Метод параллельных защищенных вычислений систем линейных уравнений с трех диагональной матрицей на основе полностью гомоморфного шифрования для рациональных чисел

Вишневский Артем Константинович[1]
Кренделев Сергей Федорович [2]

[1]к.т.н., ВА РВСН им. Петра Великого
[2] к.ф-м.н., доцент НГУ, руководитель лаборатории криптографии JatBrans

Москва, 2018

Содержание доклада

Что такое гомоморфное шифрование ?

Гомоморфное шифрование

Это шифрование, которое предназначено для выполнения математических преобразований над зашифрованными данными. Результат таких «защищенных» преобразований при расшифровании соответствует результату математических преобразований над незашифрованными данными.

Формальное описание гомоморфного шифрования

Пусть имеется два множества элементов $\mathbf{U}$ и $\mathbf{V}$, на которых заданы две арифметические операции $+^{\mathbf{U}}$, $\cdot^{\mathbf{U}}$ и $\odot^{\mathbf{V}}$,$\otimes^{\mathbf{V}}$, требуется найти такое отображение $\varphi : \mathbf{U} \to \mathbf{V}$, чтобы выполнялось свойство гомоморфности:

$$\varphi(x \cdot^{\mathbf{U}} y) = \varphi(x) \otimes^{\mathbf{V}} \varphi(y),$$
$$\varphi(x +^{\mathbf{U}} y) = \varphi(x) \oplus^{\mathbf{V}} \varphi(y),$$

где φ имеет вид односторонней функции.

Какие разработки ведутся в данной области?

Основными инициаторами научных разработок являются

DARPA – агентство передовых оборонных исследовательских проектов министерства обороны США,
IARPA – Агентство по перспективным исследованиям разведывательного ведомства, национальная разведка США.

Направления исследований

1. Обработка зашифрованных персональных данных без расшифрования. Разделение полномочий для обработки данных;
2. Защита от атаки по побочному каналу (side channel attack); Основное направление – реализация стандартных методов шифрования (AES) с помощью полностью гомоморфного шифрования;
3. Работа на недоверенном железе. Использование в интернете вещей. Применение как технологии двойного назначения;
4. Разработка операционной системы на базе гомоморфного шифрования Gentry для защиты от кибератак. Делать будет VMWare для МО США;
5. IBM проект Watson;
6. Microsoft Azure – Cloud Platform.

Критика существующих схем гомоморфного шифрования

Частично гомоморфное шифрование

определена одна операция (сложения, либо умножения). Криптосхемы: RSA (умножение), Эль-Гамаля (умножение), Гольдвассера-Микали (умножение), Пэйе (сложение), Бенало (сложение). Данные схемы имеют теоретическую ценность и используются ограниченно – секретные голосования, аукционы.

Полностью гомоморфное шифрование

определены две операции – сложения и умножения. Криптосхема Gentry (2009). Определены две операции – сложения и умножения в $\mathbb{Z}_2$. Не эффективна при реализации реальных вычислений с целыми и рациональными числами.

Проблема

Существующие методы гомоморфного шифрования не позволяют использовать преимущества облачных вычислений, так как пока неизвестно как использовать параллельные вычисления, как работать с рациональными числами при решении реальных прикладных задач (решение дифференциальных уравнений, решение линейных уравнений и т. п.), как правильно округлять и находить ошибки в вычислениях.

Построение полностью гомоморфного шифрования

Известно что любое целое число $m \in \mathbb{Z}$ можно представить в виде скалярного произведения векторов:

$$m = \mathbf{vx}, \tag{1}$$

где $\mathbf{v} = [v_1 \ v_2 \ \dots \ v_n]$, $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_n]$. Более того, любое целое число можно представить в виде скалярного произведения векторов, где один из векторов, например $\mathbf{x}$, будет зафиксирован для каждого целого числа:

$$m_1 = \mathbf{v_1 x}, \ m_2 = \mathbf{v_2 x}, \dots, \ m_k = \mathbf{v_k x}, \dots \tag{2}$$

это возможно, если хотя бы две компоненты вектора $\mathbf{x}$ являются взаимно простыми числами, то есть уравнение (1) всегда разрешимо в целых числах. Отсюда имеем отображение:

$$\varphi_{\mathbf{x}} : \mathbb{Z} \to \mathbb{Z}^n. \tag{3}$$

Далее $\mathbf{x}$ будем называть секретным ключом, $\mathbf{v}$ будем называть открытым ключом.

