УДК 519.7 DOI 10.17223/2226308X/14/13

РАЗРАБОТКА И АНАЛИЗ ОРАКУЛА ДЛЯ ГИБРИДНОЙ АТАКИ НА КРИПТОГРАФИЧЕСКУЮ СИСТЕМУ NTRU С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМА КВАНТОВОГО ПОИСКА[1]

А. О. Бахарев

В силу развития квантовых вычислений возникает необходимость в разработке и анализе криптосистем, устойчивых к атакам с использованием квантового компьютера — алгоритмов постквантовой криптографии. Стойкость многих известных постквантовых криптосистем, основанных на теории решёток, базируется на сложности решения проблемы нахождения кратчайшего вектора в решетке (SVP). Разработана и проанализирована модель квантового оракула, необходимого для реализации гибридного квантово-классического алгоритма решения задачи SVP. На примере постквантовой криптосистемы с открытым ключом NTRU, являющейся финалистом третьего раунда конкурса NIST, получены верхние оценки на число кубит и глубину схемы, требуемые для реализации данного оракула, в зависимости от параметров криптосистемы.

Ключевые слова: *криптосистема NTRU, квантовый поиск, криптография с открытым ключом, постквантовая криптография.*

Квантовые вычисления — это быстроразвивающаяся область компьютерных исследований, которая ставит под угрозу криптографическую стойкость стандартов асимметричного шифрования, используемых в настоящее время. В 2016 г. Национальный Институт Стандартов и Технологий США (NIST) объявил конкурс «Post-Quantum Cryptography Competition», по завершении которого будет принят новый — квантово-устойчивый — стандарт асимметричного шифрования. Претендентами являются подходы на основе решёток, кодов, хэш-функций, изогений и многочленов от многих переменных.

Рассмотрим подход на основе решёток.

Определение 1. Пусть $u_1, \ldots, u_n \in \mathbb{R}^m$ — линейно независимые векторы, $n \leqslant m$. *Решёткой* называется множество

$$\mathbb{Z}u_1 \oplus \ldots \oplus \mathbb{Z}u_n = \Big\{ \sum_{i=1}^{n} b_i u_i : b_i \in \mathbb{Z} \Big\}.$$

Векторы $u_1, \ldots, u_n$ называются *базисом решётки.*

Одной из задач в теории решёток является *задача нахождения кратчайшего вектора* (SVP), которая заключается в нахождении вектора, имеющего наименьшую длину, в решётке, заданной своим базисом. В общем случае SVP является NP-трудной задачей. Стойкость систем, основанных на решётках, зависит от эффективности решения SVP, так как большинство известных атак сводятся к решению этой проблемы. Перспективными являются разработка и анализ квантовых алгоритмов, которые позволяют ускорить решение данной задачи.

В [1] представлен гибридный квантово-классический подход к поиску кратчайшего вектора решётки на основе GaussSieve [2] — одного из самых эффективных классических алгоритмов (алгоритм 1).

[1]Работа выполнена в рамках госзадания ИМ СО РАН (проект № 0314-2019-0017) при поддержке лаборатории криптографии JetBrains Research.

Алгоритм 1. Алгоритм GaussSieve (D. Micciancio and P. Voulgaris, 2010)

Вход: B — базис решётки.
Выход: v — кратчайший вектор решётки.
1: Инициализировать пустой неупорядоченный список L и пустой стек S
2: **Повторять**
3: получить вектор v из стека (или сгенерировать новый).
4: **Пока** $w \leftarrow$ ПОИСК$\{w \in L : \|v \pm w\| \leqslant \|v\|\}$
5: уменьшить v с помощью w $(v \leftarrow v \pm w)$.
6: **Пока** $w \leftarrow$ ПОИСК$\{w \in L : \|w \pm v\| \leqslant \|w\|\}$
7: удалить w из листа L;
8: уменьшить w с помощью v $(w \leftarrow w \pm v)$;
9: добавить w в стек S.
10: **Если** v изменился, **то**
11: добавить v в стек S,
12: **иначе**
13: добавить v в лист L.
14: **Пока** v не станет кратчайшим вектором.
15: **Вернуть** вектор v.

