [1, 3].

Рассматриваемый ниже метод представляет собой комбинацию метода Карацубы и одной из реализаций метода сдвигов и сложений. При умножении комбинированным методом многочленов степени не выше n выполняются $\log_2(n_1)$ итерации прямого хода рекурсии по методу Карацубы, затем выполняется умножение полиномов степени не выше n_2 по методу сдвигов и сложений и проводится обратный ход рекурсии метода Карацубы. Так как на этапе умножения методом сдвигов и сложений используется таблицы умножения, будем предполагать, что многочлены степени n_2 имеют коэффициенты, являющиеся многочленами степени не выше n_3, тогда перемножаемые комбинированным методом многочлены имеют степени не выше $n = n_1 \cdot n_2 \cdot n_3$. Этот подход накладывает определённые ограничения на реализацию метода Карацубы. Так, например, на каждом шаге рекурсии необходимо затрачивать дополнительные операции на перераспределение перемножаемых частей многочленов, обеспечивая возможность перехода к умножению другим методом.

Оценка сложности схемы комбинированного метода

Для уточнения оценки сложности используем приём, предложенный в [5]. В случае характеристики поля $p = 2$ параметр n_3 равен 8 и операции сложения двух многочленов можно производить с использованием машинных операций. В таком случае в схеме умножения можно выделить следующие операции:

1) операция присваивания;
2) операция сложения по модулю 2;
3) операция адресации элемента в массиве;
4) обращение к таблице умножения;
5) адресация внутри базового слова.

Выполнение каждой из этих операций занимает определённое, вообще говоря разное, время, но поскольку все действия в схеме производится с различными элементами рабочего массива, то каждая из операций присваивания или сложения сопровождается одной или несколькими операциями адресации. Поэтому для получения предварительной оценки сложности можно использовать число, характеризующее количество операций адресации, впоследствии эту оценку

можно уточнять, используя остальные типы операций или сравнивая время выполнения полученных программ. В рассматриваемом случае функция сложности схемы будет иметь вид

$$P = C_1 3^{\log_2 n_1} \cdot n_2^2 + C_2 3^{\log_2 n_1} \cdot n_2 + C_3 \cdot n_1 \cdot n_2 + C_4 \cdot 3^{\log_2 n_1} + C_5. \qquad (1)$$

Зафиксировав набор значений параметров n_1, n_2, можно построить таблицу значений функций, входящих в формулу для определения сложности (1), и определить количество операций адресации, задействованных в схеме. Результаты подсчёта количества операций и вычислений коэффициентов приведены в таблице 1.

Таблица 1. Значения функции сложности, используемые для решения системы (2) при $p = 2$

n	n_3	n_1	n_2	c_1	c_2	c_3	c_4	c_5	Количество адресаций
32	8	1	4	16	4	4	1	1	135
64	8	2	4	48	12	8	3	1	441
128	8	2	8	192	24	16	3	1	1619
512	8	8	8	1728	216	64	27	1	15 419
512	8	4	16	2304	144	64	9	1	18 965
512	8	2	32	3072	96	64	3	1	24 311

Решая в рациональных числах систему уравнений

$$\begin{pmatrix} 16 & 4 & 4 & 1 & 1 \\ 48 & 12 & 8 & 3 & 1 \\ 192 & 24 & 16 & 3 & 1 \\ 1728 & 216 & 64 & 27 & 1 \\ 2304 & 144 & 64 & 9 & 1 \end{pmatrix} = \begin{pmatrix} 135 \\ 441 \\ 1619 \\ 15\,419 \\ 18\,965 \end{pmatrix}, \qquad (2)$$

получим вектор коэффициентов C_i:

$$(C_1; C_2; C_3; C_4; C_5) = (7{,}75; 12{,}5; -11; 1; 4).$$

В итоге формула сложности принимает вид

$$P = 7{,}75 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 12{,}5 \cdot 3^{\log_2 n_1} \cdot n_2 - 11 \cdot n_1 \cdot n_2 + 3^{\log_2 n_1} + 4.$$

Заметим, что значения последней строчки таблицы 1 для $n_1 = 2$ и $n_2 = 32$ в вычислениях констант не участвовали и потому могут быть использованы для проверки полученного результата подстановкой:

$$7{,}75 \cdot 3072 + 12{,}5 \cdot 96 - 11 \cdot 64 + 3 + 4 = 24\,311,$$

т. е. фактическое количество операций совпадает с расчётным.