Стойкость открытых ключей

Преположим, что злоумышленник не знает реальных значений $m_1, m_2, \ldots, m_k, \ldots$, тогда ему придется решить систему линейных уравнений вида:

$$\begin{aligned} \mathbf{v_1 x} &= m_1, \\ \mathbf{v_1 x} &= m_2, \\ &\vdots \\ \mathbf{v_k x} &= m_k, \end{aligned} \tag{4}$$

неизвестными в данном случае являются $\mathbf{x}$ и $m_1, m_2, \ldots m_k$, где k – количество открытых ключей, которые доступны злоумышленнику. Очевидно, что при таких условиях система (4) имеет бесконечное число решений и, таким образом, не позволяет однозначно определить секретный ключ. При этом, для $k \geq n$ на построение открытых ключей накладывается условие: для любых n открытых ключей $\operatorname{Rank} \mathbf{v} < n$.

Гомоморфизм операций

Сложение открытых ключей

Сложение целых чисел можно выразить следующим образом:

$$m_1 + m_2 = \mathbf{v_1 x} + \mathbf{v_2 x} = (\mathbf{v_1} + \mathbf{v_2})\mathbf{x}, \tag{5}$$

следовательно из (5) можно определить сложение с открытыми ключами:

$$m_1 + m_2 \Leftrightarrow \mathbf{v_1} + \mathbf{v_2} = \mathbf{v},$$

где $\mathbf{v} \in \mathbb{Z}^n$.

Умножение открытого ключа на число

Предположим, что необходимо умножить открытый ключ на число (незашифрованный элемент), тогда это будет выглядеть так:

$$mc = (\mathbf{vx})c = (\mathbf{v}c)\mathbf{x}, \tag{6}$$

следовательно из (6) можно определить умножение открытого ключа на число:

$$mc \Leftrightarrow \mathbf{v}c = \mathbf{v}^*,$$

где $\mathbf{v}^* \in \mathbb{Z}^n$.

Гомоморфизм операций

Умножение открытых ключей

Рассмотрим умножение более подробно, в координатном представлении:

$$m_1 m_2 = (\mathbf{v_1 x})(\mathbf{v_2 x}) = \left(\sum_{i=1}^{n} v_i^{(1)} x_i\right)\left(\sum_{i=1}^{n} v_i^{(2)} x_i\right) =$$

$$= \sum_{i,j=1}^{n} v_j^{(1)} x_j v_i^{(2)} x_i = \sum_{i,j=1}^{n} v_j^{(1)} v_i^{(2)} x_i x_j = \tag{7}$$

$$= \left(\sum_{i,j=1}^{n} v_j^{(1)} v_i^{(2)}\right)\left(\sum_{i,j=1}^{n} x_j x_i\right) = (\mathbf{v_1} \otimes \mathbf{v_2})(\mathbf{x} \otimes \mathbf{x}).$$

где $\otimes$ – тензорное произведение. Тогда умножение с открытыми ключами можно определить следующим образом:

$$m_1 m_2 \Leftrightarrow \mathbf{v_1} \otimes \mathbf{v_2}, \tag{8}$$

в данном виде умножение не замкнуто в $\mathbb{Z}^n$, так как $\mathbf{v_1} \otimes \mathbf{v_2} = \mathbf{u}, \quad \mathbf{u} \in \mathbb{Z}^{n^2}$.

Гомоморфизм операций

Построение матрицы линейного отображения

Для того, чтобы обеспечить замкнутость операции умножения в $\mathbb{Z}^n$ необходимо построить линейное отображения $\varphi_{\mathbf{G}} : \mathbb{Z}^{n^2} \to \mathbb{Z}^n$. Учитывая свойство $\mathbf{x}$ каждую компоненту $x_i x_j$ $(i, j = 1, 2, \ldots, n)$ вектора $\mathbf{x} \otimes \mathbf{x}$ можно выразить через скалярное произведение векторов $\mathbf{g_{i,j}x}$, то есть