На вход алгоритма поступает базис решётки, на основе которого будут строиться новые векторы при условии пустого стека S. Функция «ПОИСК» перебирает векторы w в списке и проверяет одно из *условий поиска*: $\|v \pm w\| \leqslant \|v\|$ или $\|w \pm v\| \leqslant \|w\|$; если такой вектор существует, то функция возвращает его, иначе цикл прерывается. В [2] предложено эвристическое условие остановки, которое основывается на количестве коллизий; алгоритм работает до тех пор, пока не получим такое число коллизий, что будем уверены, что нашли кратчайший вектор.

В рамках предложенного в [1] подхода ускорение достигается за счёт использования в функции «ПОИСК» квантового алгоритма Гровера поиска в неупорядоченном списке [3]. Задача, решаемая этим алгоритмом, называется *задачей поиска*. Предполагается, что есть неупорядоченный список из K элементов, в котором как минимум один элемент удовлетворяет некоторому условию. Требуется найти по крайней мере один такой элемент. Другими словами, определена булева функция f, которая по номеру элемента (его двоичному представлению) определяет, является ли элемент подходящим (в этом случае $f = 1$) или нет ($f = 0$). В такой постановке задача поиска сводится к нахождению решениий уравнения $f(x) = 1$.

В классическом варианте при условии, что решение одно, требуется $\sim K/2$ обращений к функции f для нахождения решения. Квантовый алгоритм поиска элемента в неупорядоченном списке решает задачу примерно за $\sqrt{K}$ обращений к *оракулу* — квантовому аналогу функции f.

Квантовый компьютер, в отличие от обычного, оперирует *кубитами* [4]. Их состояние можно представить как единичный вектор из $\mathbb{C}^2$. Произвольный вектор этого пространства может быть представлен в виде

$$|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle,$$

где $\alpha, \beta \in \mathbb{C}$ называются *амплитудами*; $|\alpha|^2$ и $|\beta|^2$ — вероятности обнаружения кубита после измерения в состояниях $|0\rangle$ и $|1\rangle$ соответственно. Говорят, что кубит находится в *суперпозиции* состояний $|0\rangle$ и $|1\rangle$.

В соответствии с постулатами квантовой механики, состояние системы из n кубит описывается *вектором состояний* из $\mathbb{C}^{2^n}$. Эволюция состояния замкнутой квантовой системы во времени описывается унитарным преобразованием.

Известно, что любая булева функция может быть реализована на квантовом компьютере, а квантовым алгоритмом, решающим задачу поиска, является *алгоритм Гровера* (рис. 1).

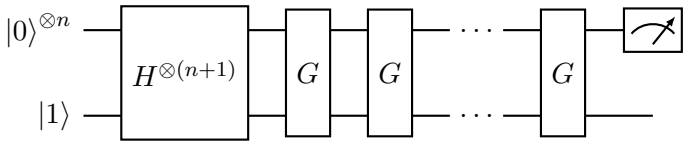

Рис. 1. Алгоритм Гровера [3]: H — вентиль Адамара; G — итерации Гровера (рис. 2)

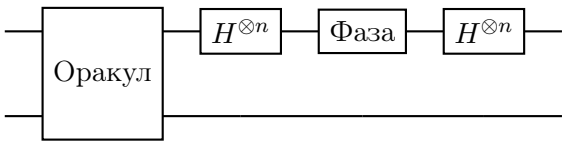

Рис. 2. Итерация Гровера

Преобразования $H^{\otimes n}$ и Фаза являются известными вентилями, в отличие от оракула, который строится под каждую задачу отдельно. В настоящей работе выполнено построение и описание оракула для квантового подхода к решению задачи поиска подходящего вектора из списка в алгоритме GaussSieve.

