УДК 004.056.55

ОЦІНКА СТІЙКОСТІ ГЕШ-ФУНКЦІЇ КУПИНА В КВАНТОВІЙ МОДЕЛІ ОБЧИСЛЕНЬ

Л. І. Пелешенко[1]

[1] Навчально-науковий Фізико-технічний інститут

Анотація

У роботі проведено аналіз стійкості геш-функції Купина у квантовій моделі обчислень. Розглянуто застосування трьох квантових алгоритмів: Гровера, Брассара-Хоєра-Таппа (BHT) та Шайу-Наї-Пласенсії-Шроттенльоєра (CNS). Наведено їхні оцінки складності (за кількістю запитів до оракула, кубітів та вентилів) та особливості реалізації для режимів Купина-256 та Купина-512. Обчислено, що квантовий алгоритм Гровера забезпечує оптимальну ефективність при пошуку прообразів геш-функції, коли задано одне цільове геш-значення. Тоді як квантовий алгоритм CNS демонструє найкращі показники для задачі пошуку колізій та є ефективнішим для пошуку прообразу у випадку наявності множини цільових значень.

Ключові слова: геш-функція Купина, квантовий алгоритми Гровера, квантовий алгоритм Брассара-Хоєра-Таппа, квантовий алгоритмШайу-Наї-Пласенсії-Шроттенльоєра

Вступ

У 2015 році геш-функція Купина була прийнята як національний криптографічний стандарт ДСТУ 7564:2014 «Інформаційні технології. Криптографічний захист інформації. Функція гешування» [1]. Геш-функція Купина використовує функцію стиснення Девіса-Мейєра, що ґрунтується на схемі Івена-Мансура, а внутрішні перестановки побудовані на перетвореннях блокового шифру Калина (ДСТУ 7624:2014).

Наразі є дуже мало досліджень, які стосуються квантового криптоанналізу геш-функції Купина, що ускладнює оцінку її стійкості в умовах квантової моделі обчислень. Тож ця тема потребує подальшого дослідження.

1. Особливості побудови геш-функції Купина

Нехай V_i — i-вимірний векторний простір над полем $GF(2)$, $i \geqslant 1$; l — розмір внутрішнього стану геш-функції; N — довжина (у бітах) повідомлення M без доповнення; n — довжина геш-значення. Із повідомлення M та вектора ініціалізації IV функція H обчислює геш-значення $H(IV, M)$:

$$H(IV, \cdot) : V_N \to V_n, n \in \{8 \cdot s | s = 1, 2, \dots, 64\},$$
$$M \in V_N, N \in \{0, 1, \dots, 2^{96} - 1\},$$
$$IV \in V_l, l \in \{512, 1024\}.$$

Нехай h_ν — значення, яке оновлює внутрішній стан на кожному блоці повідомлення m_ν ($\nu = 1, 2, \dots, k$ — номер блоку); $T_l^{\oplus}$ та T_l^{+} — бієктивні відображення $V_l \to V_l$, що реалізовані як ітеративне застосування кількох функцій, що приймають вхідний аргумент $x \in V_l$ як матрицю розміром $8 \times c$ байтів, представлену як елементи скінченного поля $GF(2^8)$; $R_{l,n}$ — функція, що повертає n старших біт з вхідного блоку x довжиною l біт ($n < 1$), де результат записується в молодші n біт обчисленого значення. Геш-значення обчислюють відповідно до такої ітеративної процедури:

$$h_0 = IV,$$
$$h_\nu = T_l^{\oplus}(h_{\nu-1} \oplus m_\nu) \oplus T_l^{+}(m_\nu) \oplus h_{\nu-1},$$
$$H(IV, M) = R_{l,n}(T_l^{\oplus}(h_k) \oplus h_k).$$

Відповідно до роботи [1], геш-функція приймає повідомлення (двійковий рядок) довжиною N бітів як вхідні дані. Доповнення кожного повідомлення, що відбувається незалежно від його довжини, розташоване після повідомлення та містить один біт «1», потім d нульових бітів, де $d = (-N - 97) \pmod{l}$, та 96 бітів, що містять довжину повідомлення N (молодші біти у представленні довжини повідомлення мають менші індекси). У результаті, доповнена бітова послідовність має довжину, кратну внутрішньому стану l, що розбивається на блоки l-бітної довжини:

$$l = \begin{cases} 512, & \text{якщо } 8 \leq n \leq 256, \\ 1024, & \text{якщо } 256 < n \leq 512. \end{cases}$$

Вектор ініціалізації IV ініціалізується як $1 \ll 510$ при $l = 512$ або $1 \ll 1023$ при $l = 1024$. Максимальна довжина повідомлення обмежена $(2^{96} - 1)$ бітами.

