Nearly Optimal Circuit Size for Sparse Quantum State Preparation

Lvzhou Li
Institute of Quantum Computing and Software, School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China

Jingquan Luo
Institute of Quantum Computing and Software, School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China

Abstract

Quantum state preparation is a fundamental and significant subroutine in quantum computing. In this paper, we conduct a systematic investigation on the circuit size (the total count of elementary gates in the circuit) for sparse quantum state preparation. A quantum state is said to be d-sparse if it has only d non-zero amplitudes. For the task of preparing an n-qubit d-sparse quantum state, we obtain the following results:

- **Without ancillary qubits:** Any n-qubit d-sparse quantum state can be prepared by a quantum circuit of size $O(\frac{nd}{\log n}+n)$ without using ancillary qubits, which improves the previous best results. It is asymptotically optimal when $d=\text{poly}(n)$, and this optimality holds for a broader scope under some reasonable assumptions.
- **With limited ancillary qubits:** (i) Based on the first result, we prove for the first time a trade-off between the number of ancillary qubits and the circuit size: any n-qubit d-sparse quantum state can be prepared by a quantum circuit of size $O(\frac{nd}{\log(n+m)}+n)$ using m ancillary qubits for any $m\in O(\frac{nd}{\log nd}+n)$. (ii) We establish a matching lower bound $\Omega(\frac{nd}{\log(n+m)}+n)$ under some reasonable assumptions, and obtain a slightly weaker lower bound $\Omega(\frac{nd}{\log(n+m)+\log d}+n)$ without any assumptions.
- **With unlimited ancillary qubits:** Given arbitrary amount of ancillary qubits available, the circuit size for preparing n-qubit d-sparse quantum states is $\Theta(\frac{nd}{\log nd}+n)$.

2012 ACM Subject Classification Theory of computation → Circuit complexity; Theory of computation → Quantum computation theory

Keywords and phrases Quantum computing, quantum state preparation, circuit complexity

Funding Lvzhou Li and Jingquan Luo were supported by the National Key Research and Development Program of China (Grant No.2024YFB4504004), the National Natural Science Foundation of China (Grants No.92465202 and No.62272492), the Guangdong Provincial Quantum Science Strategic Initiative (Grant No.GDZX2303007), and the Guangzhou Science and Technology Program (Grant No.2024A04J4892).

1 Introduction

Since the inception of quantum computing [14], an increasing number of quantum algorithms have emerged, exhibiting acceleration advantages over classical algorithms. Notable examples include Shor's algorithm [54], Grover's algorithm [20], HHL algorithm [22], and Hamiltonian simulation algorithms [10, 31, 32, 6], as well as quantum machine learning algorithms [30, 29, 28, 27, 46]. Within these algorithms, the preparation of quantum states plays a pivotal role. Usually, the first and inevitable step to process classical data by quantum algorithms is to encode the data into quantum states. If this step consumes substantial resources, it may offset the acceleration advantages of quantum algorithms. Several works have indicated that the substantial accelerations achieved by quantum machine learning stem from the underlying input assumptions [56, 9, 57, 8, 16]. Therefore, how to efficiently prepare quantum states is a fundamental and significant issue in the field of quantum computing.

Before delving into the discussion of quantum state preparation, it is essential to specify the elementary gate set and complexity metrics. In this paper, we adopt the most common gate set, which consists of single-qubit gates and CNOT gates, enabling the accurate implementation of any unitary transformation [3]. Additionally, for convenience, the Toffoli gate is permitted, which can be constructed using 10 single-qubit gates and 6 CNOT gates [41]. Various standards can be employed to gauge the efficiency of a quantum circuit, including size (the total count of elementary gates), depth (the number of layers such that gates in the same layer can be run in parallel), space (the number of ancillary qubits), and more. Given the considerably higher implementation cost of CNOT gates compared to single-qubit gates, the count of CNOT gates is also utilized as a measure of algorithmic efficiency.

When the quantum state to be prepared does not possess any structural features, we refer to it as the general quantum state preparation problem. This problem has already been extensively researched, and current preparation algorithms have achieved the optimal circuit size $\Theta(2^n)$ [19, 51, 5, 43, 24]. Given the high cost of preparing general quantum states and the fact that, in practical applications, data often exhibits special structures, there has been considerable literature focusing on the preparation of special quantum states. These include states whose amplitudes are given by a continuous function [23, 18, 45, 35, 37, 39], states whose amplitudes are accessed through a black-box oracle [50, 4, 59], and states under the low-rank assumption [2]. Another class of quantum states that holds both practical and theoretical significance is sparse quantum states [17, 33, 44, 40, 13, 12, 63, 55, 34].

An n-qubit d-sparse quantum state refers to an n-qubit quantum state with d non-zero amplitudes. Given two positive integers n and $0 < d \leq 2^n$, along with a set

$$\mathcal{P} = \{(\alpha_i, q_i)\}_{0\leq i\leq d-1} \tag{1}$$

such that $\alpha_i \in \mathbb{C}$, $q_i \in \{0,1\}^n$ for all $0 \leq i \leq d-1$, $\sum_i |\alpha_i|^2 = 1$ and $q_i \neq q_j$ for all $i \neq j$, the aim of sparse quantum state preparation (SQSP) is to generate a quantum circuit which acts on $n + m$ qubits and implements a unitary operator U satisfying

$$U\left|0\right\rangle^{\otimes n}\left|0\right\rangle^{\otimes m} = \sum_{i=0}^{d-1} \alpha_i \left|q_i\right\rangle \left|0\right\rangle^{\otimes m}, \tag{2}$$

where $m \geq 0$ is an integer and the last m qubits serve as ancillary qubits.

The importance of SQSP is self-evident, and the motivation is twofold:

- First, many practically relevant states in quantum computing and processing have the property that only a small proportion of the basis states have nonzero coefficients. Some prominent examples of sparse states are W states, GHZ states, generalized Bell states, and thermofield double states [11].
- Second, SQSP offers a finer-grained perspective on quantum state preparation. For the general quantum state preparation problem, the circuit size is proven to be $\Theta(2^n)$, which does not consider the dependence on the parameter d. In other words, it simply assumes $d = 2^n$. A more fine-grained complexity should consider this dependence, and recover the general case when $d = 2^n$.

1.1 Contributions

In this paper, we conduct a systematic investigation of the circuit size of sparse quantum state preparation, exploring the three scenarios: without, with limited, and with unlimited ancillary qubits. The complexity notations O, Ω, o and ω will be explained in detail in Section 2.

1.1.1 SQSP without Ancillary Qubits

First, we consider the scenario of not using ancillary qubits.

▶ **Theorem 1.** *Any n-qubit d-sparse quantum state can be prepared by a quantum circuit of size $O(\frac{nd}{\log n}+n)$ without using ancillary qubits.*

Previously, the best result without using ancillary qubits [17, 33] can only achieve the circuit size $O(nd)$. We demonstrate for the first time that any n-qubit d-sparse quantum state can be prepared by a circuit of size $o(nd)$ without using ancillary qubits. Moreover, in Theorem 7 we will see that the upper bound $O(\frac{nd}{\log n}+n)$ is asymptotically optimal when $d=\mathrm{poly}(n)$, and in Theorem 4 we get the lower bound $\Omega(\frac{nd}{\log n}+n)$ when $m=0$, which indicates that the upper bound is asymptotically optimal when $d\le 2^{\delta n}$ for a sufficiently small constant $\delta\in(0,1)$ under reasonable assumptions.

