TECHNICAL SCIENCES

COMPARATIVE ANALYSIS OF THE APPLICATION OF FAST FOURIER TRANSFORM AND NUMBER-THEORETICAL TRANSFORM IN FULL HOMOMORPHOUS ENCRYPTION SCHEMES

Retivykh A.
lecturer at the Cybersecurity Competence Center, Irkutsk National ResearchTechnical University, Irkutsk, Russia
https://orcid.org/0000-0001-8362-8247
Retivykh V.
lecturer at the Cybersecurity Competence Center, Irkutsk National ResearchTechnical University, Irkutsk, Russia
https://orcid.org/0009-0007-7423-5362
Marinov A.
PhD in Economic Sciences, associate Professor at the Cybersecurity Competence Center, Irkutsk National ResearchTechnical University, Irkutsk, Russia
https://orcid.org/0000–0003–2238–2751

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПРИМЕНЕНИЯ БЫСТРОГО ПРЕОБРАЗОВАНИЯ ФУРЬЕ И ТЕОРЕТИКО-ЧИСЛОВОГО ПРЕОБРАЗОВАНИЯ В СХЕМАХ ПОЛНОГО ГОМОМОРФНОГО ШИФРОВАНИЯ

Ретивых А.В.
Преподаватель центра компетенций по кибербезопасности, ФГБОУ ВО «Иркутский национальный исследовательский технический университет», г. Иркутск, Россия.
https://orcid.org/0000-0001-8362-8247
Ретивых В.В.
Преподаватель центра компетенций по кибербезопасности, ФГБОУ ВО «Иркутский национальный исследовательский технический университет», г. Иркутск, Россия.
https://orcid.org/0009-0007-7423-5362
Маринов А.А.
кандидат экономических наук, доцент центра компетенций по кибербезопасности, ФГБОУ ВО «Иркутский национальный исследовательский технический университет», г. Иркутск, Россия.
https://orcid.org/0000–0003–2238–2751
https://doi.org/10.5281/zenodo.19050762

Abstract

This paper provides an in-depth analysis of two fundamental algorithms – the Fast Fourier Transform (FFT) and the Number Theoretic Transform (NTT) – in the context of their application in Fully Homomorphic Encryption (FHE) systems. It is shown that, despite sharing a common algorithmic basis (the divide-and-conquer strategy) and identical asymptotic complexity of $O(nlog(n))$, the choice between them is dictated by the error model of a specific cryptosystem. A detailed mathematical derivation of NTT for the ring $Z_q[x]/(x^n + 1)$, which underlies the BGV, BFV, and CKKS schemes, is presented. A comparative analysis of accuracy, performance, and hardware implementability is performed. It is established that for schemes with exact integer arithmetic (BGV, BFV), NTT is the only possible choice, while in the approximate computation scheme (CKKS), FFT is used at the data encoding stage, and NTT is used for cryptographic operations.

Аннотация

В работе проводится глубокий анализ двух фундаментальных алгоритмов – Быстрого преобразования Фурье (FFT) и теоретико-числового преобразования (NTT) – в контексте их применения в системах полного гомоморфного шифрования (FHE). Показано, что, несмотря на общую алгоритмическую основу и идентичную асимптотическую сложность $O(nlog(n))$, выбор между ними диктуется моделью ошибок конкретной криптосистемы. Приводится детальный математический вывод NTT для кольца $Z_q[x]/(x^n + 1)$, лежащего в основе схем BGV, BFV и CKKS. Выполнен сравнительный анализ точности, производительности и аппаратной реализуемости. Установлено, что для схем с точной целочисленной арифметикой (BGV, BFV) NTT является единственно возможным выбором, в то время как в схеме приближенных вычислений (CKKS) FFT применяется на этапе кодирования данных, а NTT – для криптографических операций.

Keywords: Number Theoretic Transform (NTT), Fast Fourier Transform (FFT), Fully Homomorphic Encryption (FHE), Lattice-based cryptography, BGV scheme, BFV scheme, CKKS scheme.

Ключевые слова: Теоретико-числовое преобразование (NTT), Быстрое преобразование Фурье (FFT), Полное гомоморфное шифрование (FHE), Криптография на решетках, Схема BGV, Схема BFV, Схема CKKS.