$$x_i x_j = \sum_{i,j,k=1}^{n} g_{i,j,k} x_i.$$

Так как $\mathbf{x}$ фиксирован, то матрица $\mathbf{G} = g_{i,j,k}$ тоже может быть зафиксирована для всех открытых ключей. Теперь подставим матрицу $\mathbf{G}$ в произведение (7):

$$\sum_{i,j=1}^{n} v_j^{(1)} v_i^{(2)} x_i x_j = \sum_{i,j,k=1}^{n} v_j^{(1)} v_i^{(2)} g_{i,j,k} x_i = \left(\sum_{i,j,k=1}^{n} v_j^{(1)} v_i^{(2)} g_{i,j,k} \right) \left(\sum_{i=1}^{n} x_i \right) = \quad (9)$$

$$= (\mathbf{v_1} \otimes \mathbf{v_2})(\mathbf{G^T} \otimes \mathbf{x} = (\mathbf{G}(\mathbf{v_1} \otimes \mathbf{v_2}))\mathbf{x} = \mathbf{wx}.$$

где $\mathbf{w} \in \mathbb{Z}^n$, $\mathbf{G} \in \mathbb{Z}^{n^2 \times n}$.

Гомоморфизм операций

Умножение открытых ключей

Следовательно из (9) умножение открытых ключей, удовлетворяющее свойству замкнутости в $\mathbb{Z}^n$ примет вид:

$$m_1 m_2 \Leftrightarrow \mathbf{G}^{\mathsf{T}}(\mathbf{v_1} \otimes \mathbf{v_2}) = \mathbf{w},$$

где $\mathbf{w} \in \mathbb{Z}^n$.

Стойкость матрицы линейного отображения

Для того, чтобы злоумышленнику вычислить ключ из матрицы линейного отображения необходимо решить переопределенную систему нелинейных уравнений:

$$\begin{aligned} \mathbf{G_{1,1}x} &= x_1x_1, \\ \mathbf{G_{2,1}x} &= x_2x_1, \\ &\vdots \\ \mathbf{G_{n,n}x} &= x_nx_n, \end{aligned} \tag{10}$$

неизвестными в данном случае являются $\mathbf{x}$ и соответствующие компоненты $x_1x_1, x_2x_1, \ldots x_nx_n$ вектора $\mathbf{x} \otimes \mathbf{x}$. Очевидно, что при таких условиях система (10) имеет бесконечное число решений и, таким образом, не позволяет однозначно определить секретный ключ. При этом, на построение строк $\mathbf{G_{i,j}}$ $(i, j = 1, 2, \ldots, n)$ матрицы линейного отображения $\mathbf{G}$ накладывается условие, что при любом сочетании n строк ранг получаемой матрицы будет меньше n.

При практической реализации задается условие – ранг меньше 3.

Алгоритм защищенных вычислений

Вход: $m_1, m_2 \in \mathbb{Z}$.
Шаг 1 Построение криптосхемы.
Шаг 1.1 Генерация секретного ключа $\mathbf{x} \in \mathbb{Z}^n$.
Шаг 1.2 Вычисление матрицы линейного отображения $\mathbf{G} \in \mathbb{Z}^{n^2 \times n}$.
Шаг 2 Вычисление открытых ключей (зашифрование):

$$m_1 \to \mathbf{v_1}, \quad m_2 \to \mathbf{v_2}.$$

Шаг 3 Секретные вычисления:

$$\text{умножение открытых ключей: } \mathbf{G^T}(\mathbf{v_1} \otimes \mathbf{v_2}) = \mathbf{w},\ \mathbf{w} \in \mathbb{Z}^n,$$

$$\text{умножение открытого ключа на число: } \mathbf{v_1}c = \mathbf{u},\ \mathbf{u} \in \mathbb{Z}^n,$$

$$\text{сложение открытых ключей: } \mathbf{v_1} + \mathbf{v_2} = \mathbf{v},\ \mathbf{v} \in \mathbb{Z}^n.$$

Шаг 4 Расшифрование:

$$\text{произведения открытых ключей: } \mathbf{wx} = m_1 m_2,$$

$$\text{произведения открытого ключа на число: } \mathbf{ux} = (\mathbf{v_1 x})c = m_1 c,$$

$$\text{суммы открытых ключей: } \mathbf{vx} = m_1 + m_2.$$

Выход: $m_1 m_2,\ m_1 c,\ m_1 + m_2$.