1.1.2 SQSP with m Ancillary Qubits

Next, we consider the scenario of using m ancillary qubits. To the best of our knowledge, we are the first to consider and demonstrate the trade-off between the number of ancillary qubits and the circuit size of SQSP. Based on Theorem 1, we obtain the following theorem:

▶ **Theorem 2.** *For any $m\in O(\frac{nd}{\log nd}+n)$, any n-qubit d-sparse quantum state can be prepared by a quantum circuit of size $O(\frac{nd}{\log(m+n)}+n)$ using m ancillary qubits.*

▶ Remark 3. In Theorem 2, when $m=\Theta(\frac{nd}{\log nd}+n)$, we get $O(\frac{nd}{\log nd}+n)$ on the circuit size. Actually, more ancillary qubits than $\Theta(\frac{nd}{\log nd}+n)$ are useless for reducing the upper bound $O(\frac{nd}{\log nd}+n)$. Otherwise, it would contradict the fact that the number of ancillary qubits must be asymptotically less than or equal to the circuit size. Since only one-qubit gates and CNOT gates are considered as elementary quantum gates here, if the number of ancillary qubits is more than twice as much as the circuit size, there must exist unused ancillary qubits. Therefore, without loss of generality, we can assume that $m\in O(\frac{nd}{\log nd}+n)$.

We demonstrate that the upper bound established in Theorem 2 is tight under reasonable assumptions. Notice that all sparse quantum state preparation methods proposed so far use at most $O(d)$ arbitrary-angle single-qubit rotation gates (i.e., $R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$). For example, in permutation-based algorithms [33, 44], arbitrary-angle single-qubit rotation gates are only used in the first step to prepare a $\lceil\log d\rceil$-qubit quantum state, while in the second step, implementing a permutation only involves CNOT gates and $O(n)$ specific types of single-qubit gates. We believe that using $\omega(d)$ arbitrary-angle single-qubit rotation gates is pointless.

▶ **Theorem 4.** *Suppose $d\le 2^{\epsilon n}$ for a sufficiently small constant $\epsilon\in(0,1)$, and given m ancillary qubits available, if an algorithm $\mathcal{A}$ for preparing n-qubit d-sparse quantum states satisfies the following conditions:*

1. *$\mathcal{A}$ uses at most $O(d)$ single-qubit rotation gates with arbitrary angles,*
2. *all other single-qubit gates amount to a total of $O(n)$ types,*

then $\mathcal{A}$ requires $\Omega(\frac{nd}{\log(m+n)}+n)$ elementary quantum gates in the worst case.

More specifically, excluding rotation gates with arbitrary angles, the single-qubit gates considered in Theorem 4 include NOT gates and Phase($\pm\pi/2^j$) gates for $1 \le j \le n$, where the latter are useful for implementing an n-Toffoli gate without ancillae [15].

▶ Remark 5. When only $m = \mathrm{poly}(n)$ ancillary qubits are available, the lower bound in Theorem 4 is $\Omega(\frac{nd}{\log n} + n)$, which can be achieved without ancillary qubits as shown in Theorem 1. Therefore, poly(n) ancillary qubits do not help in reducing the circuit size of SQSP under some reasonable assumptions.

Without any assumptions, we prove a slightly weaker lower bound on the circuit size as follows.

▶ **Theorem 6.** *Given m ancillary qubits available, there exist n-qubit d-sparse quantum states such that any algorithm to prepare them requires $\Omega(\frac{nd}{\log(m+n)+\log d} + n)$ elementary quantum gates.*

1.1.3 SQSP with Unlimited Ancillary Qubits

We provide a complete characterization of the circuit size for SQSP in the case where an unlimited number of ancillary qubits is available.

▶ **Theorem 7.** *With unlimited number of ancillary qubits available, the circuit size for preparing n-qubit d-sparse quantum states is $\Theta(\frac{nd}{\log nd} + n)$. Furthermore, if $d = \mathrm{poly}(n)$, then the circuit size is $\Theta(\frac{nd}{\log n} + n)$, which can be achieved without using ancillary qubits.*

Proof. According to Remark 3, it is sufficient to assume $m = \Theta(\frac{nd}{\log nd} + n)$ ancillary qubits available. Therefore, by setting $m = \Theta(\frac{nd}{\log nd} + n)$ in Theorem 6, we get the lower bound $\Omega(\frac{nd}{\log nd} + n)$, which can be achieved by Theorem 2.

Furthermore, if $d = \mathrm{poly}(n)$, then $\Theta(\frac{nd}{\log nd} + n)$ becomes $\Theta(\frac{nd}{\log n} + n)$, which can be achieved without using ancillary qubits as shown in Theorem 1. ◀

1.2 Proof Techniques

1.2.1 SQSP without Ancillary Qubits

The idea of SQSP without ancillary qubits proposed in this paper aligns with prior research [33, 44]. Given the state description $\mathcal{P} = \{(\alpha_i, q_i)\}_{0\le i\le d-1}$, initially a dense $\lceil \log d\rceil$-qubit quantum state $\sum_i \alpha_i |\sigma^{-1}(q_i)\rangle$ is prepared, where σ is some permutation such that $0 \le \sigma^{-1}(q_i) \le d-1$. Subsequently, this state is transformed into the target quantum state $\sum_i \alpha_i |q_i\rangle$ by applying σ. The dense quantum state $\sum_i \alpha_i |\sigma^{-1}(q_i)\rangle$ can be prepared with a circuit of size $O(d)$. Therefore, the primary contribution to circuit size arises from the second step, i.e., the implementation of the permutation σ. Prior works [33, 44] have adopted circuits of size $O(nd)$ for implementing σ, which results in an overall circuit size of $O(nd)$. In comparison, we develop a method for implementing σ with a size of $O(\frac{nd}{\log n} + n)$, as detailed in Lemma 8.

▶ **Lemma 8.** *For any permutation $\sigma \in \mathfrak{S}_{2^n}$, there exists a quantum circuit implementing σ of size $O(\frac{n\,\mathrm{size}(\sigma)}{\log n} + n \log\min\{\mathrm{size}(\sigma), \log n\})$ without using ancillary qubits.*

In the above, $\mathfrak{S}_{2^n}$ denotes the set of all permutations on $\{0,1\}^n$, and size(σ) denotes the number of non-fixed points of a permutation σ (see the formal definition in Equation (5)). Reversible circuits, which implement permutations, have been extensively studied in the literature [53, 7, 60, 49, 48, 47, 1, 62, 61, 36, 25]. Shende et al. [53] proposed a synthesis method for realizing a permutation $\sigma \in \mathfrak{S}_n$ using a circuit of size $O(n2^n)$; a similar result was obtained in [7]. Saeedi et al. [49, 48] introduced cycle-based methods that reduced the circuit size in practice, although the asymptotic complexity remained unchanged. Significant improvements were later made independently by Zakablukov [62] and Wu and Li [61], who achieved an asymptotically optimal circuit size of $O(\frac{n2^n}{\log n})$. However, Refs. [62, 61] did not consider the sparsity of permutations[1]. Lemma 8 improves upon the results in [62, 61], and plays a key role in reducing the circuit size from $O(nd)$ to $O(\frac{nd}{\log n}+n)$ in Theorem 1. We believe this result is of independent interest and may find applications in other contexts.

1.2.2 SQSP with m Ancillary Qubits

To further reduce the circuit size with ancillary qubits, we introduce a new framework for SQSP that is totally different from the approach without ancillary qubits. Specifically, we employ a novel integer encoding method, denoted as the (n,r)-unary encoding (for the formal definition, see Definition 15). The new framework is roughly divided into two steps: firstly, prepare an intermediate state using the (n,r)-unary encoding, and secondly transform the intermediate state to a state in the binary encoding. Intuitively, the (n,r)-unary encoding divides an n-bit integer to $\frac{n}{r}$ continuous parts, each of length r, and represents each part using the corresponding unary encoding. When $r=n$, the (n,r)-unary encoding recovers to the standard unary encoding. This integer representation method was previously introduced in [26], where the authors showed that a general quantum state using the (n,r)-unary encoding can be prepared by a quantum circuit of size $O(2^n)$ and depth $O(r2^{n-r})$ with $\frac{n2^r}{r}$ qubits. However, they focused on reducing the circuit depth and did not account for the sparsity of the quantum state. Also, their technique does not apply to SQSP. We demonstrate that a sparse quantum state using the (n,r)-unary encoding can be efficiently prepared.