ВВЕДЕНИЕ

Современная криптография переживает переход к постквантовым алгоритмам, среди которых лидирующее положение занимают протоколы, основанные на сложности решения задач в решетках. Одной из наиболее мощных концепций в этом направлении является полное гомоморфное шифрование (Fully Homomorphic Encryption, FHE), позволяющее выполнять произвольные вычисления над зашифрованными данными без их расшифровки [1]. Математической основой большинства эффективных схем FHE (BGV [2], BFV [3], CKKS [4]) являются кольца многочленов вида $R_q = Z_q[x]/(x^n + 1)$, где n – степень двойки (1024, 2048, 4096), а q – большой модуль. Ключевая операция – умножение многочленов в этом кольце. Наивное умножение имеет сложность $O(n^2)$, что является неприемлемым для практического использования FHE.

Задача ускорения умножения многочленов решается с помощью алгоритмов быстрого преобразования Фурье (FFT) [5] и его аналога в конечных полях – теоретико-числового преобразования (NTT) [6]. Оба алгоритма, используя теорему о свертке, позволяют достичь асимптотической сложности $O(nlog(n))$. Однако принципиальные различия в арифметике, комплексные числа с плавающей точкой против целочисленной модульной арифметики, определяют их применимость в FHE.

Целью данной работы является проведение сравнительного анализа NTT и FFT с акцентом на их математические основы и применение в FHE-схемах. Для достижения цели ставятся следующие задачи: 1) дать детальное математическое описание NTT для кольца $Z_q[x]/(x^n + 1)$; 2) провести сравнительный анализ алгоритмов по ключевым параметрам (точность, производительность, гибкость); 3) определить области оптимального использования каждого алгоритма в схемах BGV, BFV и CKKS.

1. Математические основы теоретико-числового преобразования (NTT) в контексте FHE

Теоретико-числовое преобразование является дискретным аналогом преобразования Фурье, перенесенным в конечное поле Z_q.

1.1.Определение и условия существования NTT

Пусть задан вектор a = (a_0, a_1, ..., $a_{(n-1)}$) с элементами из Z_q, где q – простое число. Дискретное преобразование Фурье над полем Z_q (NTT) вектора a есть вектор A = (A_0, A_1, ..., $A_{(n-1)}$), компоненты которого вычисляются по формуле:

$$A_k = \sum_{j=0}^{n-1} a_j \cdot \omega^{jk} \, (mod q), k = 0, 1, \ldots, n-1, (1)$$

где ω – примитивный корень n-й степени из единицы в поле Z_q. Для существования преобразования необходимо, чтобы n делило порядок мультипликативной группы поля, т.е. n | (q-1). Обратное преобразование (INTT) имеет вид:

$$a_j = n^{-1} \cdot \sum_{k=0}^{n-1} A_k \cdot \omega^{-jk} \, (mod \, q), j = 0, 1, \ldots, n-1, (2)$$

где n^{-1} – мультипликативно обратный элемент к n по модулю q.

1.2.Негациклическое NTT для кольца $R_q = Z_q[x]/(x^n + 1)$

Для кольца $x^n + 1$, используемого в FHE, применяется модифицированное (негациклическое) NTT. В этом случае необходим примитивный корень ψ степени $2n$ из единицы, для которого $\psi^n \equiv -1 \, (mod \, q)$. Прямое преобразование вычисляется как:

$$A_k = \sum_{j=0}^{n-1} a_j \cdot \psi^{(2k+1)j} \, (mod \, q), k = 0, 1, \ldots, n-1. (3)$$

Это преобразование является изоморфизмом между кольцом R_q и векторами длины n, при котором умножению многочленов в кольце соответствует покомпонентное умножение их образов. Эффективное вычисление (3) осуществляется с помощью алгоритма Кули-Тьюки за время $O(n \log(n))$.

2. Сравнительный анализ FFT и NTT для FHE

Выбор между FFT и NTT определяется не только производительностью, но и фундаментальными требованиями к точности вычислений.

