УДК 681.3.06:519.248.681

С.В. Дуденко, С.В. Алексеев, В.В. Добровольский

Харьковский университет Воздушных Сил им. И. Кожедуба

АЛГОРИТМ БЫСТРОГО ПРЕОБРАЗОВАНИЯ ФУРЬЕ ГУДА-ТОМАСА В КОНЕЧНЫХ ПОЛЯХ ДЛЯ УСЕЧЕННОГО ВЕКТОРА

Для классического преобразования Фурье в конечных полях существуют быстрые алгоритмы, среди которых выделяют алгоритмы Кули-Тьюки и Гуда-Томаса, позволяющие значительно снизить вычислительную сложность за счет переиндексации точек векторов. Показано, что для усеченного преобразования Фурье в остаточных классах применим только алгоритм Кули-Тьюки, так как свойство четности длин векторов для усеченного преобразования Фурье и требование взаимной простоты множителей Гуда-Томаса не позволяют использовать его алгоритмическое решение.

Ключевые слова: *преобразование Фурье, быстрое преобразование Фурье, алгоритм.*

Введение

Постановка задачи. В теории помехоустойчивого кодирования, в системах обработки информации, а также при разработке криптографических протоколов широко используют различные теоретико-числовые преобразования.

Наиболее известны преобразование Фурье и усеченное преобразование Фурье в конечных полях. Они, не перекрывая друг друга, формируют простран-

ство длин векторов, которые могут быть использованы при расчетах.

Анализ исследований и публикаций. Для классического преобразования Фурье в конечных полях существуют быстрые алгоритмы (реализующие так называемое быстрое преобразование Фурье (БПФ)), среди которых выделяют алгоритмы Кули-Тьюки и Гуда-Томаса [1], позволяющие значительно снизить вычислительную сложность преобразований за счет переиндексации точек векторов.

Цель статьи – получение аналитических выражений БПФ-алгоритма Гуда-Томаса для усеченного преобразования Фурье.

Основной материал

1. Теоретико-числовые преобразования.

1.1. Преобразование Фурье над полем Галуа.

Определение [1]. Пусть $v = \{v_i, i = 0, \ldots, n-1\}$ – вектор над полем Галуа GF (q), где n делит q^m-1 при некотором m; w – элемент порядка n в поле GF (q^m). Тогда преобразование Фурье вектора v в поле Галуа

$$V_j=\sum_{i=0}^{n-1} w^{ij}\cdot v_i \;;\; v_i=\left(1/n\right)\sum_{j=0}^{n-1} w^{-ij}\cdot V_j, \quad (1)$$

где n интерпретируется как число поля, т.е. по модулю p; w – элемент порядка n в поле GF (q).

В качестве длины преобразования Фурье можно выбрать произвольный делитель числа q^m-1. Например, в поле GF (2^8) преобразование Фурье существует для n = 3, 5, 15, 17, 51, 85, 255. Для большинства применений таких длин векторов оказывается достаточно.

1.2. Усеченное преобразование Фурье над полем Галуа. Доказательство теоремы о существовании усеченного преобразования Фурье в конечных полях расширило пространство допустимых длин векторов.

Теорема [2]. Над полем GF (q) характеристики p существует преобразование вида:

$$V_j=\sum_{i=1}^{n} w^{ij}\cdot v_i;\; v_i=\left(\frac{1}{n \bmod p}\right)\sum_{j=1}^{n}\left(w^{-ij}\oplus L\right)\cdot V_j, \quad (2)$$

где w – элемент порядка n+1 в поле GF (q); $\oplus$ – означает операцию сложения в поле; L = –1.

Отметим, что значение w для классического и усеченного преобразований будет различным. Так, например, для четырехточечного усеченного преобразования необходимо проводить расчеты через элемент пятого порядка, а не четвертого.

Пример. В поле GF (2^8) преобразование (2) существует для векторов длины 2, 4, 14, 16, 50, 84, 254. Таким образом, применяя классическое преобразование Фурье и усеченное в поле GF (2^8) мы можем обрабатывать вектора длины 2, 3, 4, 5, 14, 15, 16, 17, 50, 51, 84, 85, 254, 255, что существенно расширяет область использования теоретико-числовых преобразований.

2. Алгоритмы быстрого преобразования Фурье.

Преобразование Фурье (1) требует порядка n^2 умножений и n^2 сложений, что при обработке векторов с большим значением n требует значительных временных затрат. С целью минимизации вычислительной сложности преобразований были разработаны алгоритмы быстрого преобразования Фурье: БПФ-алгоритм Кули-Тьюки и алгоритм Гуда-Томаса [1]. С вычислительной точки зрения использование указанных алгоритмов приводит к существенно более эффективной форме, хотя понять такую структуру труднее. Так первый алгоритм требует приблизительно n·(n' + n") + n умножений, а второй – около n·(n' + n").