Основними рекомендованими режимами геш-функції Купина-n, де n — довжина геш-значення, є Купина-256 та Купина-512.

2. Конструкція квантових алгоритмів Гровера, BHT та CNS

Нехай n — довжина геш-значення; m_h — кількість кубітів, необхідна для реалізації геш-функції $H(x)$; t_h — кількість вентилів, необхідна для реалізації геш-функції $H(x)$; N — кількість вхідних даних геш-функції.

Особливості алгоритму Гровера

Головною перевагою алгоритму Гровера, призначеного для знаходження елементу у невідсортованому списку, є те, що він надає квадратичне прискорення відносно кількості елементів у списку у порівнянні з класичним лінійним пошуком. Це відбувається завдяки використанню квантової суперпозиції, що забезпечує одночасну обробку всіх станів.

Оракул O у алгоритмі Гровера реалізує таке унітарне відображення: $|x\rangle|q\rangle \to |x\rangle|q\rangle \oplus f(x)$, де q є цільовим кубітом, а $f(x)$ — булева функція, яка визначає правильний розв'язок. При запуску алгоритма Гровера оракул застосовується до стану $|x\rangle((|0\rangle-|1\rangle)/\sqrt{2}) \to (-1)^{f(x)}|x\rangle((|0\rangle-|1\rangle)/\sqrt{2})$, що еквівалентно фазовому зсуву на основі значення $f(x)$. Таким чином, оракул модифікує фазу лише тих станів, які відповідають розв'язку задачі. Детальніше функціонування оракула описано у роботі [2].

Застосування так званої ітерації Гровера, яка включає застосування оракула O_f, перетворення Адамара $H^{\otimes n}$, виконання умовного фазового зсуву для кожного стану, окрім нульового, та повторного застосування перетворення Адамара визначає оператор Гровера

$$G = H^{\otimes n}(2|0\rangle\langle 0| - I)\,H^{\otimes n}O_f = (2|\psi\rangle\langle\psi| - I)\,O_f,$$

де $|\psi\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x\rangle$ — суперпозиція усіх станів; I — одиничний оператор.

У загальному випадку алгоритм Гровера забезпечує складність $O(\sqrt{N})$ відносно кількості запитів, де N — розмір простору пошуку; $O(n)$ відносно кількості кубітів та $O(t_h \cdot \sqrt{N})$ відносно кількості вентилів, де n — довжина геш-значення.

Особливості алгоритму BHT

Алгоритм Брассара-Хоєра-Таппа (BHT), запропонований у роботі [3], комбінує класичні обчислення та алгоритм Гровера для пошуку колізій.

Нехай i — це індекс у межах списку з k елементів ($k = |K|$), а j — індекс з-поміж усіх N можливих значень, крім тих, які вже входять у K. Робота алгоритму полягає у виборі підмножини $K \subset X$ розміру k ($|X| = N$), побудові списку L із пар $(x_i, H(x_i))$, де $x_i \in K$, і подальший квантовий пошук $x_j \notin K$, для якого $H(x_j) = H(x_i)$. Довжина x_i і $H(x_i)$ — n біт.

Згідно з роботою [4], якщо замість множини K використовується фіксоване геш-значення h, то буде отримано варіант алгоритму BHT для пошуку прообразу.

У той час як, складність відносно кількості запитів становить $O(2^{n/3})$, оцінки складності алгоритму BHT відносно кількості кубітів та вентилів обчислені у роботі [4] такі:

- для пошуку колізії:
  - кубіти: $8n \cdot \sqrt[3]{N} + 2 \cdot \sqrt[3]{N} + m_h$;
  - вентилі: $t_h \cdot \sqrt[3]{N} + \sqrt[3]{N}\log\sqrt[3]{N} + (\sqrt[3]{N} - 1)(127(n-1)+36) + (127(n-1)+36+13n^2) \cdot \frac{\pi}{4}\sqrt{N}$.
- для пошуку прообразу:
  - кубіти: $4n \cdot \sqrt[3]{N} + 2n + 2$;
  - вентилі: $(13n^2 + 127(n-1) + 36) \cdot \frac{\pi}{4}\sqrt{N}$.