2.1. Сравнение по ключевым параметрам

Сравнительный анализ по основным критериям представлен в табл. 1.

Таблица 1.

Сравнение FFT и NTT

Критерий	FFT	NTT
Область определения	Поле комплексных чисел C.	Конечное поле (поле Галуа) GF(q) или кольцо вычетов Z_q, где q – простое число.
Базовый элемент (корень из единицы)	Комплексный корень N-й степени из единицы: $\omega_N = e^{(-2\pi i/N)}$. Является трансцендентным элементом.	Целочисленный корень n-й степени из единицы по модулю q: $\omega \in Z_q$, такой что $\omega^n \equiv 1\,(mod\ q)$ и $\omega^k \neq 1$ для $0 < k < n$.
Условие существования преобразования длины n	Отсутствуют алгебраические ограничения, n может быть любым целым числом (для эффективности обычно степень 2).	Строгое алгебраическое условие: $n \mid (q-1)$. Для негациклической свертки (кольцо $x^n + 1$): *2n / (q-1).* Модуль q должен быть выбран специальным образом.
Точность вычислений	**Приближенная.** Используется арифметика с плавающей точкой (FP32, FP64). Присутствуют ошибки округления, накапливающиеся с глубиной преобразования ($O(\varepsilon \log(n))$.	**Абсолютная (точная).** Используется целочисленная модульная арифметика. Ошибки округления отсутствуют. Результат детерминирован и обратим.
Тип арифметических операций	Операции над числами с плавающей точкой. Сложение, умножение комплексных чисел. Требуют наличия FPU (Floating Point Unit).	Операции над целыми числами по модулю q. Сложение, вычитание, умножение с приведением по модулю. Требуют наличия ALU и умножителей.
Асимптотическая сложность	$O(Nlog(N))$ для длины N (основание 2).	$O(Nlog(N))$ для длины N (основание 2).
Реализация алгоритма «бабочка»	Вычисляется по формуле: $X = A + \omega \cdot B;\ Y = A - \omega \cdot B$, где ω – комплексный поворотный коэффициент.	Вычисляется по формуле аналогичной FFT, но все операции выполняются по модулю q. $A_k = A_{even} + \omega^k \cdot A_{odd}\ (mod\ q)$.
Стоимость базовой операции («бабочки»)	Высокая. Требует 2 комплексных умножения и 2 комплексных сложения (что эквивалентно множеству операций с плавающей точкой).	Средняя. Требует 2 целочисленных умножения, 2 сложения и 2 операции приведения по модулю (редукции).
Управление динамическим диапазоном	Автоматическое. Числа с плавающей точкой масштабируют результат. Требуется контроль переполнения экспоненты.	Критически важное. Промежуточные результаты могут расти. Требуется либо использование модулей специального вида
Гибкость выбора параметров	Высокая. Длина преобразования N может быть любой, диктуется лишь задачей и кратна степени двойки для максимальной эффективности.	Низкая. Длина преобразования n жестко связана с модулем q. Выбор параметров безопасности FHE (n, q) подчинен условию существования NTT.
Реализация на CPU (x86/ARM)	Высоко оптимизирована. Использует SIMD-инструкции (SSE, AVX, AVX-512, NEON) для векторизации операций с плавающей точкой. Производительность очень высока.	Зависит от оптимизации модульной арифметики. Используются инструкции для целочисленной арифметики (ADCX, ADOX, MULX). При хорошей оптимизации производительность сопоставима с FFT.
Реализация на GPU (CUDA/OpenCL)	Отлично распараллеливается. Существуют готовые библиотеки (cuFFT). Широко используется в научных расчетах и машинном обучении.	Хорошо распараллеливается, но требует оптимизации модульной редукции. Используется в специализированных библиотеках для FHE.
Реализация на ПЛИС (FPGA)	Сложная и ресурсоемкая. Требует реализации блоков операций с плавающей точкой (DSP-блоки в режиме FP), что потребляет много логических ресурсов и энергии.	Высокоэффективная. Целочисленные модульные операции отлично ложатся на аппаратную архитектуру ПЛИС, позволяя достичь высокой тактовой частоты и пропускной способности при низком энергопотреблении.