Аналогичным образом вычисляются и функции для определения количества других операций. Данные функции приведены ниже.

Функция сложности для количества операций сложения по модулю 2:

$$2{,}125 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 2 \cdot 3^{\log_2 n_1} \cdot n_2 - 2 \cdot n_1 \cdot n_2 + 0{,}5 \cdot 3^{\log_2 n_1} + 0{,}5.$$

Функция сложности для количества операций присваивания:

$$3{,}4375 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 8 \cdot 3^{\log_2 n_1} \cdot n_2 - 7 \cdot n_1 \cdot n_2 + 0{,}5 \cdot 3^{\log_2 n_1} - 0{,}5.$$

Функция сложности для количества обращений к таблице умножения:

$$3^{\log_2 n_1} \cdot n_2^2.$$

Функция сложности для количества адресаций к рабочему массиву схемы:

$$1{,}375 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 12{,}5 \cdot 3^{\log_2 n_1} \cdot n_2 - 11 \cdot n_1 \cdot n_2 + 1 \cdot 3^{\log_2 n_1} - 1.$$

Функция сложности для количества адресаций внутри базового слова:

$$5{,}375 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 5.$$

При умножении в поле $GF(p^n)$, $p > 2$, дополнительно возникает необходимость табулирования операций сложения и вычитания, для реализации которых в $GF(2^n)$ использовались машинные операции «исключающего или» (XOR). Это значительно усложняет схему комбинированного метода по сравнению со схемой в поле характеристики 2. Дополнительно с увеличением характеристики поля схема неявно усложняется вследствие увеличения количества бинарных разрядов, которые отводятся на один элемент поля $GF(p)$. Для вычисления оценки количества операций адресации зафиксируем $p = 3$. Параметр n_3 зависит только от характеристики поля (при $p = 3$ n_3 равен 4) и может быть учтён в константах, поэтому функция сложности имеет вид (2). Для определения коэффициентов C_i, $i = 1, \ldots, 5$, вновь можно выполнить пять построений схем при различных значениях параметров n_1 и n_2. Значения функции $P(n_1, n_2)$ и коэффициентов при константах приведены в таблице 2.

Таблица 2. Значения функции сложности, используемые для решения системы (3) при $p = 3$

n	n_3	n_1	n_2	c_1	c_2	c_3	c_4	c_5	Количество адресаций
16	4	1	4	16	4	4	1	1	258
32	4	2	4	48	12	8	3	1	1174
64	4	2	8	192	24	16	3	1	3776
256	4	8	8	1728	216	64	27	1	13 000
256	4	4	16	2304	144	64	9	1	42 368
256	4	2	32	3072	96	64	3	1	49 628

Решая систему линейных уравнений

$$\begin{pmatrix} 16 & 4 & 4 & 1 & 1 \\ 48 & 12 & 8 & 3 & 1 \\ 192 & 24 & 16 & 3 & 1 \\ 1728 & 216 & 64 & 27 & 1 \\ 2304 & 144 & 64 & 9 & 1 \end{pmatrix} = \begin{pmatrix} 258 \\ 1174 \\ 3776 \\ 13\,000 \\ 42\,368 \end{pmatrix}, \tag{3}$$

получим вектор коэффициентов

$$(C_1; C_2; C_3, C_4, C_5) = (15; 107{,}5; -106; 0; 12).$$

Таким образом, функция сложности будет иметь вид

$$15 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 107{,}5 \cdot 3^{\log_2 n_1} \cdot n_2 - 106 \cdot n_1 \cdot n_2 + 12.$$

Используя последнюю строчку таблицы 2, не участвовавшую в вычислениях, можно выполнить проверку результата:

$$15 \cdot 3072 + 107{,}5 \cdot 96 - 106 \cdot 64 + 12 = 49\,628.$$

Аналогично определяется количество операций и других типов. Результаты приведены ниже.