Пример 1

Секретный ключ:

$$\mathbf{x} = \begin{bmatrix} 5 & 7 & 4 \end{bmatrix}^T.$$

Матрица линейного отображения:

$$\mathbf{G} = \begin{bmatrix} -33 & -507 & -372 & -507 & -249 & -60 & -372 & -60 & -528 \\ 14 & 186 & 136 & 186 & 94 & 24 & 136 & 24 & 192 \\ 23 & 317 & 232 & 317 & 159 & 40 & 232 & 40 & 328 \end{bmatrix}^T.$$

Несложно проверить, что $\mathbf{G^T x} = \mathbf{x} \otimes \mathbf{x}$.
Вычисления будем проводить над $m_1 = 17$ и $m_2 = 6$, которым соответствуют открытые ключи:

$$\mathbf{v_1} = \begin{bmatrix} -345 & 126 & 215 \end{bmatrix},$$

$$\mathbf{v_2} = \begin{bmatrix} -54 & 20 & 34 \end{bmatrix}.$$

Сложение открытых ключей:

$$\mathbf{v_1} + \mathbf{v_2} = \begin{bmatrix} -345 & 126 & 215 \end{bmatrix} + \begin{bmatrix} -54 & 20 & 34 \end{bmatrix} = \begin{bmatrix} -399 & 146 & 249 \end{bmatrix}.$$

Расшифрование:

$$\begin{bmatrix} -399 & 146 & 249 \end{bmatrix} \cdot \begin{bmatrix} 5 & 7 & 4 \end{bmatrix}^T = 23.$$

Пример 1

Умножение открытых ключей:

$$\begin{bmatrix} -33 & -507 & -372 & -507 & -249 & -60 & -372 & -60 & -528 \\ 14 & 186 & 136 & 186 & 94 & 24 & 136 & 24 & 192 \\ 23 & 317 & 232 & 317 & 159 & 40 & 232 & 40 & 328 \end{bmatrix} \cdot \left(\begin{bmatrix} -345 \\ 126 \\ 215 \end{bmatrix} \otimes \begin{bmatrix} -54 \\ 20 \\ 34 \end{bmatrix} \right) =$$

$$= \begin{bmatrix} 10013418 \\ -3615948 \\ -6188838 \end{bmatrix}.$$

Расшифрование:

$$\begin{bmatrix} 10013418 \\ -3615948 \\ -6188838 \end{bmatrix}^T \cdot \begin{bmatrix} 5 \\ 7 \\ 4 \end{bmatrix} = 102.$$

Применение модулярной арифметики

- В данном случае возможность применения модулярной арифметики объясняется тем, что все преобразования выполняются над целыми числами. Условием применения является знание верхней оценки получаемого результата, это необходимо для выбора соответствующих модулей.
- Применение Китайской теоремы об остатках дает возможность распараллеливания вычислений, при этом подбор модулей, соответствующих разряднсти вычислителей, обеспечивает их функционирование без переполнения разрядной сетки.
- Еще о стойкости и модулярной арифметике. Применение модулярной арифметики позволяет не только распараллеливать процесс вычислений, но и усиливать стойкость. Это связано с тем, что для восстановления результата необходимо знать остатки по всем модулям, в противном случае, результат восстановить не возможно (в случае, когда мы не используем избыточные модули для контроля ошибок). Поэтому для усиления стойкости часть параллельных процессов может обрабатываться в доверенной вычислительной среде.

Пример 2

Теперь рассмотрим пример 1 с применением модулярной арифметики. Верхний диапазон вычислений в нашем случае равен 102. Необходимо взять модуль больше 102, для этого нам подойдут модули 3, 5, 7, так как $3 \times 5 \times 7 = 105$. Представим матрицу линейного отображения $\mathbf{G}$ и открытые ключи $\mathbf{v_1}$ и $\mathbf{v_2}$ по модулям 3, 5, 7:

$$|\mathbf{G}|_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 2 & 2 & 1 & 2 & 0 & 1 & 1 & 1 & 1 \end{bmatrix},$$

$$|\mathbf{G}|_5 = \begin{bmatrix} 2 & 3 & 3 & 3 & 1 & 0 & 3 & 0 & 2 \\ 4 & 1 & 1 & 1 & 4 & 4 & 1 & 4 & 2 \\ 3 & 2 & 2 & 2 & 4 & 0 & 2 & 0 & 3 \end{bmatrix},$$