Особливості алгоритму CNS

Квантовий алгоритм CNS [5], призначений для пошуку колізій у геш-функціях, використовує класичну обробку для збереження частини гешів та квантовий пошук для виявлення елементів, що утворюють колізії.

Нехай $H : \mathcal{X} \to \mathcal{Y}$ — геш-функція з $|\mathcal{X}| = |\mathcal{Y}| = 2^n$. Алгоритм CNS формує підмножину $S \subset \mathcal{X}$ розміру $2^{n/5}$ шляхом випадкового вибору елементів $x \in \mathcal{X}$ та зберігає відповідні геш-значення $H(x)$ у таблиці T, а потім виконує квантовий пошук $x \notin S$ для якого $H(x) \in H(S)$, тобто геш-значення $H(x)$ збігається з деяким значенням $H(x')$, де $x' \in S$.

Згідно з роботою [5], алгоритм CNS використовує метод підсилення амплітуд, завдяки чому за кількістю запитів досягає складності $O(2^{2n/5})$ для пошуку колізій та $O(2^{3n/7})$ для пошуку прообразу до множини гешів при використанні одного квантового процесора. При цьому необхідна кількість кубітів для пошуку колізій становить $O(n)$ та вентилів — $O(t_h \cdot 2^{2n/5})$, а для пошуку прообразу до множини гешів — $O(n)$ кубітів та $O(t_h \cdot 2^{3n/7})$ вентилів.

3. Складність застосування квантових алгоритмів до Купини

Для коректного порівняння складності алгоритмів при застосуванні їх до геш-функції Купина необхідно врахувати кількість запитів, кубітів та вентилів, які вимагає кожен з алгоритмів.

Нехай c_i, де $i \in \mathbb{N}, i \geq 1$ — деяка константа, що буде використовуватись у визначенні складності алгоритмів для фіксованого значення. Вона є незначною, тож при оформленні таблиці її буде упущено для запобігання перевантаження таблиці.

Алгоритм Гровера

Складність алгоритму Гровера стосовно кількості запитів оцінюється як $O(\sqrt{2^n})$. Геш-функція Купина має довжину гешу $n = s \cdot 8, s \in \{1, 2, \ldots, 64\}$, тобто $n \in \{8, 16, \ldots, 512\}$. Оскільки розглядаємо режими геш-функції Купина-256 (n = 256) та Купина-512 (n = 512), то отримаємо складність за кількістю викликів $c_1 \cdot 2^{128}$ та $c_2 \cdot 2^{256}$ відповідно. Кількість необхідних кубітів та вентилів:

- для геш-функції Купина-256 — $c_3 \cdot 2^9$ кубітів та $c_4 \cdot 2^{147}$ вентилів;
- для геш-функції Купина-512 — $c_5 \cdot 2^{10}$ кубітів та $c_6 \cdot 2^{277}$ вентилів.

Застосування квантового алгоритму Гровера до геш-функції Купина (пошук прообразу):

1. Визначте фіксоване значення гешу $y \in \{0,1\}^n$, для якого шукається прообраз.
2. Використайте оракул O_f, який реалізує перетворення: $O_f|x\rangle = (-1)^{f(x)}|x\rangle$.
3. Створіть квантовий стан $|\psi\rangle = 1/\sqrt{2^m} \sum_{x\in\{0,1\}^m} |x\rangle$.
4. Застосуйте оператор Гровера $G = H^{\oplus n}(2|0\rangle\langle 0| - I)H^{\oplus n}O_f$ до $|\psi\rangle$.
5. Проведіть вимірювання регістру $|x\rangle$ для отримання кандидата на прообраз.
6. Перевірте отримане значення класичним обчисленням $H(x) = y$. У разі невідповідності — повторіть процес.

Алгоритм BHT

У таблиці 1 наведено значення зміних, що необхідні для коректного обрахунку складності застосування алгоритму BHT до Купина-256 та Купина-512.