Функция сложности для количества операций присваивания:

$$3{,}8125 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 20{,}75 \cdot 3^{\log_2 n_1} \cdot n_2 - 19{,}75 \cdot n_1 \cdot n_2 + 5.$$

Функция сложности для количества обращений к таблицам умножения, сложения и вычитания:

$$3{,}5 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 14 \cdot 3^{\log_2 n_1} \cdot n_2 - 14 \cdot n_1 \cdot n_2.$$

Функция сложности для количества адресаций к рабочему массиву схемы:

$$2{,}25 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 51{,}5 \cdot 3^{\log_2 n_1} \cdot n_2 - 50 \cdot n_1 \cdot n_2 + 4.$$

Функция сложности для количества адресаций внутри базового слова:

$$9{,}25 \cdot 3^{\log_2 n_1} \cdot n_2^2 + 42 \cdot 3^{\log_2 n_1} \cdot n_2 - 42 \cdot n_1 \cdot n_2 + 8.$$

Для обобщения данного результата на $GF(p^n)$ заметим, что все операции в поле $GF(p)$ задаются таблицами, поэтому увеличение характеристики поля не требует перегенерации схемы, но влияет на её сложность посредством параметра n_3. Так, например, заменив таблицы операций в $GF(3)$ таблицами операций в $GF(7)$, схему, сгенерированную для умножения многочленов степени 256 над полем характеристики 3, можно использовать для умножения многочленов степени 128 над полем характеристики 7. Если обозначить схему умножения многочленов степени n над полем $GF(p)$ как $S(n, p)$, то при условии $p < 256$ получим $S(n, 2) = S\big((n/8) \cdot \lfloor 8/(\lceil \log_2(p) \rceil) \rfloor, p\big)$. Таким образом, для любого p, $p < 2^8$, можно построить таблицу, аналогичную таблице 1.

Обратим внимание, что различие таблиц 1 и 2 является следствием различия состава операций, выполняемых на этапе умножения методом сдвигов и сложений, в частности использованием при $p = 3$ таблиц сложения и вычитания. Это

же является причиной различия вычисленных для этих двух случаев векторов коэффициентов уравнения (2). Декомпозиционная схема умножения на этапе использования метода Карацубы остаётся неизменной при любых характеристиках поля из указанного диапазона. При этом вычисляемые для конкретных значений p, $p > 2$, таблицы будут отличаться от таблицы 2 лишь числами n_3 во втором столбце: вместо 4 будут 2, если $p = 11$ или $p = 13$, и 1 при бóльших p, не превышающих 2^8.

При переходе к операциям в полях $GF(p^{2n})$ для реализации операции умножения в поле $GF(p^2)$ можно использовать формулу $(a + b\sigma) \cdot (c + d\sigma) = (ac - bd) + (ad + bc)\sigma$. Составление байтовых таблиц для этой операции в $GF(3)$ и в $GF(7)$ позволит использовать рассматриваемые выше алгоритмы и для умножения в полях $GF(3^{2n})$ и $GF(7^{2n})$. При этом $S(n, p) = S(n/2, p^2)$.

В таблице, аналогичной таблице 2, $n_3 = 2$ для полей $GF(3^{2n})$ и $n_3 = 1$ для полей $GF(7^{2n})$.

В таблицах 3, 4 показаны преимущество комбинированного метода и схемы над методом сдвигов и сложений, где в первой строке приведены результаты для метода сдвигов и сложений с таблицей умножения восьмиразрядных слов; во второй строке приведены результаты для метода Карацубы с таблицей умножения восьмиразрядных слов, циклическая реализация; в третьей строке приведены результаты для метода Карацубы с использованием метода сдвигов и сложений для умножения шестидесятичетырёхразрядных слов и таблицы умножения восьмиразрядных слов, циклическая реализация; в четвёртой строке приведены результаты для декомпозиционной схемы по методу Карацубы с использованием метода сдвигов и сложений для умножения шестидесятичетырёхразрядных слов и таблицы умножения восьмиразрядных слов.