▶ **Lemma 9.** *Given two positive integers n and r such that r divides n, and a set of size d $\{(q_i,\alpha_i)\}_{0\le i\le d-1}$ such that $\sum_i|\alpha_i|^2=1$, $q_i\in\{0,1\}^n$ for all $0\le i\le d-1$ and $q_i\neq q_j$ for any $i\neq j$, there exists a quantum circuit preparing the following $\frac{n2^r}{r}$-qubit state*

$$\sum_{i=0}^{d-1}\alpha_i\left|e_{q_i(n-r,n)}\right\rangle\left|e_{q_i(n-2r,n-r)}\right\rangle\dots\left|e_{q_i(0,r)}\right\rangle=\sum_{i=0}^{d-1}\alpha_i\left(\bigotimes_{j=0}^{\frac{n}{r}-1}\left|e_{q_i(jr,(j+1)r)}\right\rangle\right). \tag{3}$$

The circuit is of size $O(\frac{nd}{r})$, using one ancillary qubit.

This state using the (n,r)-unary encoding serves as an intermediate form, ultimately transformed into the binary encoding. We believe that this intermediate state will also have applications in other tasks.

1.2.3 Lower Bound for Circuit Size

Prior proofs of lower bounds for quantum circuit size relied on dimensionality arguments [52, 43], resulting in a trivial lower bound of $\Omega(d)$ when applied to the sparse quantum state preparation problem. Here we employ the counting argument. The challenge in applying this method to establish lower bounds for quantum circuit size lies in the existence of an infinite number of single-qubit quantum gates, i.e., the parameters of $\{R_x(\theta),R_y(\theta),R_z(\theta)\}$ are continuous. The fundamental idea in the proof is to discretize the parameters of the

[1] The sparsity of a permutation $\sigma\in\mathfrak{S}_{2^n}$ is measured by the quantity size(σ), which equals 2^n in the worst case, but can be significantly smaller in many practical applications.

single-qubit quantum gates. Specifically, we consider approximately preparing sparse states in the following set to sufficient precision, ensuring that the resulting states are distinct from one another:

$$\mathcal{D}_d \coloneqq \left\{ |\psi_{\mathcal{S}}\rangle \coloneqq \frac{1}{\sqrt{d}} \sum_{i\in\mathcal{S}} |i\rangle \mid \mathcal{S} \subset \{0,\dots,2^n-1\}, |\mathcal{S}| = d \right\}. \tag{4}$$

To achieve this, we only need to discretize the single-qubit gates to a certain precision, and we demonstrate that this discretization does not increase the circuit size. Consequently, we can derive a lower bound on the circuit size for SQSP based on the lower bound for the task of approximately preparing the states in $\mathcal{D}_d$ using the discretized single-qubit gates and CNOT gates. We hope the discretization technique will find more applications in proving lower bounds on the circuit size for other subclasses of quantum states.

1.3 Related Work

The problem of SQSP has garnered significant attention [17, 33, 44, 40, 13, 12, 63, 55, 34]. Gleinig and Hoefler [17] were the first to address this problem and presented an algorithm generating a quantum circuit of size $O(nd)$ without using ancillary qubits for any n-qubit d-sparse quantum state. Subsequently, Malvetti et al. [33] and Ramacciotti et al. [44] presented permutation-based preparation algorithms, achieving the same circuit size using 0 and 1 ancillary qubits, respectively. de Veras et al. [13] improved upon the ideas in [42, 58] and introduced a preparation algorithm based on quantum state splitting, achieving the same circuit size by employing 2 ancillary qubits. They further optimized this algorithm [12], presenting a circuit size depending on the Hamming weights of the basis states with non-zero amplitudes: $O(\sum_i |q_i|)$. In the worst case, the circuit size remains $O(nd)$. Mozafari et al. [40] employed decision diagrams to represent quantum states and introduced an algorithm with circuit size $O(kn)$, using 1 ancillary qubit, where k denotes the number of paths in the decision diagram. For a d-sparse quantum state, as the number of paths in its associated decision diagram satisfies $k \le d$, the circuit size of this method in the worst-case remains $O(nd)$. In summary, the circuit sizes of all the aforementioned algorithms are $O(nd)$. Recently, Mao et al. [34] propose a sparse state preparation algorithm which generates circuits of size $O(\frac{nd}{\log n} + n)$ with two ancillary qubits, improving the previous bound $O(nd)$.

In addition to optimizing circuit size, Zhang et al. [63] focused on reducing circuit depth. They showed that with $O(nd \log d)$ ancillary qubits and $O(nd \log d)$ gates[2], any n-qubit d-sparse quantum state can be prepared by a circuit of depth $O(\log nd)$. They also proved a matching lower bound of $\Omega(\log nd)$ on the depth. Sun et al. [55] focused on circuit depth for general quantum state preparation, and noted that their method can also be extended to sparse states, yielding a depth complexity of $O(n \log nd + \frac{nd^2 \log d}{n+m})$ with m ancillary qubits. This was later improved to $O(\frac{nd \log d \log m}{m} + \log nd)$ in [64].

1.4 Discussion

We achieve a relatively complete understanding of the circuit size for SQSP in this paper; however, several interesting problems remain open and are worthy of further discussion:

[2] The circuit size was not explicitly stated in [63], but can be inferred from the algorithm presented therein.

- In the scenario of using m ancillary qubits, can we prove a matching lower bound $O(\frac{nd}{\log(m+n)}+n)$ on the circuit size without assumptions?
- In this paper, all of our results focus on exact preparation, meaning the precise preparation of the states. What about the approximate preparation of d-sparse quantum states?
- What is the optimal trade-off between the number of ancillary qubits and the circuit depth for SQSP? To the best of our knowledge, the only known results are a rough upper bound of $O(n\log nd+\frac{nd^2\log d}{n+m})$ mentioned in [55], and a refined bound of $O(\frac{nd\log d\log m}{m}+\log nd)$ given in [64].

1.5 Organization

In Section 2, we recall some notations. Section 3 is devoted to the proof of Theorem 1. In Section 4, we prove Theorems 2, 4 and 6. Some conclusions are made in Section 5.

2 Preliminaries

In this paper, all logarithms are base 2. A permutation σ on a finite set A is a bijective function $\sigma\colon A\to A$. Given a positive integer n, we denote by $\mathfrak{S}_{2^n}$ the set of all permutations on $\{0,1\}^n$, and the permutations considered in this paper all belong to $\mathfrak{S}_{2^n}$. A cycle $(a_1,a_2,\ldots,a_k)$ is a permutation σ such that $\sigma(a_1)=a_2$, $\sigma(a_2)=a_3$, $\ldots$, and $\sigma(a_k)=a_1$. The length of a cycle is the number of elements it contains. A cycle of length two is called a transposition. A cycle of length k is called a k-cycle. The composition of two permutations σ_1 and σ_2 is denoted by $\sigma_2\circ\sigma_1$ where the right one is applied first. The composition operation is typically not commutative, but when two permutations are disjoint, we can interchange them.

Any permutation can be decomposed as a composition of a finite number of transpositions. A permutation is even (odd) if it can be written as a composition of an even (odd) number of transpositions.

▶ **Lemma 10** ([21]). *Any permutation can be written as the composition of a finite number of pairwise disjoint cycles.*

A fixed point of a permutation σ is an element $x\in\{0,1\}^n$ satisfying $\sigma(x)=x$. For any permutation σ, let size(σ) denote the number of non-fixed points of σ:

$$\mathrm{size}(\sigma):=|\{x\in\{0,1\}^n\mid\sigma(x)\neq x\}|. \tag{5}$$

Here we give a simple example to show the notions given above. Consider the permutation $\sigma\in\mathfrak{S}_8$ given as follows:

$$\begin{aligned}&\sigma(0)=1, &&\sigma(1)=5,\sigma(2)=4,\sigma(3)=3,\\ &\sigma(4)=2, &&\sigma(5)=7,\sigma(6)=6,\sigma(7)=0.\end{aligned}$$

σ could be written as

$$\begin{aligned}\sigma&=(0,1,5,7)\circ(2,4)\circ(3)\circ(6) &&(6)\\ &=(0,1,5,7)\circ(2,4) &&(7)\\ &=(0,5)\circ(0,1)\circ(5,7)\circ(2,4). &&(8)\end{aligned}$$

In Equation (6), σ is written as the composition of pairwise disjoint cycles, where $(0,1,5,7)$ is a 4-circle and $(2,4)$ is a transposition. In Equation (7), we omit 1-circles. In Equation (8),

σ is decomposed as the composition of 5 transpositions, thus σ is an odd permutation. The number of non-fixed points of σ is $\text{size}(\sigma) = 6$.