Таблиця 1. Значення зміних

Змінна	Купина-256	Купина-512
n	256	512
m_h	512	1024
t_h	$13 \cdot 256^2$	$13 \cdot 512^2$
N	2^{256}	2^{512}

Згідно з оцінками складності алгоритму BHT у роботі [4] при його застосуванні до геш-функції Купина оцінки для знаходження прообразу та колізії є приблизно однаковими:

- для геш-функції Купина-256 необхідно: $c_7 \cdot 2^{96}$ кубітів, $c_8 \cdot 2^{147}$ вентилів;
- для геш-функції Купина-512 необхідно: $c_{10} \cdot 2^{182}$ кубітів, 2^{277} вентилів.

Складність за кількістю запитів до оракула становить $c_{11} \cdot 2^{85}$ для геш-функції Купина-256 та $c_{12} \cdot 2^{170}$ для геш-функції Купина-512.

Застосування квантового алгоритму BHT до геш-функції Купина (пошук колізій):

1. Обреріть довільну підмножину $K \subseteq X$, де $|K| = 2^{n/3}$.
2. Обчисліть $H(x)$ для всіх $x \in K$.
3. Побудуйте список L з пар $(x_i, H(x_i))$ та відсортуйте його за другим значенням у списку.
4. Якщо в L знайдено колізію, вивести відповідні пари $\{x_i, x_j\}$.
5. Інакше — побудуйте функцію $f(x)$, що перевіряє, чи існує $x_0 \in K$ таке, що $H(x) = H(x_0)$ і $x \neq x_0$.
6. Застосуйте алгоритм Гровера з оракулом O_f для знаходження x з $X \setminus K$.
7. Знайдіть x_0 з K, $(x_0, H(x_i)) \in L$, для якого $H(x) = H(x_0)$.
8. Поверніть колізію $\{x_0, x\}$.

Застосування алгоритму BHT до геш-функції Купина (пошук прообразу):

1. Задайте значення $h = H(x)$ для деякого невідомого x.
2. Побудуйте функцію $f(x) = [H(x) = h]$.
3. Застосуйте алгоритм Гровера з оракулом O_f.
4. Поверніть знайдене x як прообраз h.

Алгоритм CNS

Для геш-функції Купина з довжиною вихідного гешу n застосування алгоритму CNS для пошуку колізій буде мати складність $O(2^{2n/5})$ відносно кількості запитів. Тобто, для геш-функції Купина-256, де $n = 256$, складність за кількістю запитів до оракула оцінюється як $c_{13} \cdot 2^{102}$, а для геш-функції Купина-512 $(n = 512)$ — $c_{14} \cdot 2^{204}$.

При цьому необхідна кількість кубітів та вентилів:

- для геш-функції Купина-256 — $c_{15} \cdot 2^8$ кубітів та $c_{16} \cdot 2^{119}$ вентилів;
- для геш-функції Купина-512 — $c_{17} \cdot 2^9$ кубітів та $c_{18} \cdot 2^{223}$ вентилів.

Застосування квантового алгоритму CNS для геш-функції Купина (пошук колізій):

1. Оберіть множину $S \subseteq X$ розміру $2^{n/5}$.
2. Обчисліть геш-функцію $H(x)$ для всіх $x \in S$ та збережіть результати у таблиці.
3. Створіть квантовий стан: $|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{x\in\mathcal{X}} |x\rangle$.
4. Використайте оператор Гровера до $|\psi\rangle$ $O(2^{2n/5})$ разів.
5. Проведіть вимірювання стану та перевірте чи існує $x' \in S$ таке, що $H(x) = H(x')$. Якщо існує — колізію знайдено.
6. Поверніть колізію $\{x, x'\}$.

Застосування квантового алгоритму CNS для геш-функції Купина (пошук прообразу до множини гешів):

1. Побудуйте множину $Y = \{h_1, h_2, \ldots, h_k\}$ з можливих цільових гешів.
2. Побудуйте функцію $f(x) = 1$, якщо $H(x) \in Y$.
3. Створіть квантовий стан $|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{x\in\mathcal{X}} |x\rangle$.
4. Застосуйте алгоритм Гровера з оракулом O_f
5. Застосуйте амплітудне підсилення для пошуку x такого, що $H(x) \in Y$.
6. Проведіть вимірювання стану та перевірте чи $H(x) \in Y$. Якщо так, то x є прообразом одного з гешів у Y.
7. Поверніть x як прообраз до деякого $h_i \in Y$.