$$|\mathbf{G}|_7 = \begin{bmatrix} 2 & 4 & 6 & 4 & 3 & 3 & 6 & 3 & 4 \\ 0 & 4 & 3 & 4 & 3 & 3 & 3 & 3 & 3 \\ 2 & 2 & 1 & 2 & 5 & 5 & 1 & 5 & 6 \end{bmatrix},$$

$$|\mathbf{v_1}|_3 = \begin{bmatrix} 0 & 0 & 2 \end{bmatrix}, \quad |\mathbf{v_1}|_5 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}, \quad |\mathbf{v_1}|_7 = \begin{bmatrix} 5 & 0 & 5 \end{bmatrix},$$

$$|\mathbf{v_2}|_3 = \begin{bmatrix} 0 & 2 & 1 \end{bmatrix}, \quad |\mathbf{v_2}|_5 = \begin{bmatrix} 1 & 0 & 4 \end{bmatrix}, \quad |\mathbf{v_2}|_7 = \begin{bmatrix} 2 & 6 & 6 \end{bmatrix}.$$

Пример 2

Сложение открытых ключей:

$$|\mathbf{v_1} + \mathbf{v_2}|_3 = \begin{bmatrix} 0 & 2 & 0 \end{bmatrix},$$

$$|\mathbf{v_1} + \mathbf{v_2}|_5 = \begin{bmatrix} 1 & 1 & 4 \end{bmatrix},$$

$$|\mathbf{v_1} + \mathbf{v_2}|_7 = \begin{bmatrix} 0 & 6 & 4 \end{bmatrix}.$$

Выполняем обратное восстановление по Китайской теореме об остатках:

$$\left| 2 \cdot 35 \cdot \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix} + 1 \cdot 21 \cdot \begin{bmatrix} 1 \\ 1 \\ 4 \end{bmatrix} + 1 \cdot 15 \cdot \begin{bmatrix} 0 \\ 6 \\ 4 \end{bmatrix} \right|_{105} = \left| \begin{bmatrix} 21 \\ 41 \\ 39 \end{bmatrix} \right|_{105}.$$

Расшифрование:

$$\left| \begin{bmatrix} 21 & 41 & 39 \end{bmatrix} \begin{bmatrix} 5 \\ 7 \\ 4 \end{bmatrix} \right|_{105} = 23.$$

Пример 2

Умножение открытых ключей:

$$|\mathbf{G}^{\mathbf{T}}(\mathbf{v_1} \otimes \mathbf{v_2})|_3 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix},$$

$$|\mathbf{G}^{\mathbf{T}}(\mathbf{v_1} \otimes \mathbf{v_2})|_5 = \begin{bmatrix} 3 & 2 & 2 \end{bmatrix},$$

$$|\mathbf{G}^{\mathbf{T}}(\mathbf{v_1} \otimes \mathbf{v_2})|_7 = \begin{bmatrix} 2 & 0 & 2 \end{bmatrix}.$$

Выполняем обратное восстановление по Китайской теореме об остатках:

$$\left| 2 \cdot 35 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + 1 \cdot 21 \cdot \begin{bmatrix} 3 \\ 2 \\ 2 \end{bmatrix} + 1 \cdot 15 \cdot \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix} \right|_{105} = \left| \begin{bmatrix} 93 \\ 42 \\ 72 \end{bmatrix} \right|_{105}.$$

Расшифрование:

$$\left| \begin{bmatrix} 93 & 42 & 72 \end{bmatrix} \begin{bmatrix} 5 \\ 7 \\ 4 \end{bmatrix} \right|_{105} = 102.$$

Шифрование с рациональными числами

Необходимо определить операции с рациональными числами, так как это пригодится для вычисления систем линейных уравнений. Рациональное число представляется парой чисел, наприм, (q, p). Шифровать будем отдельно – числитель $q \to \mathbf{v}$ и знаменатель $p \to \mathbf{u}$. Для рациональных чисел умножение определено так:

$$(q_1, p_1)(q_2, p_2) = (q_1 q_2, p_1 p_2),$$

сложение:

$$(q_1, p_1) + (q_2, p_2) = (q_1 p_2 + q_2 p_1, p_1 p_2).$$

Тогда операции с открытыми ключами примут вид:
умножение:

$$(q_1, p_1)(q_2, p_2) \Leftrightarrow (\mathbf{G}(\mathbf{v_1} \otimes \mathbf{v_2}), \mathbf{G}(\mathbf{u_1} \otimes \mathbf{u_2})),$$