Оракул, представленный на рис. 3, состоит из двоичного представления номера вектора в списке, K векторов размерности d, каждая координата которых кодируется строкой длины m, переключателя, проверки условия поиска и ответа. Его работа происходит следующим образом:

1) получение номера вектора на вход и передача его в переключатель;
2) выбор по номеру вектора из списка и копирование его;
3) проверка условия поиска для скопированного вектора;
4) вывод ответа: 1 — если вектор удовлетворяет условию, 0 — если нет.

Переключатель представляет собой векторную булеву функцию, которая номеру вектора сопоставляет строку: $i \to (0, \ldots, 0, 1, 0, \ldots, 0)$, где 1 стоит на i-м месте. Тогда, применяя вентиль CCNOT, можно удобно копировать вектор с номером i из списка. Пример для $i = 2$ и $K = 2$ приведён на рис. 4. Здесь первые два кубита представляют собой строку, полученную из переключателя; v_1 и v_2 — векторы размерности d, каждая координата которых кодируется строкой длины m; нижний регистр оставлен для копирования нужного вектора. В итоге работы вектор v_2 будет скопирован в нижний регистр.

Проверка условия поиска содержит следующие операции: сложение, вычитание, возведение в квадрат и сравнение целых чисел. Предлагается использовать дополнительный код числа для операции вычитания, таким образом, сложение и вычитание реализуются одной операцией, а сравнение чисел определяется знаком результата вычитания. Сложность реализации операций на квантовом компьютере оцениваетя количеством кубит и глубиной схемы. В табл. 1 представлены оценки сложности операций. При каждом изменении вектора v или списка L в ходе работы алгоритма GaussSieve оракул строится заново.