Реализация на специализированных микросхемах (ASIC)	Экономически нецелесообразна для массового применения из-за сложности FP-блоков, за исключением специализированных процессоров (DSP).	Крайне эффективна. Позволяет создавать специализированные ускорители для FHE с высокой производительностью и энергоэффективностью. Активная область исследований.
Чувствительность к атакам по побочным каналам (Side-Channel Attacks)	Низкая. Время выполнения и энергопотребление могут зависеть от данных из-за особенностей FPU, но это редко используется.	Высокая. Время выполнения модульного умножения и ветвления в алгоритме могут зависеть от секретных ключей.

2.2. Применение в схемах полного гомоморфного шифрования

Различия в природе арифметики приводят к четкому разделению областей применения алгоритмов:

1. Схемы BGV и BFV. Эти схемы работают с целочисленными коэффициентами и требуют точного управления шумом. Ошибки округления FFT неприемлемы. Умножение многочленов реализуется исключительно через NTT. Стандартный пайплайн включает в себя: прямое NTT (3), покомпонентное умножение и обратное INTT. Это де-факто стандарт для библиотек Microsoft SEAL [7] и HElib [8].

2. Схема CKKS. Эта схема предназначена для вычислений над числами с плавающей точкой. Она использует гибридный подход [4]:

- на этапе кодирования вектора комплексных чисел в многочлен применяется IFFT;
- сами гомоморфные операции (умножение шифртекстов) выполняются в кольце R_q и требуют точной арифметики, поэтому здесь используется NTT;
- на этапе декодирования результата применяется FFT.

Заключение

В работе проведен сравнительный анализ применения алгоритмов FFT и NTT в современных схемах полного гомоморфного шифрования. Показано, что, несмотря на общую алгоритмическую основу, их применение не является взаимозаменяемым, а диктуется математической моделью криптосистемы. Главные выводы по данной работе:

1. Для схем BGV и BFV, основанных на точной модульной арифметике, применение FFT невозможно из-за неконтролируемых ошибок округления. NTT является критически важным компонентом, обеспечивающим практическую производительность умножения многочленов.

2. В схеме CKKS реализован гибридный подход: FFT применяется для кодирования и декодирования данных в комплексной области, в то время как криптографические операции по-прежнему выполняются с использованием NTT для обеспечения корректности вычислений в кольце вычетов.

3. Выбор алгоритма также влияет на аппаратную реализацию: NTT значительно более эффективен при реализации на ПЛИС (FPGA) благодаря простоте целочисленных модульных операций.

Таким образом, понимание математических основ и ограничений NTT и FFT является необходимым условием для разработки эффективных и корректных реализаций систем полного гомоморфного шифрования.

Список литературы:

1. Gentry C. A fully homomorphic encryption scheme. – Stanford: Stanford University, 2009. – 209 p.

2. Brakerski Z., Gentry C., Vaikuntanathan V. (Leveled) fully homomorphic encryption without bootstrapping // ACM Transactions on Computation Theory. – 2014. – Vol. 6, no. 3. – с. 1–36.

3. Fan J., Vercauteren F. Somewhat practical fully homomorphic encryption // IACR Cryptol. ePrint Arch. – 2012. – Vol. 2012. – с. 144.

4. Cheon J. H. et al. Homomorphic encryption for arithmetic of approximate numbers // Advances in Cryptology–ASIACRYPT 2017. – Cham: Springer International Publishing, 2017. – с. 409–437.

5. Cooley J. W., Tukey J. W. An algorithm for the machine calculation of complex Fourier series // Mathematics of computation. – 1965. – Vol. 19, no. 90. – с. 297–301.

6. McClellan J. H., Rader C. M. Number theory in digital signal processing. – Prentice-Hall, Inc., 1979. – с. 284.

7. Chen H., Laine K., Player R. Simple encrypted arithmetic library-SEAL v2. 1 // Financial Cryptography and Data Security: FC 2017 International Workshops. – Cham: Springer International Publishing, 2017. – с. 3–18.

8. Halevi S., Shoup V. Algorithms in HElib // Advances in Cryptology–CRYPTO 2014. – Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. – с. 554–571.