Квантовые и классические алгоритмы для задачи о кратчайшей общей надстроке с множественными вхождениями и запросами

К. Р. Хадиев*, Р. И. Капралов‡, А. Р. Иликаев§

Рассматривается задача SUM-SCS: по словарю из n строк построить кратчайшую строку, содержащую не менее k вхождений строк словаря. Изучается версия с множественными запросами (MSUM-SCS): для одного словаря нужно найти длины кратчайших надстрок для m количеств вхождений $k_1, \ldots, k_m$. Рассматривается случай, когда строки словаря не являются подстроками друг друга (бесфакторный случай). Предложены квантовый алгоритм со сложностью $O^*(mn^{1.5} \log k_{\max} + n^{2.5} \log k_{\max} + L)$ и классический со сложностью $O^*(mn^2 \log k_{\max} + n^3 \log k_{\max} + L)$. Здесь L — суммарная длина строк, $k_{\max} = \max_{1 \le r \le m} k_r$, обозначение O^* скрывает не только константу, но и полилогарифмическую функцию, т.е. полином от $\log m, \log n, \log \log k_{\max}, \log L$.

Ключевые слова: строки, квантовые алгоритмы, запросная сложность, кратчайшая общая надстрока.

1. Введение

Задача о кратчайшей общей надстроке (SCS) NP-трудна и связана со сборкой ДНК из коротких фрагментов, но не учитывает многократные вхождения фрагментов. Сформулируем версию задачи с k вхождениями

* *Хадиев Камиль Равилевич* — доцент, Институт вычислительной математики и информационных технологий, Казанский (Приволжский) федеральный университет, e-mail: kamilhadi@gmail.com, ORCID: 0000-0002-5151-9908.

Khadiev Kamil Ravilevich — Assistant professor, Institute of Computational Mathematics and Information Technologies, Kazan Federal University.

‡ *Капралов Руслан Илнарович* — аспирант, Институт вычислительной математики и информационных технологий, Казанский (Приволжский) федеральный университет, e-mail: kapralov_ruslan@mail.ru.

Kapralov Ruslan Ilnarovich — PhD student, Institute of Computational Mathematics and Information Technologies, Kazan Federal University.

§ *Иликаев Артём Рашитович* — аспирант, Институт вычислительной математики и информационных технологий, Казанский (Приволжский) федеральный университет, e-mail: artemka.tema1998@gmail.com.

Ilikaev Artyom Rashitovich — PhD student, Institute of Computational Mathematics and Information Technologies, Kazan Federal University.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

(SUM-SCS) формально. Пусть имеется n строк $S = (s^1, \ldots, s^n)$. Требуется построить кратчайшую строку t (называемую надстрокой) такую, что $\sum_{i=1}^{n} occ(t, s^i) \geq k$, где $occ(t, s^i)$ — это количество вхождений строки s^i в строку t в качестве подстроки. В этой постановке задачи нас интересуют только длина строки и сама строка t.

Рассматривается случай, когда строки словаря не являются подстроками друг друга (бесфакторный случай), известен алгоритм [1] с временной сложностью $O(n^3 \log k + L)$ в случае, если нужно определить длину результирующей строки, и $O(n^3 \log k + L + \ell)$, если нужно найти саму строку, где ℓ — длина результирующей строки. В данной работе предложен квантовый алгоритм со сложностью $O^*(n^{2.5} \log k + L)$ в случае, если нужно найти длину строки, и $O^*(n^{2.5} \log k + L + \ell)$, если нужно найти саму строку.

Для версии задачи с множественными запросами (MSUM-SCS) нужно ответить на m запросов $k_1, \ldots, k_m$ (вариант задачи, когда нужно найти только длину строки). Независимые запуски выполняются со сложностью $O^*(mn^{2.5} \log k_{\max} + L)$ в квантовом случае. В работе показывается, что, используя предварительную обработку, для каждого k_i результат можно посчитать со сложностью $O^*(n^{1.5} \log k_{\max})$.

В рамках данной работы предполагается, что имеется оракул для доступа к символам входной строки и к данным о построенном графе (ребра и вершины). В работе сформулированы оценки запросной сложности в терминах запросов к такому оракулу. В то же время, квантовые алгоритмы, базирующиеся на алгоритме Гровера, имеют временную сложность отличающуюся от запросной на логарифмический множитель [4]. В данном случае, наш алгоритм использует модификации алгоритма Гровера.