В алгоритме Гуда-Томаса форма переиндексации линейного массива в двумерную $(n'\times n'')$-таблицу отличается от формы переиндексации в алгоритме Кули-Тьюки, он немного сложнее в принципе, но и немного проще в вычислительном отношении. В нем n' и n'' предполагаются взаимно простыми, и в основе отображения лежит китайская теорема об остатках. Алгоритм можно наглядно представить в виде отображения одномерной таблицы в двумерную (рис. 1) для 15-точечного вектора (n = 15, n' = 3, n"= 5). Упорядочение (рис. 1) начинается с левого верхнего угла таблицы, и компоненты записываются вдоль продолжения диагонали. Так как число строк и число столбцов взаимно просты, то продолженная диагональ пройдет через все элементы таблицы. После выполнения двумерного преобразования компоненты спектра выписываются в таблице в другом порядке.

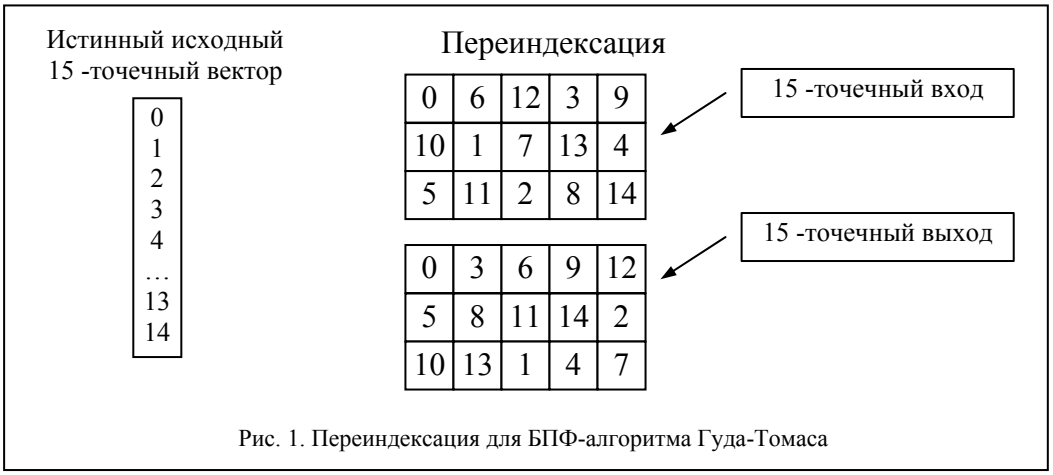

Рис. 1. Переиндексация для БПФ-алгоритма Гуда-Томаса

Входные индексы определяются своими вычетами по правилу $i' = i \pmod{n'}$, $i'' = i \pmod{n''}$, задающему отображение входного индекса i в пару индексов (i', i''), лежащих на продолженной диагонали двумерной таблицы. Согласно китайской теореме об остатках, существуют целые числа N' и N'', такие что $N'n' + N''n'' = 1$. Это позволяет однозначно вычислить исходный индекс i по паре (i', i''): $i = i'N''n'' + i''N'n' \pmod{n}$. Выходные индексы упорядочиваются несколько иным способом.

Определим индексы: $j' = N''j \pmod{n'}$, $j'' = N'j \pmod{n''}$. В эквивалентном виде их можно записать как $j'' = ((N' \bmod n'')\, j \bmod n'')$ и $j' = ((N'' \bmod n')\, j \bmod n')$. Тогда выходной индекс j можно вычислить как $j = n''j' + n'j'' \pmod{n}$.

При такой переиндексации формула $V_l = \sum_{i=0}^{n-1} \alpha^{ij} \upsilon_i$ принимает вид

$$V_{n''j'+n'j''} = \sum_{i''=0}^{n''-1} \sum_{i'=0}^{n'-1} \alpha^{(i'N''n''+i''N'')(n''j'+n'j'')} \upsilon_{i'N''n''+i''N'n'}.$$

Так как порядок элемента α равен $n'n''$, то члены, содержащие $n'n''$, можно отбросить. Используя выписанные соответствия для индексов, рассматривают входной и выходной векторы как двумерные таблицы. Тогда

$$V_{j',j''} = \sum_{i'=0}^{n'-1} \sum_{i''=0}^{n''-1} \alpha^{N''(n'')^2 i'j'} \alpha^{N'(n')^2 i''j''} \upsilon_{i',i''} = \sum_{i'=0}^{n'-1} \sum_{i''=0}^{n''-1} \beta^{i'j'} \gamma^{i''j''} \upsilon_{i',i''},$$