An n-qubit Toffoli gate is a quantum gate acting on n qubits and applying an X gate on the last qubit when the preceding $n-1$ qubits are all in the state $|1\rangle$. Gidney [15] demonstrated the following lemma in his well-known blog:

▶ **Lemma 11** ([15]). *Any n-qubit Toffoli gate can be implemented by a quantum circuit of size $O(n)$ without ancillary qubits.*

We briefly introduce several complexity notations. Given two functions $f(n)$ and $g(n)$ defined on $\mathbb{N}$:

- $f(n) = O(g(n))$ if there exist positive constants c and n_0 such that $f(n) \le c \cdot g(n)$ for all $n \ge n_0$.
- $f(n) = \Omega(g(n))$ if there exist positive constants c and n_0 such that $f(n) \ge c \cdot g(n)$ for all $n \ge n_0$.
- $f(n) = \Theta(g(n))$ if both $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.
- $f(n) = o(g(n))$ if, for any positive constant $c > 0$, there exists a constant n_0 such that $f(n) \le c \cdot g(n)$ for all $n \ge n_0$.
- $f(n) = \omega(g(n))$ if, for any positive constant $c > 0$, there exists a constant n_0 such that $f(n) \ge c \cdot g(n)$ for all $n \ge n_0$.

3 SQSP without Ancillary Qubits

In this section, we present a sparse quantum state preparation algorithm without using ancillary qubits. The algorithm presented here relies on the efficient implementation of a permutation $\sigma \in \mathfrak{S}_n$. As mentioned earlier, the circuit size for implementing σ in prior works [33, 44] is $O(n\,\text{size}(\sigma))$. Improving upon the results of [62, 61], we show in the following lemma that this bound can be further reduced.

▶ **Lemma 12** (Restatement of Lemma 8). *For any permutation $\sigma \in \mathfrak{S}_{2^n}$, there exists a quantum circuit implementing σ of size $O(\frac{n\,\text{size}(\sigma)}{\log n} + n \log \min\{\text{size}(\sigma), \log n\})$ without using ancillary qubits.*

Proof. The basic steps to implement σ are as follows:

- **Step 1**: Decompose σ into the composition of as few permutations σ_i as possible, where each permutation σ_i consists of at most m pairwise disjoint transpositions and satisfies that $\text{size}(\sigma_i)$ is a power of 2. m is a parameter to be determined, and will be a power of 2.
- **Step 2**: Assume $\sigma_i = (x_0, x_1) \circ (x_2, x_3) \circ \cdots \circ (x_{2m-2}, x_{2m-1})$ and implement each σ_i successively by the following steps:
  - **Step 2a**: Execute the permutation $\sigma_{i,1}$ satisfying the condition: for any $0 \le j \le 2m-1$, $\sigma_{i,1}(x_j) = j$;
  - **Step 2b**: Execute the permutation $\sigma_{i,2} = (0,1) \circ (2,3) \circ \cdots \circ (2m-2, 2m-1)$;
  - **Step 2c**: Execute the inverse of permutation $\sigma_{i,1}$.

Below, we illustrate how to implement this process, set the parameter m, and analyze the circuit size.

In Step 1, we aim to decompose σ into a composition of at most

$$M = \left\lfloor \frac{\text{size}(\sigma)}{m} \right\rfloor + O(\log(\min\{m, \text{size}(\sigma)\})) \tag{9}$$

permutations $\{\sigma_i \mid 0 \le i \le M-1\}$, i.e., $\sigma = \sigma_{M-1} \circ \cdots \circ \sigma_0$, where each σ_i consists of at most m pairwise disjoint transpositions, and $\text{size}(\sigma_i)$ is a power of 2. Firstly, by Lemma 10, we decompose σ into the composition of K disjoint cycles for some integer $K \geq 0$, i.e., $\sigma = \rho_{K-1} \circ \cdots \circ \rho_0$, where each ρ_k is an r_k-cycle for some integer $r_k > 0$. It is clear that $\sum_k r_k = \mathrm{size}(\sigma)$. Each r_k-cycle can be decomposed into two sets of disjoint transpositions, each of size either $\lfloor \frac{r_k}{2} \rfloor$ or $\lfloor \frac{r_k}{2} \rfloor - 1$ [38]. For example, suppose $\rho = (x_0, x_1, \ldots, x_{2k-1})$ is a cycle of even length. Then we have $\rho = \rho'' \circ \rho'$, where

$$\rho' = (x_0, x_{2k-1}) \circ (x_1, x_{2k-2}) \circ \cdots \circ (x_{k-1}, x_k), \tag{10}$$

$$\rho'' = (x_1, x_{2k-1}) \circ (x_2, x_{2k-2}) \circ \cdots \circ (x_{k-1}, x_{k+1}). \tag{11}$$

A similar construction applies when ρ is an odd-length cycle. Thus, σ can be decomposed into two sets of disjoint transpositions, each of size at most $\lfloor \frac{\mathrm{size}(\sigma)}{2} \rfloor$. Each such set can be further partitioned into at most $\lfloor \frac{\mathrm{size}(\sigma)}{2m} \rfloor + \lceil \log(\min\{m, \mathrm{size}(\sigma)\}) \rceil$ sets of disjoint transpositions, where each set contains at most m transpositions and the number of transpositions is a power of 2.

In Step 2, we show each σ_i can be efficiently implemented. To avoid introducing more parameters, in the following we assume that $\sigma_i = (x_0, x_1) \circ (x_2, x_3) \circ \cdots \circ (x_{2m-2}, x_{2m-1})$ is composed of m pairwise disjoint transpositions. Step 2 is divided into three substeps. First, we illustrate how to implement a permutation $\sigma_{i,1}$ that satisfies $\sigma_{i,1}(x_j) = j$ for any $0 \leq j \leq 2m-1$. Note that x_j is an integer represented by n bits. We treat x_j as an n-dimensional row vector $x_j = (x_{j,0}, x_{j,1}, \ldots, x_{j,n-1})$ satisfying $x_j = \sum_{k=0}^{n-1} x_{j,k} 2^k$, where $x_{j,k}$ represents the k-th bit and the bits are arranged from the least significant bit to the most significant bit, contrary to the usual convention. We construct a matrix composed of x_j to track the changes in x_j:

$$A = \begin{pmatrix} x_0 \\ x_1 \\ \ldots \\ x_{2m-2} \\ x_{2m-1} \end{pmatrix} = \begin{pmatrix} x_{0,0} & x_{0,1} & \cdots & x_{0,n-2} & x_{0,n-1} \\ x_{1,0} & x_{1,1} & \cdots & x_{1,n-2} & x_{1,n-1} \\ \vdots & & \vdots & & \vdots \\ x_{2m-2,0} & x_{2m-2,1} & \cdots & x_{2m-2,n-2} & x_{2m-2,n-1} \\ x_{2m-1,0} & x_{2m-1,1} & \cdots & x_{2m-1,n-2} & x_{2m-1,n-1} \end{pmatrix} \tag{12}$$

Now we introduce additional conditions to m: suppose m is a power of 2 and satisfies $2m \leq \log n$. With these conditions, realizing the permutation $\sigma_{i,1}$ is equivalent to transforming the matrix A into another matrix $\widetilde{A}$ using elementary gates:

$$\widetilde{A} = \left(\begin{array}{ccccccc} \multicolumn{4}{c}{\overbrace{\hphantom{0\quad 0\quad \ldots\quad 0}}^{\log 2m}} & \multicolumn{3}{c}{\overbrace{\hphantom{0\quad \ldots\quad 0}}^{n-\log 2m}} \\ 0 & 0 & \ldots & 0 & 0 & \ldots & 0 \\ 1 & 0 & \ldots & 0 & 0 & \ldots & 0 \\ \vdots & & & \vdots & \vdots & & \\ 0 & 1 & \ldots & 1 & 0 & \ldots & 0 \\ 1 & 1 & \ldots & 1 & 0 & \ldots & 0 \end{array}\right) \tag{13}$$