2. Квантовый алгоритм для SUM-SCS

Строится полный взвешенный ориентированный граф перекрытий с вершинами $v_1, \ldots, v_n$, где вершина v_i соответствует строке s^i. Вес ребра (v_i, v_j) равен $w(i, j) = |s^j| - r(i, j)$, где $r(i, j)$ — длина наибольшего суффикса s^i, совпадающего с префиксом s^j. При этом суффикс не должен совпадать с самой строкой, т.е. $|s^j| \neq r(i, j)$. Граф строится за $O(L+n^2)$ [3]. Путь $P = (v_{i_1}, \ldots, v_{i_k})$ с повторами задаёт строку длины $|s^{i_1}| + w(P)$, составленную из строк $s^{i_1}, \ldots, s^{i_k}$ с максимальным перекрытием соседних (заданным весом ребер). Здесь $w(P)$ — суммарный вес ребер пути P. Путь, состоящий из k вершин и минимизирующий $|s^{i_1}| + w(P)$, дает решение задачи SUM-SCS в бесфакторной версии. Его можно найти с помощью

матрицы $W^{\boxplus(k-1)}$ — мин-плюс степени матрицы W с показателем $k-1$ (при $k=1$ используется единичная матрица мин-плюс алгебры). Здесь $(A \boxplus B)[i,j] = min\{A[i,q] + B[q,j] : 1 \leq q \leq n\}$ для $1 \leq i,j \leq n$. В работе предлагается выполнить данную операцию с помощью квантового алгоритма поиска минимума [2]. Общая сложность выполнения одной операции в квантовом случае составляет $O^*(n^{2.5})$, а для $O(\log k)$ операций умножения составляет $O^*(n^{2.5}\log k)$, используя технику быстрого возведения в степень. В случае построения самой строки к сложности добавляется слагаемое $O(\ell)$, где ℓ — длина результирующей строки.

3. Алгоритм для MSUM-SCS

В случае задачи MSUM-SCS предварительно считается набор матриц W^{2^p} для $p \in \{0, \ldots, \lceil \log_2 k_{max} \rceil\}$. Они используются для вычислений каждого числа вхождений $k_1, \ldots, k_m$.

Задача решается детерминированным алгоритмом с временной сложностью $O^*(mn^2 \log k_{\max} + n^3 \log k_{\max} + L)$ и квантовым алгоритмом с временной и запросной сложностью $O^*(mn^{1.5} \log k_{\max} + n^{2.5} \log k_{\max} + L)$.

Для малых значений k решение базируется на методе динамического программирования. Пусть $D(h,v)$ — минимальная длина строки для пути из h вершин, оканчивающегося в v, тогда $D(h,v) = \min_u\{D(h-1,u) + W[u,v]\}$. Функция вычисляется один раз до $h = k_{\max}$, а ответ на запрос k_r равен $\min_v D(k_r, v)$. В квантовом случае минимум ищем квантовым алгоритмом поиска минимума [2].

В этом случае временная сложность детерминированного алгоритма составляет $O^*(mn + n^2 k_{\max} + L)$. Временная и запросная сложность квантового алгоритма составляет $O^*(m\sqrt{n} + n^{1.5} k_{\max} + n^2 + L)$.

4. Заключение

Было получено полиномиальное ускорение в квантовых алгоритмах по сравнению с классическими, а также решена более общая задача, чем рассматриваемая ранее.

Финансирование:

Работа выполнена при поддержке РНФ, грант № 25-11-00366.

Список литературы

[1] M. Crochemore, M. Cygan, C. Iliopoulos, M. Kubica, J. Radoszewski, W. Rytter, T. Waleń, "Algorithms for three versions of the shortest common superstring prob-

lem", *Combinatorial Pattern Matching (CPM 2010)*, Lecture Notes in Computer Science, **6129**, Springer, 2010, 299–309. DOI: 10.1007/978-3-642-13509-5_27.

[2] C. Dürr, M. Heiligman, P. Høyer, M. Mhalla, "Quantum query complexity of some graph problems", *SIAM Journal on Computing*, **35**:6 (2006), 1310–1328. DOI: 10.1137/050644719.

[3] D. Gusfield, G. M. Landau, B. Schieber, "An efficient algorithm for the all pairs suffix-prefix problem", *Information Processing Letters*, **41**:4 (1992), 181–185. DOI: 10.1016/0020-0190(92)90176-v.

[4] L. Grover, "Trade-offs in the quantum search algorithm", *Physical Review A*, **66**:5 (2002), 052314. DOI: 10.1103/PhysRevA.66.052314.

Статья поступила 9 ноября 2026 г.

Quantum and classical algorithms for the multiple query version of the uniform multiple occurrences shortest common superstring problem

K. R. Khadiev, R. I. Kapralov, A. R. Ilikaev

We consider the SUM-SCS problem: given a dictionary of n strings, construct a shortest string containing at least k occurrences of dictionary strings. We study the multiple query version (MSUM-SCS): for a fixed dictionary, find the lengths of shortest superstrings for m values $k_1, \ldots, k_m$. We consider the factor-free case when non of the strings can be substring of any another. We present a quantum algorithm with $O^*(mn^{1.5} \log k_{\max} + n^{2.5} \log k_{\max} + L)$ complexity, a classical one with $O^*(mn^2 \log k_{\max} + n^3 \log k_{\max} + L)$ complexity. Here L is the total length of the strings, $k_{\max} = \max_{1 \le r \le m} k_r$, and O^* notation hides not only a constant, but also polylogarithmic multiple, i.e. a polynomial on $\log m, \log n, \log \log k_{\max}, \log L$.

Keywords: strings, quantum algorithms, query complexity, shortest common superstring.

Received on November 9, 2026