При застосуванні алгоритму CNS до геш-функції Купина-256, для знаходження прообразу до множини гешів, складність відносно кількості запитів до оракула становить $c_{19} \cdot 2^{109}$, а для геш-функції Купина-512 $(n = 512)$ — $c_{20} \cdot 2^{219}$.

При цьому необхідна кількість кубітів та вентилів:

- для геш-функції Купина-256 — $c_{21} \cdot 2^8$ кубітів та $c_{22} \cdot 2^{219}$ вентилів;

Таблиця 2. Складність квантових алгоритмів для геш-функції Купина

Алго-ритм	**Купина-256**			**Купина-512**		
	запити	кубіти	вентилі	запити	кубіти	вентилі
Гровер (прообраз)	2^{128}	2^{9}	2^{147}	2^{256}	2^{10}	2^{277}
BHT (колізія і прообраз)	2^{85}	2^{96}	2^{147}	2^{170}	2^{182}	2^{277}
CNS (колізія)	2^{102}	2^{8}	2^{119}	2^{204}	2^{9}	2^{223}
CNS (прообраз до множини гешів)	2^{109}	2^{8}	2^{219}	2^{219}	2^{9}	2^{438}

- для геш-функції Купина-512 — $c_{23} \cdot 2^9$ кубітів та $c_{24} \cdot 2^{438}$ вентилів.

У таблиці 2 подано результати усіх обчислень складності квантових алгоритмів при застосуванні до геш-функції Купина у двох режимах: Купина-256 та Купина-512.

Висновки

У роботі здійснено аналіз стійкості геш-функції Купина у квантовій моделі обчислень шляхом застосування трьох квантових алгоритмів: Гровера, Брассара-Хоєра-Таппа (BHT) та Шайу-Наї-Пласенсії-Шроттенльоєра (CNS). Отримано такі результати:

1. Квантовий алгоритм Гровера демонструє найкращу ефективність для задачі пошуку прообразу, завдяки простоті реалізації та найменшим вимогам до ресурсів.
2. Квантовий алгоритм BHT, попри низьку кількість запитів до оракула, виявився найменш придатним для практичного використання і для знаходження колізій, і прообразу. Це пов'язано із значними витратами на реалізацію відносно кількості необхідних кубітів.
3. Квантовий алгоритм CNS є найефективнішим для пошуку колізій, оскільки забезпечує найнижчу асимптотичну складність за всіма показниками серед розглянутих квантових алгоритмів. Також він є ефективним для пошуку прообразу, коли наявні декілька цільових гешів.

Отже, при аналізі стійкості геш-функцій в квантовій моделі обчислень, зокрема для геш-функції Купина, квантовий алгоритм Гровера є рекомендованим для застосування при пошуку прообразу у класичному випадку, коли наявне одне цільове значення. При цьому квантовий алгоритм CNS є більш ефективним за умови наявності декількох цільових гешів для пошуку прообразу. Також алгоритм CNS показує значну перевагу для пошуку колізій порівняно з алгоритмом BHT.

Перелік використаних джерел

1. A New Standard of Ukraine: The Kupyna Hash Function / R. Oliynykov, I. Gorbenko, O. Kazymyrov, V. Ruzhentsev, O. Kuznetsov, Y. Gorbenko, A. Boiko, O. Dyrda, V. Dolgov, A. Pushkaryov. — 2015. — URL: https://eprint.iacr.org/2015/885. Cryptology ePrint Archive, Paper 2015/885.
2. *Nielsen M. A., Chuang I. L.* Quantum Computation and Quantum Information: 10th Anniversary Edition. — USA, 2011.
3. *Brassard G., Høyer P., Tapp A.* Quantum cryptanalysis of hash and claw-free functions. — 06.1997. — DOI: 10.1145/261342.261346. — URL: https://doi.org/10.1145/261342.261346.
4. *Peleshenko L.* Refinement of Quantum Collision Search Algorithms Complexity for General Hash Functions. — 2023. — URL: https://ela.kpi.ua/items/7b24a4a4-38d8-49b0-8a0a-23af410cb6a9.
5. *Chailloux A., Naya-Plasencia M., Schrottenloher A.* An Efficient Quantum Collision Search Algorithm and Implications on Symmetric Cryptography. — 2017. — URL: https://eprint.iacr.org/2017/847. Cryptology ePrint Archive, Paper 2017/847.