Таблица 3. Сравнение различных методов для операции умножения в $GF(2^n)$

Метод	$n < 64$	$2^6 \leqslant n < 2^7$	$2^7 \leqslant n < 2^8$	$2^8 \leqslant n < 2^9$	$2^9 \leqslant n < 2^{10}$
1	100 %	100 %	100 %	100 %	100 %
2	250 %	235 %	293 %	233 %	188 %
3	150 %	118 %	98 %	80 %	51 %
4	1033 %	400 %	188 %	80 %	43 %

Из таблицы 4 видно, что для поля $GF(p^n)$, где $n/l < 64$, оптимальным будет алгоритм сдвигов и сложений; если $64 < n/l < 128$, предпочтительной является циклическая реализация метода комбинации метода Карацубы и метода сдвигов и сложений; если $128 < n/l < 1024$, то рекомендуется использовать последовательную (схемную) реализацию комбинации метода Карацубы и метода сдвигов и сложений.

Таблица 4. Сравнение различных методов для операции умножения в $GF(p^n)$

Метод	$n/l < 64$	$2^6 \leqslant n/l < 2^7$	$2^7 \leqslant n/l < 2^8$	$2^8 \leqslant n/l < 2^9$	$2^9 \leqslant n/l < 2^{10}$
1	100 %	100 %	100 %	100 %	100 %
2	113 %	91 %	72 %	57 %	41 %
3	—	86 %	67 %	51 %	37 %
4	763 %	153 %	63 %	43 %	29 %

9. Заключение

Методы, алгоритмы и оценки, полученные в настоящей статье, были использованы для оценки сложности реализации алгоритма Дуурсма, Ли и Квона. Из этих оценок следует, что с увеличением характеристики поля его сложность существенно понижается. Этот алгоритм может использоваться в основанных на спариваниях криптографических протоколах, например в трёхстороннем протоколе согласования ключей А. Жу [21], BLS-протоколе короткой цифровой подписи [16], протоколе шифрования [15] и цифровой подписи [20] с личным ключом и в других подобных протоколах.

Литература

[1] Болотов А. А., Гашков С. Б., Фролов А. Б., Часовских А. А. Программные и схемные методы умножения многочленов для эллиптической криптографии // Изв. РАН. Теория и системы управления. — 2000. — № 5. — С. 66—75.

[2] Болотов А. А., Гашков С. Б., Фролов А. Б., Часовских А. А. О методах имплементации арифметических операций в криптографических системах // Изв. РАН. Теория и системы управления. — 2002. — № 1. — С. 86—96.

[3] Болотов А. А., Гашков С. Б., Фролов А. Б., Часовских А. А. Элементарное введение в эллиптическую криптографию. Алгебраические и алгоритмические основы. — М.: Комкнига. 2006.

[4] Болотов А. А., Гашков С. Б., Фролов А. Б., Часовских А. А. Элементарное введение в эллиптическую криптографию. Протоколы криптографии на эллиптических кривых. — М.: КомКнига, 2006.

[5] Жебет С. Ю. Сравнительный анализ алгоритмов умножения бинарных многочленов в полиномиальном базисе // Вестн. МЭИ. — 2006. — № 6. — С. 52—61.

[6] Карацуба А. А., Офман Ю. П. Умножение многозначных чисел на автоматах // ДАН СССР. — 1962. — Т. 145, № 2. — С. 293—294.

[7] Тоом А. Л. О сложности схемы из функциональных элементов, реализующей умножение целых чисел // ДАН СССР. — 1963. — Т. 150. — С. 496—498.

[8] Ноден П., Китте К. Алгебраическая алгоритмика. — М.: Мир, 1999.

[9] Bailey D. V., Paar C. Efficient arithmetic in finite field extensions with application in elliptic curve cryptography // J. Cryptology. — 2001. — Vol. 14, no. 3. — P. 156—173.

[10] Barreto P. S. L. M., Kim H. Y., Lynn B., Scott M. Efficient algorithms for pairing-based cryptosystems // Advances in Cryptology — CRYPTO 2002. — Berlin: Springer, 2002. — (Lect. Notes Comput. Sci.; Vol. 2442). — P. 354—369.