Suppose that the matrix A has ℓ distinct non-zero columns. Since the matrix A has $2m$ distinct rows, we have $\log 2m \leq \ell \leq 2^{2m} - 1$. When the j-th column is identical to the k-th column, i.e., $x_{i,j} = x_{i,k}$ for all $0 \leq i \leq 2m-1$, and they are not all-zero columns, we apply, without loss of generality, a CNOT gate with the j-th qubit as the control qubit and the k-th qubit as the target qubit, transforming the k-th column into an all-zero column. This process is repeated until the matrix no longer contains identical non-zero columns. This step requires at most $n - \ell$ CNOT gates. Next, by using at most ℓ swap gates (each of which can be implemented with 3 CNOT gates), we swap the ℓ non-zero columns to the first ℓcolumns, obtaining matrix A_1:

$$A_1 = \left(\begin{array}{ccccccc} \multicolumn{4}{c}{\overbrace{\hspace{10em}}^{\ell}} & \multicolumn{3}{c}{\overbrace{\hspace{3em}}^{n-\ell}} \\ a_{0,0} & a_{0,1} & \cdots & a_{0,\ell} & 0 & \cdots & 0 \\ a_{1,0} & a_{1,1} & \cdots & a_{1,\ell} & 0 & \cdots & 0 \\ \vdots & & & \vdots & & \vdots & \\ a_{2m-2,0} & a_{2m-2,1} & \cdots & a_{2m-2,\ell} & 0 & \cdots & 0 \\ a_{2m-1,0} & a_{2m-1,1} & \cdots & a_{2m-1,\ell} & 0 & \cdots & 0 \end{array}\right) \tag{14}$$

Next, we proceed to transform each row of matrix A_1 step by step, gradually converting it to matrix $\widetilde{A}$. We start with the first row (indexed as row 0). If $a_{0,k} = 1$ for any $0 \leq k \leq n-1$, we apply an X gate to the k-th qubit. This step requires at most ℓ X gates. Upon completing this step, matrix A_1 becomes matrix A_2:

$$A_2 = \left(\begin{array}{ccccccc} \multicolumn{4}{c}{\overbrace{\hspace{10em}}^{\ell}} & \multicolumn{3}{c}{\overbrace{\hspace{3em}}^{n-\ell}} \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ b_{1,0} & b_{1,1} & \cdots & b_{1,\ell} & 0 & \cdots & 0 \\ \vdots & & & \vdots & & \vdots & \\ b_{2m-2,0} & b_{2m-2,1} & \cdots & b_{2m-2,\ell} & 0 & \cdots & 0 \\ b_{2m-1,0} & b_{2m-1,1} & \cdots & b_{2m-1,\ell} & 0 & \cdots & 0 \end{array}\right) \tag{15}$$

Suppose the rows indexed as 0 to $j-1$ have been successfully transformed, we turn our attention to row indexed as j, which can be divided into two scenarios:

- In the case where there exists an element $b_{j,k} \neq 0$ in row j with $k > \log 2m$, for each $0 \leq k' < \ell$ satisfying $k' \neq k$ and $b_{j,k'} \neq \widetilde{A}_{j,k}$, we apply a CNOT gate with the k-th qubit as the control qubit and the k'-th qubit as the target qubit. This step requires at most ℓ CNOT gates. Subsequently, to eliminate the value $b_{j,k}$, we employ a multi-control Toffoli gate. The control qubits for this gate correspond to the non-zero elements in row j of matrix $\widetilde{A}$, while the target qubit is the k-th qubit. This Toffoli gate can be implemented by a circuit of size $O(\log 2m)$ according to Lemma 11.
- In the case where there is no element $b_{j,k} \neq 0$ in row j satisfying $k > \log 2m$, we first apply a multi-control Toffoli gate, where the control qubits are those corresponding to the non-zero elements in the current row, while the target qubit is the $(\log 2m + 1)$-th qubit. It is asserted that this Toffoli gate will not alter the elements in the rows indexed by 0 to $j-1$ of the current matrix, as otherwise, there would exist some $0 \leq j' \leq j-1$ such that the j'-th row is identical to the j-th row, contradicting the assumption that each row is distinct. This Toffoli gate contains at most $\log 2m$ control qubits. Hence, it can be implemented by a circuit of size $O(\log 2m)$ according to Lemma 11. Upon completion of this Toffoli gate, we revert to the first case.

Combining the above two cases, the transformation of row j can be implemented using a circuit of size $O(\ell + \log 2m)$. Therefore, the permutation $\sigma_{i,1}$ can be implemented using a circuit of size $O(n + m\ell + m \log m)$.

The final component for implementing σ is $\sigma_{i,2} = (0,1) \circ (2,3) \circ \cdots \circ (2m-2, 2m-1)$. Given that m is a power of 2, the permutation $\sigma_{i,2}$ essentially flips the first qubit when the last $n - \log 2m$ qubits are all in the state $|0\rangle$. Thus, this permutation can be realized using

$2(n - \log 2m)$ X gates and a multi-controlled Toffoli gate with the last $n - \log 2m$ qubits as the control qubits and the first qubit as the target qubit. According to Lemma 11, the circuit size for this Toffoli gate is $O(n - \log 2m)$.

In summary, the circuit size for implementing any permutation σ_i composed of m pairwise disjoint transpositions is $O(n + m\ell + m \log m)$. As long as $m \leq \frac{\log n}{4}$, the circuit size for implementing σ_i simplifies to $O(n)$ by noting that $\log 2m \leq \ell \leq 2^{2m} - 1$. Let $m := 2^{\lfloor \log(\frac{\log n}{4}) \rfloor}$. In Step 1, we decompose σ as $\sigma = \sigma_{M-1} \circ \cdots \circ \sigma_0$, where each σ_i consists of at most m pairwise disjoint transpositions and $M = \lfloor \frac{\text{size}(\sigma)}{m} \rfloor + O(\log(\min\{m, \text{size}(\sigma)\}))$. Thus, the circuit size for implementing σ is $O(\frac{n\,\text{size}(\sigma)}{\log n} + n \log \min\{\,\text{size}(\sigma), \log n\,\})$. ◀

Step 1 decomposes σ into a composition of $\{\sigma_i\}$ in time $O(d)$. Step 2 then generates the corresponding quantum circuit for each σ_i, with each requiring $O(n\,\text{size}(\sigma_i))$ classical preprocessing time. Hence, the total classical time complexity of Lemma 12 is $O(n\,\text{size}(\sigma))$.

▶ Remark 13. In particular, it is shown in the proof of Lemma 12 that, if the permutation σ is composed of pairwise disjoint transpositions and $\text{size}(\sigma) \leq \frac{\log n}{2}$ is a power of 2, then σ can be implemented by a quantum circuit of size $O(n)$.

Now, we are ready to derive the main theorem in this section.

▶ **Theorem 14** (Restatement of Theorem 1). *Any n-qubit d-sparse quantum state can be prepared by a quantum circuit of size $O(\frac{nd}{\log n} + n)$ without using ancillary qubits.*

Proof. The algorithm is illustrated in Algorithm 1. The basic idea is to first prepare a $\lceil \log d \rceil$-qubit dense quantum state $\sum_i \alpha_i |\sigma^{-1}(q_i)\rangle$ where σ is a specific permutation satisfying that $0 \leq \sigma^{-1}(q_i) \leq d - 1$, and then transform it to the target state $\sum_i \alpha_i |q_i\rangle$ by applying σ. The circuit C_1 prepares the $\lceil \log d \rceil$-qubit quantum state, and therefore can be implemented with circuit size $O(d)$. In the following, we turn to the implementation of σ.