где $\beta = \alpha^{N''(n'')^2}$ – элемент порядка n', а $\gamma = \alpha^{N'(n')^2}$ – элемент порядка n''. Полученная формула представляет собой двумерное $(n' \times n'')$-точечное преобразование Фурье, для вычисления которого требуется примерно $n(n' + n'')$ умножений и примерно столько же сложений. Если длина преобразования по столбцам или по строкам является составным числом, то можно провести дальнейшее упрощение, снова применяя быстрое преобразование Фурье. Поступая таким образом, можно преобразование длины n, распадающейся на взаимно простые множители n_i, привести к форме, для вычисления которой требуется порядка $n \sum_i n_i$ умножений и сложений.

Используя рассмотренный алгоритм, мы можем в поле GF (2^8) обрабатывать вектора длины 15, 51, 85 и 255, которые могут быть разложены на составляющие, а остальные длины 3, 5, 17 соответствуют простым числам и не подходят для алгоритма. Однако существуют также длины 4, 16, 50, 84, 254, которые могут быть разложены на составляющие и соответствуют усеченному преобразованию Фурье. Можно заметить, что усеченное преобразование оперирует четными длинами векторов, что априорно определяет неприменимость подхода Гуда-Томаса к усеченному вектору.

Выводы

Для выполнения классического преобразования Фурье в остаточных классах можно выбрать как алгоритм Кули-Тьюки, так и алгоритм Гуда-Томаса. Иногда возможно даже использование обоих алгоритмов одновременно.

Для усеченного преобразования Фурье в остаточных классах можно выбрать только алгоритм Кули-Тьюки, так как свойство четности длин векторов для усеченного преобразования Фурье и требование взаимной простоты множителей Гуда-Томаса не позволяют использовать его алгоритмическое решение.

Список литературы

1. Блейхут Р. Теория и практика кодов, контролирующих ошибки / Р. Блейхут. – М.: Мир, 1986. – 576 с.

2. Долгов В.И. Построение нелинейных систем на основе усеченного преобразования Фурье в конечных полях / В.И. Долгов, И.В. Рубан, С.В. Дуденко // Радиотехника: Всеукр. межвед. науч.-техн. сб. – 2003. – № 134. – С. 121-132.

3. Дуденко С.В. БПФ алгоритм Кули-Тьюки для усеченного преобразования Фурье в конечных полях / С.В. Дуденко, В.А. Пудов // Зб. наук. праць ІПМЕ ім. Г.Є. Пухова. – К., 2004. – Вип. 25. – С. 30-34.

Поступила в редколлегию 16..01.2009

Рецензент: д-р физ.-мат. наук, проф. С.В. Смеляков, Харьковский университет Воздушных Сил им. И. Кожедуба, Харьков.

АЛГОРИТМ ШВИДКОГО ПЕРЕТВОРЕННЯ ФУРЬЕ ГУДА-ТОМАСА В КІНЦЕВИХ ПОЛЯХ ДЛЯ УСІЧЕНОГО ВЕКТОРА

С.В. Дуденко, С.В. Алексєєв, В.В. Добровольський

Для класичного перетворення Фурье в кінцевих полях існують швидкі алгоритми, серед яких виділяють алгоритми Кулі-Тьюки і Гуда-Томаса, які дозволяють значно зменшити обчислювальну складність за рахунок переіндексації точок векторів. Показано, що для усіченого перетворення Фурье в залишкових класах можна застосувати тільки алгоритм Кулі-Тьюки, оскільки властивість парності довжин векторів для усіченого перетворення Фурье і вимога взаємної простоти множників Гуда-Томаса не дозволяють використовувати його алгоритмічне рішення.

Ключові слова: *перетворення Фурье, швидке перетворення Фурье, алгоритм.*

FAST FOURIER TRANSFORM OF GOOD-THOMAS ALGORITHM FOR TRUNCATED VECTOR IN THE EVENTUAL FIELDS

S.V. Dudenko, S.V. Alekseev, V.V. Dobrovolsky

For classic Fourier transform in the eventual fields there are fast algorithms, among which select the Cooley-Tukey and Good-Thomas algorithms, allowing it is considerably to reduce calculable complication due to indexation points of vectors. It is shown that for the truncated Fourier transformation in remaining classes we will apply the Cooley-Tukey algorithm only, because property of lengths evenness of vectors for the truncated Fourier transformation and requirement of mutual simplicity of Good-Thomas multipliers does not allow to use his algorithmic decision.

Keywords: *Fourier transform, fast Fourier transform, algorithm.*