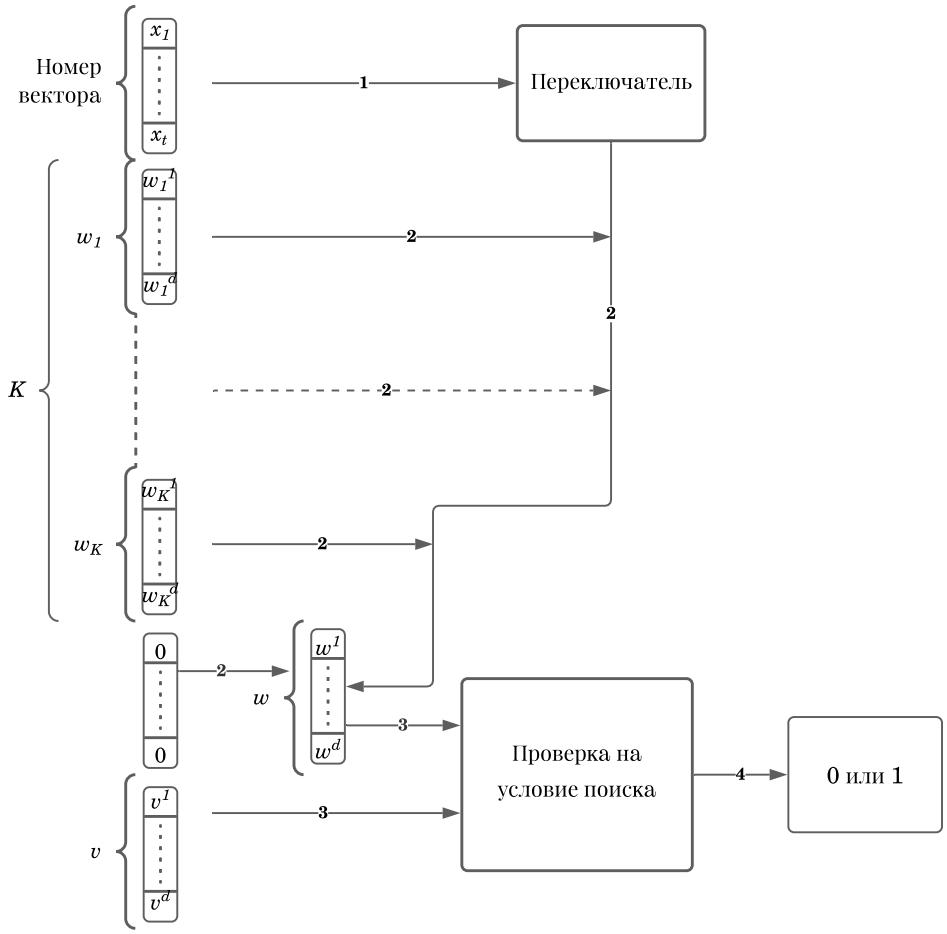

Рис. 3. Предлагаемая схема квантового оракула

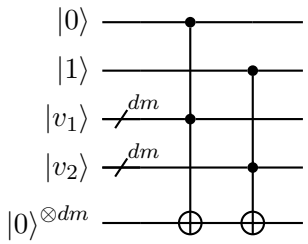

Рис. 4. Пример копирования векторов из списка

Таблица 1

Количество кубит и глубина схемы, достаточные для реализации операций

Операция	Количество кубит	Глубина схемы
Сложение, вычитание целых m-битных чисел	$4m-1$	$5m-2$
Возведение целого m-битного числа в квадрат	$6m^2-5m+2$	$11m^2-15m+4$
Переключатель, где номер вектора представляется целым m-битным числом	2^m+m	3^m
Перевод целого m-битного числа в дополнительный код	$4m$	$4m+1$

Утверждение 1. Пусть имеется список длины K, состоящий из целочисленных векторов размерности d, каждая координата которых кодируется битовой строкой длины m. Тогда для реализации квантового оракула, представленного на рис. 3, потребуется не более $\lceil \log_2 K \rceil + Kdm + K + 18dm^2 - 33dm + 6d^2 + 25d + 2m + 4$ кубит. Глубина схемы не превосходит $3^{\lceil \log_2 K \rceil} + Kdm + 33dm^2 - 67dm + 15d^2 + 35d - 2m + 19$.

Для анализа была выбрана криптосистема NTRU, так как она прошла в третий раунд конкурса NIST [5] и является одним из четырёх претендентов на новый постквантовый стандарт асимметрического шифрования. NTRU зависит от трёх целочисленных параметров (N, p, q), где $(p, q) = 1$. Работа осуществляется в кольце R полиномов степени не выше $N-1$ с целочисленными коэффициентами, то есть $R = \mathbb{Z}[x]/(x^N - 1)$. Элемент $F = \sum\limits_{i=0}^{N-1} F_i x^i \in R$ можно представить как вектор

$$F = [F_0, \ldots, F_{N-1}].$$

Операция умножения «$*$» в R определяется как результат циклической свёртки:

$$F * G = H,$$

$$H_k = \sum_{i=0}^{k} F_i G_{k-i} + \sum_{i=k+1}^{N-1} F_i G_{N+k-i} = \sum_{i+j=k(\mathrm{mod}N)} F_i G_j.$$

Если выполняется умножение полиномов по модулю числа, то коэффициенты приводятся по этому модулю.

Секретный ключ: f, g — полиномы из R с координатами из множества $\{-1, 0, 1\}$.

Открытый ключ: N, p, q, $h = f_q * g \bmod q$, где $f_q * f = 1 \pmod q$.

Зашифрование: Пусть m — сообщение, представленное в виде полинома из R с коэффициентами из интервала $(-p/2, p/2]$. Тогда зашифрованное сообщение c вычисляется следующим образом: $c = p\varphi * h + m \bmod q$, где φ — полином из R с некоторыми ограничениями на координаты из множества $\{-1, 0, 1\}$.

Расшифрование: Определим полином $a = f * c \bmod q$. Тогда исходное сообщение восстанавливается следующим образом: $m = f_q * a \bmod p$.