сложение:

$$(q_1, p_1) + (q_2, p_2) \Leftrightarrow (\mathbf{G}(\mathbf{v_1} \otimes \mathbf{u_2}) + \mathbf{G}(\mathbf{u_1} \otimes \mathbf{v_2}), \mathbf{G}(\mathbf{u_1} \otimes \mathbf{u_2})).$$

Решение систем линейных уравнений с трех диагональной матрицей

Система линейных уравнений с трех диагональной матрицей в обычном виде выглядит так:

$$\begin{bmatrix} a_{1,1} & a_{1,2} & 0 & 0 & 0 & \dots & 0 \\ a_{2,1} & a_{2,2} & a_{2,3} & 0 & 0 & \dots & 0 \\ 0 & a_{3,2} & a_{3,3} & a_{3,4} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & \dots & 0 & 0 & a_{n,n-1} & a_{n,n} \end{bmatrix} \cdot \begin{bmatrix} \chi_1 \\ \chi_2 \\ \chi_3 \\ \vdots \\ \chi_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_n \end{bmatrix}. \qquad (11)$$

где $a_{i,j}$ и b_j – целые числа (систему уравнений можно привести к целочисленным коэффициентам) $(i, j = 1, 2, \dots, n)$, что касается χ_j, то с большой вероятностью они будут рациональными числами.

Решение систем линейных уравнений с трех диагональной матрицей

Далее заменим коэффициенты $a_{i,j}$ и b_j отличные от нуля на открытые ключи $\mathbf{v_{i,j}}$ и $\mathbf{w_{i,j}}$ соответственно, χ_j будут иметь представление в виде пары векторов $(\mathbf{u_i}, \mathbf{q_i})$:

$$\begin{bmatrix} \mathbf{v_{1,1}} & \mathbf{v_{1,2}} & 0 & 0 & 0 & \dots & 0 \\ \mathbf{v_{2,1}} & \mathbf{v_{2,2}} & \mathbf{v_{2,3}} & 0 & 0 & \dots & 0 \\ 0 & \mathbf{v_{3,2}} & \mathbf{v_{3,3}} & \mathbf{v_{3,4}} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & \dots & 0 & 0 & \mathbf{v_{n,n-1}} & \mathbf{v_{n,n}} \end{bmatrix} \cdot \begin{bmatrix} (\mathbf{u_1}, \mathbf{q_1}) \\ (\mathbf{u_2}, \mathbf{q_2}) \\ (\mathbf{u_3}, \mathbf{q_3}) \\ \vdots \\ (\mathbf{u_n}, \mathbf{q_n}) \end{bmatrix} = \begin{bmatrix} \mathbf{w_1} \\ \mathbf{w_2} \\ \mathbf{w_3} \\ \vdots \\ \mathbf{w_n} \end{bmatrix}. \quad (12)$$

Применение модулярной арифметики к вычислению системы уравнений (12) имеет ряд ограничений, необходимых для успешного применения шифрования:

1. Система уравнений (11) должна быть устойчивой, так как, в противном случае, это может привести к слишком большому росту размерности элементов открытых ключей.
2. Необходимо знать верхнюю оценку результата вычисления (11) для выбора соответствующих модулей.
3. Необходимо выбирать модули, при которых определитель матрицы системы уравнений (12) не равен нулю.

Пример 3

Дана СЛУ с трех диагональной матрицей:

$$\begin{bmatrix} 3 & 2 & 0 \\ 5 & 3 & 10 \\ 0 & 7 & 4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 5 \\ 11 \\ 1 \end{bmatrix}.$$