After the first two for loops of Algorithm 1, we construct a permutation σ composed of at most d disjoint transpositions. When $d = \Omega(\log n \log \log n)$, according to Lemma 12, the permutation σ can be implemented by a circuit of size $O(\frac{nd}{\log n} + n)$. When $d = o(\log n \log \log n)$, we need to consider it a bit more carefully. Note that we can decompose σ as $\sigma = \sigma' \circ \sigma_{M-1} \cdots \circ \sigma_1 \circ \sigma_0$, where $M = O(\frac{d}{\log n})$, each σ_i is composed of pairwise disjoint transpositions such that $\text{size}(\sigma_i) \leq \frac{\log n}{2}$ is a power of 2, and σ' is a residual permutation composed of less than $\frac{\log n}{2}$ pairwise disjoint transpositions. However, we can always replace σ' with σ'' which is made up by complementing σ' with some *irrelevant* transpositions such that $\text{size}(\sigma'')$ is a power of 2. For example, (i, j) can be an irrelevant transposition if $i, j \notin \{q_i\}_{0 \leq i \leq d-1} \cup \{0, \ldots, d - 1\}$. We can always find enough irrelevant transpositions when $d = o(\log n \log \log n)$. Denote $\sigma'' \circ \sigma_{M-1} \cdots \circ \sigma_1 \circ \sigma_0$ as $\widetilde{\sigma}$, it is easily to verify that $\sum_i \alpha_i |\widetilde{\sigma}(\sigma^{-1}(q_i))\rangle = \sum_i \alpha_i |\sigma(\sigma^{-1}(q_i))\rangle = \sum_i \alpha_i |q_i\rangle$. According to Remark 13, $\widetilde{\sigma}$ can be implemented by a quantum circuit of size $O(\frac{nd}{\log n} + n)$.

In conclusion, for any d-sparse quantum state, Algorithm 1 constructs a preparation circuit without ancillary qubits, of size $O(\frac{nd}{\log n} + n)$. ◀

4 SQSP with m Ancillary Qubits

4.1 Algorithm for SQSP

In this section, we focus on investigating the trade-off between the number of ancillary qubits and the circuit size of SQSP. To achieve this, we introduce the following new representation of

Algorithm 1 SQSP without Ancillary Qubits

```
Input: $\mathcal{P} = \{(\alpha_i, q_i)\}_{0 \le i \le d-1}$.
Output: A quantum circuit for preparing state $\sum_i \alpha_i |q_i\rangle$.
Let Flag[d] be a Boolean vector with all elements initialized to 0 ;
Let $\sigma$ be an identity permutation ;
for $(\alpha_i, q_i)$ in $\mathcal{P}$ do
    if $q_i < d$ then
        Flag$[q_i] \leftarrow 1$;
    end if
end for
for $(\alpha_i, q_i)$ in $\mathcal{P}$ do
    if $q_i \ge d$ then
        find a index $k$ such that Flag$[k] = 0$;
        Flag$[k] \leftarrow 1$; $\sigma \leftarrow \sigma \circ (k, q_i)$;
    end if
end for
Construct a quantum circuit $C_1$ for preparing the $\lceil \log d \rceil$-qubit state $\sum_{i=0}^{d-1} \alpha_i |\sigma^{-1}(q_i)\rangle$
    on the $\lceil \log d \rceil$ least significant qubits ;
Construct a quantum circuit $C_2$ for implementing $\sigma$ (or $\widetilde{\sigma}$, see the proof of Theorem 1);
Output the circuit $C_2 C_1$ ;
```

integers, whose encoding efficiency lies between binary encoding and unary encoding. Recall that the unary encoding e_x of an integer x sets the x-th bit to 1, while all other bits are set to 0.

▶ **Definition 15.** *Given an n-bit integer $x = \sum_{i=0}^{n-1} 2^i x_i$ and another integer $r > 0$ such that r divides n, define $x(j,k)$ for any integers $0 \le j < k < n$ as $x(j,k) := \sum_{i=j}^{k-1} 2^{i-j} x_i$. The (n,r)-unary encoding of the integer x is expressed as $e_{x(n-r,n)} e_{x(n-2r,n-r)} \cdots e_{x(0,r)}$. In particular, when $r = n$, the (n,n)-unary encoding corresponds to the standard unary encoding.*

For example, when $n = 8$, $r = 2$, and $x = 11011000$, the $(8,2)$-unary encoding of x is given by $e_{11} e_{01} e_{10} e_{00} = 0001\ 0100\ 0010\ 1000$.

▶ Remark 16. For convenience in presentation, we assume that r divides n. However, even if r does not divide n, this will not affect the correctness of any subsequent conclusions.

In the following lemma, we show that any sparse quantum state can be efficiently prepared using the (n,r)-unary encoding.

▶ **Lemma 17** (Restatement of Lemma 9). *Given two positive integers n and r such that r divides n, and a set of size d $\{(q_i, \alpha_i)\}_{0 \le i \le d-1}$ such that $\sum_i |\alpha_i|^2 = 1$, $q_i \in \{0,1\}^n$ for all $0 \le i \le d-1$ and $q_i \ne q_j$ for any $i \ne j$, there exists a quantum circuit preparing the following $\frac{n2^r}{r}$-qubit state*

$$\sum_{i=0}^{d-1} \alpha_i |e_{q_i(n-r,n)}\rangle |e_{q_i(n-2r,n-r)}\rangle \dots |e_{q_i(0,r)}\rangle = \sum_{i=0}^{d-1} \alpha_i \left(\bigotimes_{j=0}^{\frac{n}{r}-1} |e_{q_i(jr,(j+1)r)}\rangle \right). \tag{16}$$

The circuit is of size $O(\frac{nd}{r})$, using one ancillary qubits.

Proof. See Appendix A. ◀

The following lemma transforms the unary encoding to the binary encoding.

▶ **Lemma 18.** *There exists a quantum circuit that converts the unary encoding of n-bit integers to the binary encoding, that is, achieving the following transformation for any $0 \leq i \leq 2^n - 1$:*

$$|e_i\rangle |0\rangle^n \rightarrow |0\rangle^{\otimes 2^n} |i\rangle \tag{17}$$

The circuit has a size of $O(n2^n)$.

Proof. See Appendix B. ◀

With the above two key lemmas established, we obtain the following theorem for a wide parameter regime.

▶ **Theorem 19.** *For any $d = \Omega(n \log n)$ and any $m \in [\Omega(n^2), O(\frac{nd}{\log nd} + n)]$, any n-qubit d-sparse quantum state can be prepared by a quantum circuit of size $O(\frac{nd}{\log(m+n)} + n)$ using m ancillary qubits.*

Proof. The algorithm is presented in Algorithm 2. Let $r := \lfloor \log \frac{m}{n} \rfloor$ such that $\frac{n2^r}{r} \leq m$. In Algorithm 2, we use an ancillary register $\mathtt{M}$ consisting of $\frac{n2^r}{r}$ qubits, which is further divided into $\frac{n}{r}$ sub-registers $\{\mathtt{M}_j\}_{0 \leq j \leq \frac{n}{r}-1}$ as in Lemma 9. The target state is prepared in another register $\mathtt{R}$, which is also divided into $\frac{n}{r}$ sub-registers $\{\mathtt{R}_j\}_{0 \leq j \leq \frac{n}{r}-1}$, each consisting of r qubits.

Let us first show the correctness of Algorithm 2. The initial state is

$$\left(\bigotimes_{j=0}^{\frac{n}{r}-1} |0\rangle_{\mathtt{M}_j}^{\otimes 2^r} \right) \otimes \left(\bigotimes_{j=0}^{\frac{n}{r}-1} |0\rangle_{\mathtt{R}_j}^{\otimes r} \right). \tag{18}$$

After Step 2, we have

$$\sum_{i=0}^{d-1} \alpha_i \left(\bigotimes_{j=0}^{\frac{n}{r}-1} |e_{q_i(jr,(j+1)r)}\rangle_{\mathtt{M}_j} \right) \otimes \left(\bigotimes_{j=0}^{\frac{n}{r}-1} |0\rangle_{\mathtt{R}_j}^{\otimes r} \right). \tag{19}$$

Note that each $|e_{q_i(jr,(j+1)r)}\rangle_{\mathtt{M}_j}$ is the unary encoding of the r-bit integer $q_i(jr,(j+1)r)$. Therefore, Step 3 is to convert $|e_{q_i(jr,(j+1)r)}\rangle_{\mathtt{M}_j} |0\rangle_{\mathtt{R}_j}^{\otimes r}$ to $|0\rangle_{\mathtt{M}_j}^{\otimes 2^r} |q_i(jr,(j+1)r)\rangle_{\mathtt{R}_j}$ for all $0 \leq j \leq \frac{n}{r} - 1$ according to Lemma 18. After Step 3, omitting the ancillary qubits, the state stored in the register $\mathtt{R}$ now is

$$\sum_{i=0}^{d-1} \alpha_i \left(\bigotimes_{j=0}^{\frac{n}{r}-1} |q_i(jr,(j+1)r)\rangle_{\mathtt{R}_j} \right) = \sum_{i=0}^{d-1} \alpha_i |q_i\rangle_{\mathtt{R}}, \tag{20}$$

as desired.

