**Факторизация целых чисел методом квадратичного решета.**

Гамов П.А.

Научный руководитель —Ухов П.А.

МАИ, Москва

Факторизацией натурального числа называется разложение этого числа в произведение простых сомножителей. Эта задача имеет большую вычислительную сложность. Один из самых популярных методов криптографии с открытым ключом, метод RSA, основан на трудоемкости задачи факторизации длинных целых чисел. [1]

Первым серьёзным прорывом было квадратичное решето, quadratic sieve (QS). Это относительно простой алгоритм факторизации, предложенный Carl Pomerance в 1981 г., который может разлагать на множители числа до 110 десятичных разрядов или около того и для таких чисел остается лучшим. [2]

Решение задачи факторизации имеет прикладное значение, так как ставит под сомнение безопасность шифровальных алгоритмов, на которых строится современная криптография. Метод квадратичного решета является вторым по скорости факторизации чисел, его обгоняет только метод общего решета числового поля. Принцип работы алгоритма квадратичного решета основана на идее факторизации Ферма – поиске двух чисел, которые являются полными квадратами по модулю искомого числа. Нахождение таких чисел ставит перед нами необходимость долгого поиска необходимой факторной базы, гладких чисел, решение достаточно большой разряженной матрицы в конечном числовом поле.

Работа состоит в исследовании и реализации метода квадратичного решета, сравнении его с другими существующими алгоритмами, подборе оптимальных начальных условий, необходимых для ускорения алгоритма. Так же в исследовании различных способов решения разряженных матриц, алгоритмов поиска больших простых чисел.

Понимание этого алгоритма является фундаментальным, он лежит в основе понимания более тяжелых алгоритмов, которые способны факторизовать числа, длинною более ста цифр.

Самым распространенным методом факторизации можно назвать P метод Полларда. Данный алгоритм основывается на алгоритме Флойда поиска длины цикла в последовательности и некоторых следствиях из парадокса дней рождения. Алгоритм наиболее эффективен при факторизации составных чисел с достаточно малыми множителями в разложении. Сложность алгоритма оценивается как O(N^{1/4}). Параллельность возможна, но не дает линейного выигрыша в скорости.

Таким образом данный метод эффективен на сравнительно малых числах с небольшими делителями и не имеет значительных возможностей для дальнейшего улучшения.

Рассматривая алгоритм квадратичного решета, метод его работы значительно усложняется, вводятся новые понятия, такие как гладкие числа, факторные базы, матрицы в конечных полях, зато появляется огромный простор для улучшения. Сам по себе алгоритм является субэкспоненциальным со сложностью О(exp(sqrt(log n log log n))).

Первый этап заключается в поиске факторной базы. Выбирается некоторое число B как верхняя граница поиска и используя решето Эратосфена находятся простые числа. Сложность такой операции считается О(B log log B). Использование улучшенного алгоритма решета Аткина еще больше понижает асимптотическую сложность, позволяя найти факторную базу быстрее.

Второй этап сводится к поиску гладких чисел, используя генерирующий многочлен на некотором конечном интервале. Данный этап можно разбить на подзадачи, так как регионы независимы, следовательно имея достаточно вычислительных машин, поиск ускоряется в несколько раз. Таких чисел нам потребуется несколько больше, чем размер факторной базы. Чем больше факторная база, тем чаще встречаются гладкие числа, так же верна и обратная зависимость, следовательно надо понимать, какие гиперпараметры подавать на вход программе, для достижения приемлемых пропорций количества и скорости.

Третий этап подразумевает составления и решения матричного уравнения, для нахождения всех возможных совокупностей чисел, которые будут давать полный квадрат, чтобы удовлетворять критерию Ферма. Для этого коэффициенты разложения гладких чисел можно перевести в поле нуля и единицы, просто взяв остаток от деления на два. Классический метод решения, с использованием метода Гаусса показывает кубическую сложность, и мною не было найдено каких-либо способов линейно ускорить данный процесс. Однако существует возможность получения решения, не имея полного необходимого списка гладких чисел и их разложений, написание агента, который действует по принципам реактивного программирования – будет пытаться найти решение заранее, и каждое последующее добавление нового гладкого числа в существующую матрицу будет запускать процесс пересчета, формируя новые линейные пары чисел. Так же так как мы храним только остатки деления чисел на два, можно привести элементы матрицы в формат битовых срезов, и использовать побитовые операции, сомнительное решение, но оно возможно.

Список использованных источников:

1. Ишмухаметов Ш. Т. Методы факторизации натуральных чисел: учебное пособие. — Казань: Казан. ун., 2011.
2. И. В. Агафонова Факторизация больших целых чисел и криптография.