Одна из самых эффективных атак [6] на NTRU сводится к решению SVP в решётке, базис которой образован строками матрицы M, построенной на основе открытого ключа:

$$M = \begin{pmatrix} 1 & 0 & \ldots & 0 & h_0 & h_1 & \ldots & h_{N-1} \\ 0 & 1 & \ldots & 0 & h_{N-1} & h_0 & \ldots & h_{N-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 & h_1 & h_2 & \ldots & h_0 \\ 0 & 0 & \ldots & 0 & q & 0 & \ldots & 0 \\ 0 & 0 & \ldots & 0 & 0 & q & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 & 0 & 0 & \ldots & q \end{pmatrix}$$

С большой вероятностью кратчайший вектор решётки, порождённой этим базисом, имеет вид $r = (f, g)$, а параметры оракула можно определить (оценить) следующим образом: $K \leqslant 2N$, $d = 2N$, $m = \lceil \log_2 q \rceil + 1$.

На основе оценок из табл. 1 посчитана взаимосвязь между параметрами NTRU, количеством кубит и глубиной схемы, достаточных для реализации гибридной квантово-классической атаки (табл. 2).

Таблица 2

Верхние оценки числа кубит и глубины схемы

Параметры NTRU	Количество кубит	Глубина схемы
$N = 1$, $q = 2$	105	332
$N = 2$, $q = 2$	266	428
$N = 8$, $q = 4$	3742	3498
$N = 256$, $q = 128$	4138013	2945510

Таким образом, в работе получены верхние оценки сложности реализации квантового оракула из алгоритма Гровера для реализации гибридного квантово-классического алгоритма на основе GaussSieve, который может быть использован для атак на криптосистемы, стойкость которых зависит от решения задачи SVP. Проанализирована сложность реализации квантового оракула для атаки на постквантовую криптосистему NTRU. На сегодняшний день количество кубит, с которыми оперирует квантовый компьютер, не превосходит 76 [7]. Из полученных оценок следует, что предложенная модель квантового оракула не может быть реализована на квантовом компьютере даже для самых малых параметров NTRU, так как ещё не существует квантового компьютера, оперирующего достаточным количеством кубит. В рамках дальнейшей работы предлагается оптимизировать квантовую схему оракула, получить необходимые оценки для реализации оракула данного класса, а также проанализировать другие известные классические атаки на постквантовые криптосистемы с целью изучения возможности их ускорения с помощью квантовых вычислений.

ЛИТЕРАТУРА

1. *Laarhoven T., Mosca M., and van de Pol J.* Finding shortest lattice vectors faster using quantum search // Des. Codes Cryptogr. 2015. V. 77. No. 2–3. P. 375–400.
2. *Micciancio D. and Voulgaris P.* Faster exponential time algorithms for the Shortest Vector roblem // 21st Ann. ACM Symp. Discrete Algorithms (SODA). 2010. P. 1468–1480.
3. *Grover L. K.* A fast quantum mechanical algorithm for database search // 28th Ann. ACM Symp. Theory Comput. (STOC). 1996. P. 212–219.
4. *Nielsen M. A. and Chuang I. L.* Quantum Computation and Quantum Information. Cambridge: Cambridge University Press, 2010.
5. `https://csrc.nist.gov/Projects/post-quantum-cryptography/round-3-submissions`.
6. *Chen C., Danba O., Hoffstein J., et al.* NTRU Algorithm Specifications and Supporting Documentation. `https://ntru.org/`, 2019.
7. *Zhong H.-S., Wang H., Deng Y.-H., et al.* Quantum computational advantage using photons. Science. 2020. V. 370. Iss. 6523. P. 1460–1463.

УДК 519.7 DOI 10.17223/2226308X/14/14

КРИПТОАНАЛИТИЧЕСКАЯ ОБРАТИМОСТЬ ФУНКЦИЙ ДВУХ АРГУМЕНТОВ

Н. Ю. Бердникова, И. А. Панкратова

Предложены тесты криптоаналитической обратимости всех возможных типов для произвольных функций от двух аргументов. Сформулированы алгоритмы построения функции восстановления и генерации обратимых функций; посчитано количество обратимых функций некоторых типов.