Finally, we analyze the size of the circuit. According to Lemma 9 and Lemma 18, the size of Step 2 and Step 3 is $O(\frac{nd}{r})$ and $\frac{n}{r} * O(r2^r)$, respectively. Therefore, the circuit size is $O(\frac{nd}{\log m - \log n} + m)$, which is $O(\frac{nd}{\log(m+n)} + n)$ when $m \in [\Omega(n^2), O(\frac{nd}{\log nd} + n)]$. ◀

The classical time to generate the quantum circuit in Steps 2 and 3 is $O(\frac{nd}{r})$ and $O(n2^r)$, respectively. Therefore, the total classical time complexity of Algorithm 2 is $O(\frac{nd}{\log(m+n)} + n)$, matching the circuit size.

Combining Theorem 1 with Theorem 19, we achieve the trade-off between the number of the ancillary qubits and the circuit size for any $d \geq 1$ and $m \in O(\frac{nd}{\log nd} + n)$.

Algorithm 2 SQSP with m Ancillary Qubits

```
Input: $\mathcal{P} = \{(\alpha_i, q_i)\}_{0\le i\le d-1}$ such that $d = \Omega(n\log n)$ and an integer $m \in [\Omega(n^2), O(\frac{nd}{\log d})]$.
Output: A quantum circuit for preparing state $\sum_{i=0}^{d-1}\alpha_i|q_i\rangle$ using at most $m$ ancillary qubits.
Let $r := \lfloor \log\frac{m}{n}\rfloor$, and prepare the initial state $\left(\bigotimes_{j=0}^{\frac{n}{r}-1}|0\rangle_{\mathtt{M}_j}^{\otimes 2^r}\right)\otimes\left(\bigotimes_{j=0}^{\frac{n}{r}-1}|0\rangle_{\mathtt{R}_j}^{\otimes r}\right)$;
Prepare state $\sum_{i=0}^{d-1}\alpha_i\left(\bigotimes_{j=0}^{\frac{n}{r}-1}|e_{q_i(jr,(j+1)r)}\rangle_{\mathtt{M}_j}\right)$ on the first $\frac{n2^r}{r}$ qubits according to Lemma 9;
Convert $|e_{q_i(jr,(j+1)r)}\rangle_{\mathtt{M}_j}$ to $|q_i(jr,(j+1)r)\rangle$ on the $jr$-th to $(j+1)r-1$-th qubits of the second register for all $j$.
```

▶ **Theorem 20** (Restatement of Theorem 2). *For any $m \in O(\frac{nd}{\log nd}+n)$, any n-qubit d-sparse quantum state can be prepared by a quantum circuit of size $O(\frac{nd}{\log(m+n)}+n)$ using m ancillary qubits.*

Proof. When $d = \Omega(n\log d)$ and $m \in \Omega(n^2)$, we resort to Theorem 19, and the size is $O(\frac{nd}{\log(m+n)}+n)$, satisfying the requirement of the theorem. Otherwise, we have $d = o(n\log n)$ (implying $m \in o(n^2)$) or $m \in o(n^2)$, and we turn to the algorithm without ancillary qubits, i.e., Theorem 1, and the circuit size is $O(\frac{nd}{\log n}+n)$, also satisfying the requirement of the theorem. The proof is completed. ◀

4.2 Lower Bound on the Circuit Size

In this section, we prove lower bounds on the circuit size for preparing sparse quantum states. The following proof is based on the counting argument. To overcome the challenge posed by the existence of an infinite number of single-qubit quantum gates, we discretize the parameters of these gates. For the sake of clarity, we first establish the lower bound on the circuit size without any assumption, and then proceed to the case with reasonable assumptions.

▶ **Theorem 21** (Restatement of Theorem 6). *Given m ancillary qubits available, there exist n-qubit d-sparse quantum states such that any algorithm to prepare them requires $\Omega(\frac{nd}{\log(m+n)+\log d}+n)$ elementary quantum gates.*

Proof. Consider the universal quantum gate set $G := \{R_x(\theta), R_y(\theta), R_z(\theta), CNOT\}$, where $\theta \in [0, 2\pi)$ is a parameter, and the set of quantum states

$$\mathcal{D}_d := \left\{ |\psi_{\mathcal{S}}\rangle := \frac{1}{\sqrt{d}}\sum_{i\in\mathcal{S}}|i\rangle \mid \mathcal{S}\subset\{0,\dots,2^n-1\}, |\mathcal{S}| = d\right\}. \tag{21}$$

For any $|\psi\rangle, |\phi\rangle \in \mathcal{D}_d$ such that $|\psi\rangle \ne |\phi\rangle$, we have $\||\psi\rangle - |\phi\rangle\| \ge \sqrt{\frac{2}{d}}$. Suppose the number of elementary quantum gates required to prepare a quantum state $|\psi\rangle \in \mathcal{D}_d$ in the worst case is T. Note that $T \le cdn$ for some constant $c > 0$ [17, 33].

Now we consider a new set of quantum gates $\widetilde{G}$ and a new quantum state preparation task. The new set $\widetilde{G}$ is constructed from G by restricting the precision of the rotation angles of the single-qubit gates to $\delta := \sqrt{\frac{1}{4c^2d^3n^2}}$. For instance, given a single-qubit gate $R_x(\theta)$ from G with a parameter θ, we set $\widetilde{\theta} = \delta\lfloor\frac{\theta}{\delta}\rfloor$, acquire $R_x(\widetilde{\theta})$ in $\widetilde{G}$, and approximate $R_x(\theta)$ with $R_x(\widetilde{\theta})$. The new quantum state preparation task is to construct a circuit with $\widetilde{G}$ to prepare a quantum state $|\widetilde{\psi}\rangle$ for any given $|\psi\rangle \in \mathcal{D}_d$ such that $\| \, |\psi\rangle - |\widetilde{\psi}\rangle \, \| \leq \sqrt{\frac{1}{4d}}$. Note that $\| \, |\phi\rangle - |\widetilde{\psi}\rangle \, \| > \sqrt{\frac{1}{4d}}$ for any $|\phi\rangle \in \mathcal{D}_d$ such that $|\psi\rangle \neq |\phi\rangle$.

The circuit to prepare $|\widetilde{\psi}\rangle$ with $\widetilde{G}$ can be constructed as follows: first, we construct a circuit $U = U_{T-1} \ldots U_0$ with G to prepare the quantum state $|\psi\rangle$ exactly, where $U_i \in G$ or $U_i = I$; then, we replace each single-qubit gate $R_l(\theta)$ ($l \in \{x, y, z\}$) in U with $R_l(\widetilde{\theta})$ from $\widetilde{G}$. It can be verified that this new circuit $\widetilde{U}$ prepare an approximate quantum state $|\widetilde{\psi}\rangle$ for $|\psi\rangle$, satisfying $\| \, |\psi\rangle - |\widetilde{\psi}\rangle \, \| \leq \sqrt{\frac{1}{4d}}$. This demonstrates the feasibility of achieving the new quantum state preparation task with $\widetilde{G}$. Note that the discussion also reduces the task of preparing $|\widetilde{\psi}\rangle$ with $\widetilde{G}$ to preparing $|\psi\rangle$ with G. Therefore, the lower bound on the circuit size of the former immediately implies the lower bound on the circuit size of the latter.