После зашифрования (секретный ключ **x** и матрица **G** из примеров 1, 2) СЛУ примет вид:

$$\begin{bmatrix} [-92\ \ 53\ \ 23] & [67\ \ -67\ \ 34] & 0 \\ [116\ \ -65\ \ -30] & [-92\ \ 53\ \ 23] & [-92\ \ 54\ \ 23] \\ 0 & [-120\ \ -207\ \ 514] & [-120\ \ -208\ \ 515] \end{bmatrix} \cdot \begin{bmatrix} (\mathbf{u_1}, \mathbf{q_1}) \\ (\mathbf{u_2}, \mathbf{q_2}) \\ (\mathbf{u_3}, \mathbf{q_3}) \end{bmatrix} = \begin{bmatrix} [116\ \ -65\ \ -30] \\ [-120\ \ -207\ \ 515] \\ [-120\ \ -205\ \ 509] \end{bmatrix}.$$

После прямой прогонки СЛУ примет вид:

$$\begin{bmatrix} [-92\ 53\ 23] & [67\ \ -67\ 34] & 0 \\ 0 & [475593\ \ -171742\ 293943] & [5146290\ \ -1858380\ \ -3180690] \\ 0 & 0 & [-96625732026\ \ 34892625436\ \ 59720070466] \end{bmatrix} \cdot \begin{bmatrix} (\mathbf{u_1}, \mathbf{q_1}) \\ (\mathbf{u_2}, \mathbf{q_2}) \\ (\mathbf{u_3}, \mathbf{q_3}) \end{bmatrix} =$$

$$= \begin{bmatrix} [116\ \ -65\ \ -30] \\ [-1046856\ 378032\ 647016] \\ [31246993089\ \ -11283636398\ \ -19312377679] \end{bmatrix}.$$

Пример 3

Обратная прогонка:

$$(\mathbf{u_3}, \mathbf{q_3}) = ([31246993089\ -\ 11283636398\ -\ 19312377679], [-96625732026\ 34892625436, 59720070466]),$$

$$\chi_3 = (\mathbf{u_3}, \mathbf{q_3})\mathbf{x} = (-57, -214) = \frac{57}{214}.$$

$$(\mathbf{u_2}, \mathbf{q_2}) = ([-3173857753900563569 4\ 11461153000196479556\ 19616204173413205394],$$
$$[-24450361909295938566\ 8829297356134644500\ 15111682013384295386]),$$

$$\chi_2 = (\mathbf{u_2}, \mathbf{q_2})\mathbf{x} = (-2, 214) = -\frac{1}{107}.$$

$$(\mathbf{u_1}, \mathbf{q_1}) = ([-109070463681729360786114\ 39386556329513380283964\ 6741160602551328548 5974],$$
$$[13607115409761375740682 06\ -\ 4913680564636052350801 32\ -\ 8409953274088628061948 66]),$$

$$\chi_1 = (\mathbf{u_1}, \mathbf{q_1})\mathbf{x} = (1074, 642) = \frac{179}{107}.$$

Вот так результаты обратной прогонки будут выглядеть по модулям 11, 13, 17:

$$|(\mathbf{u_3}, \mathbf{q_3})|_{11} = ([4 \ \ -1 \ \ -1], [-10\ 9\ 1]),$$
$$|(\mathbf{u_3}, \mathbf{q_3})|_{13} = ([0 \ \ -8 \ \ -10], [-5\ 6\ 4]),$$
$$|(\mathbf{u_3}, \mathbf{q_3})|_{17} = ([0 \ \ -9 \ \ -7], [-13\ 10\ 9]).$$

$$|(\mathbf{u_2}, \mathbf{q_2})|_{11} = ([0 \ \ 9 \ \ 3], [-5 \ \ 1 \ \ 3]),$$
$$|(\mathbf{u_2}, \mathbf{q_2})|_{13} = ([-6 \ \ 2 \ \ 10], [-2 \ \ 7 \ \ 8]),$$
$$|(\mathbf{u_2}, \mathbf{q_2})|_{17} = ([-15 \ \ 1 \ \ 8], [-2 \ \ 11 \ \ 7]).$$

$$|(\mathbf{u_1}, \mathbf{q_1})|_{11} = ([-7 \ \ 8 \ \ 2], [10 \ \ -10 \ \ -5]),$$
$$|(\mathbf{u_1}, \mathbf{q_1})|_{13} = ([-9 \ \ 5 \ \ 11], [11 \ \ -5 \ \ -7]),$$
$$|(\mathbf{u_1}, \mathbf{q_1})|_{17} = ([-10 \ \ 10 \ \ 0], [5 \ \ -13 \ \ -10]).$$