[11] Barreto P. S. M. L., Galbraith S., Ó hÉigeartaigh C., Scott M. Efficient pairing computation on supersingular Abelian varieties. — Cryptology ePrint Archive, Report 2004/375. — http://eprint.iacr.org/2004/375.

[12] Beuchat J.-L., Shiraze M., Takagi T., Okamoto E. An algorithm for the η_T-pairing calculation in characteristic three and its hardware implementation. — Cryptology ePrint Archive, Report 2006/327. — http://eprint.iacr.org/2006/327.

[13] Blake I., Seroussi G., Smart N. Elliptic Curves in Cryptography. — Cambridge: Cambridge Univ. Press, 1999.

[14] Blake I., Seroussi G., Smart N. Advances in Elliptic Curve Cryptograhy. — Cambridge: Cambridge Univ. Press, 2005.

[15] Boneh D., Franklin M. Identity-based encryption from the Weil pairing // Advances in Cryptology — Crypto 2001. — Berlin: Springer, 2001. — (Lect. Notes Comput. Sci.; Vol. 2139). — P. 213—229.

[16] Boneh D., Lynn B., Shacham H. Short signatures from the Weil pairing // Asiacrypt-2001. — Berlin: Springer, 2002. — (Lect. Notes Comput. Sci.; Vol. 2248). — P. 514—532.

[17] Duursma I., Lee H.-S. Tate pairing implementation for hyperelliptic curves $y^2 = x^p - x + d$ // Asiacrypt-2003. — Berlin: Springer, 2003. — (Lect. Notes Comput. Sci.; Vol. 2894). — P. 111—123.

[18] Duursma I., Lee H.-S. Tate pairing implementation for tripartite key agreement. — Cryptology ePrint Archive, Report 2003/053. — http://eprint.iacr.org/2003/053.

[19] Granger R., Page D., Stam M. Hardware and software normal basis arithmetic for pairing based cryptography in characteristic three // IEEE Trans. Comput. — 2005. — Vol. 54, no. 7. — P. 852—860.

[20] Hess F. Exponent group signature schemes and efficient identity based signature schemes based on pairings. — Cryptology ePrint Archive, Report 2002/012. — http://eprint.iacr.org/2002/012/.

[21] Joux A. A one-round protocol for tripartite Diffie—Hellman // Algorithm Number Theory Symposium — ANTS IV. — Berlin: Springer, 2000. — (Lect. Notes Comput. Sci.; Vol. 1838). — P. 385—394.

[22] Kerins T., Marname W. P., Popovici E. M., Barreto P. S. L. M. Efficient hardware for Tate pairing calculation in characteristic three // Cryptographic Hardware and Embedded Systems — CHES 2005. — Berlin: Springer, 2005. — (Lect. Notes Comput. Sci.; Vol. 3659). — P. 412—426.

[23] Kwon S. Efficient Tate pairing computation for supersingular elliptic curves over binary fields. — Cryptology ePrint Archive, Report 2004/303. — http://eprint.iacr.org/2004/303.

[24] Lee E., Lee H.-S., Lee Y. Fast computation of Tate pairing on general divisors for hyperelliptic curves of genus 3. — Cryptology ePrint Archive, Report 2006/125. — http://eprint.iacr.org/2006/125.

[25] Page D., Smart N. P. Hardware implementation of finite fields of characteristic three // Cryptographic Hardware and Embedded Systems — CHES 2002. — Berlin: Springer, 2003. — (Lect. Notes Comput. Sci.; Vol. 2523). — P. 283—295.

[26] Scott M., Barreto P. S. M. L. Compressed pairing // Advances in Cryptology — CRYPTO 2004. — Berlin: Springer, 2004. — (Lect. Notes Comput. Sci.; Vol. 3152). — P. 140—156.

[27] Shiraze M., Takagi T., Okamoto E. Some efficient algorithms for the final exponentiation of an η_T-pairing. — Cryptology ePrint Archive, Report 2006/431. — `http://eprint.iacr.org/2006/431`.