Note that $|\widetilde{G}| = \frac{6\pi}{\delta} + 1$. For each U_i, there are a total of $(\frac{6\pi}{\delta} + 2)$ choices of quantum gates (including the identity gate I) to select from. The position where the quantum gates act has at most $(m+n)^2$ choices. Therefore, by the counting argument, we have

$$\left((m+n)^2 \cdot (\frac{6\pi}{\delta} + 2) \right)^T \geq |\mathcal{D}_d| \geq (\frac{2^n}{d})^d \tag{22}$$

Thus, we have $T = \Omega(\frac{dn - d\log d}{\log(m+n) + \log d})$.

In addition, we have $T = \Omega(d)$ from the dimensionality argument. Also, we consider the preparation of the state $|1\rangle^{\otimes n}$ from the initial state $|0\rangle^{\otimes n}$ with arbitrary single-qubit gates and CNOT. Undoubtedly, we have to access all the qubits to prepare $|1\rangle^{\otimes n}$. Therefore, we get a lower bound $\Omega(n)$ on the circuit size.

Combining the argument above, we conclude that

$$T = \Omega\left(\frac{dn - d\log d}{\log(m+n) + \log d} + d + n \right) = \Omega\left(\frac{nd}{\log(m+n) + \log d} + n \right). \tag{23}$$

◀

▶ **Theorem 22** (Restatement of Theorem 4). *Suppose $d \leq 2^{\epsilon n}$ for a sufficiently small constant $\epsilon \in (0,1)$, and given m ancillary qubits, if an algorithm $\mathcal{A}$ for preparing n-qubit d-sparse quantum states satisfies the following conditions:*

1. *$\mathcal{A}$ uses at most $O(d)$ single-qubit rotation gates with arbitrary angles,*
2. *all other single-qubit gates amount to a total of $O(n)$ types,*

then $\mathcal{A}$ requires $\Omega(\frac{nd}{\log(m+n)} + n)$ elementary quantum gates in the worst case.

Proof. Following the same discussion as in the proof of Theorem 21, it remains to estimate the number of quantum circuits with T gates and containing only $O(d)$ single-qubit rotation gates with arbitrary angles. We distinguish $\{R_x(\theta), R_y(\theta), R_x(\theta)\}$ from other single-qubit gates in the circuit, so there are $O(n)$ types of single-qubit gates in total. First, we estimate the number of circuit templates without specifying the parameters of $\{R_x(\theta), R_y(\theta), R_x(\theta)\}$ in the circuit, and then we specify the angles of single-qubit rotation gates. Therefore, we have

$$((m+n)^2 \cdot O(n))^T \cdot (\frac{6\pi}{\delta})^{O(d)} \geq |\mathcal{D}_d| \geq (\frac{2^n}{d})^d. \tag{24}$$

Hence, we have $O(\log(m+n)) \cdot T + O(d \log nd) \geq nd - d\log d$, implying $T = \Omega(\frac{nd}{\log(m+n)})$ when $d \leq 2^{\epsilon n}$ for a sufficiently small constant $\epsilon \in (0,1)$. Similarly, combining $T = \Omega(d+n)$, we obtain the lower bound on the circuit size as $\Omega(\frac{nd}{\log(m+n)} + n)$. ◀

5 Conclusion

In this paper, we focus on optimizing the circuit size of preparing sparse quantum states in two scenarios: without and with ancillary qubits. First, we have demonstrated that, without ancillary qubits, any n-qubit d-sparse quantum state can be prepared by a circuit of size $O(\frac{nd}{\log n}+n)$. Second, we have proved that with $m\in O(\frac{nd}{\log nd}+n)$ ancillary qubits available, the circuit size is $O(\frac{nd}{\log(m+n)}+n)$. Finally, we have established a matching lower bound $\Omega(\frac{nd}{\log(m+n)}+n)$ on the circuit size for the case of m ancillary qubits available when $d\leq 2^{\delta n}$ for a sufficiently small constant $\delta\in(0,1)$ under reasonable assumptions. Additionally, we have provided a slightly weaker lower bound of $\Omega(\frac{nd}{\log(m+n)+\log d}+n)$ without any assumptions. Putting the above results together, we have obtained the optimal bound $\Theta(\frac{nd}{\log nd}+n)$ on the circuit size in the case of unlimited number of ancillary qubits available.

References

1 Nabila Abdessaied, Mathias Soeken, Michael Kirkedal Thomsen, and Rolf Drechsler. Upper bounds for reversible circuits based on young subgroups. *Information Processing Letters*, 114(6):282–286, 2014. doi:10.1016/j.ipl.2014.01.003.

2 Israel F Araujo, Carsten Blank, Ismael CS Araújo, and Adenilton J da Silva. Low-rank quantum state preparation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2023. doi:10.1109/TCAD.2023.3297972.

3 Adriano Barenco, Charles H Bennett, Richard Cleve, David P DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical Review A*, 52(5):3457, 1995. doi:10.1103/PhysRevA.52.3457.

4 Johannes Bausch. Fast black-box quantum state preparation. *Quantum*, 6:773, 2022. doi:10.22331/q-2022-08-04-773.

5 Ville Bergholm, Juha J Vartiainen, Mikko Möttönen, and Martti M Salomaa. Quantum circuits with uniformly controlled one-qubit gates. *Physical Review A*, 71(5):052330, 2005. doi:10.1103/PhysRevA.71.052330.

6 Dominic W Berry, Andrew M Childs, Richard Cleve, Robin Kothari, and Rolando D Somma. Simulating hamiltonian dynamics with a truncated taylor series. *Physical Review Letters*, 114(9):090502, 2015. doi:10.1103/physrevlett.114.090502.

7 Alex Brodsky. Reversible circuit realizations of Boolean functions. In *Exploring New Frontiers of Theoretical Informatics, IFIP 18th World Computer Congress, TC1 3rd International Conference on Theoretical Computer Science*, volume 155 of *IFIP*, pages 67–80. Kluwer/Springer, 2004. doi:10.1007/1-4020-8141-3_8.

8 Nai-Hui Chia, András Pal Gilyén, Tongyang Li, Han-Hsuan Lin, Ewin Tang, and Chunhao Wang. Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning. *Journal of the ACM*, 69(5):1–72, 2022. doi:10.1145/3549524.

9 Nai-Hui Chia, Tongyang Li, Han-Hsuan Lin, and Chunhao Wang. Quantum-inspired sublinear algorithm for solving low-rank semidefinite programming. In *Proceedings of the 45th International Symposium on Mathematical Foundations of Computer Science*, volume 170 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 23:1–23:15, 2020. doi:10.4230/LIPIcs.MFCS.2020.23.

10 Andrew M Childs, Dmitri Maslov, Yunseong Nam, Neil J Ross, and Yuan Su. Toward the first quantum simulation with quantum speedup. *Proceedings of the National Academy of Sciences*, 115(38):9456–9461, 2018. doi:10.1073/pnas.1801723115.

11 William Cottrell, Ben Freivogel, Diego M Hofman, and Sagar F Lokhande. How to build the thermofield double state. *Journal of High Energy Physics*, 2019(2):1–43, 2019. doi:10.1007/jhep02(2019)058.

12 Tiago ML de Veras, Leon D da Silva, and Adenilton J da Silva. Double sparse quantum state preparation. *Quantum Information Processing*, 21(6):204, 2022. `doi:10.1007/s11128-022-03549-y`.

13 Tiago ML de Veras, Ismael CS De Araujo, Daniel K Park, and Adenilton J da Silva. Circuit-based quantum random access memory for classical data with continuous amplitudes. *IEEE Transactions on Computers*, 70(12):2125–2135, 2020. `doi:10.1109/TC.2020.3037932`.

14 Richard P Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6/7), 1982.

15 Craig Gidney. Constructing large controlled nots. `https://algassert.com/circuits/2015/06/05/Constructing-Large-Controlled-Nots.html`, 2015.

16 András Gilyén, Zhao Song, and Ewin Tang. An improved quantum-inspired algorithm for linear regression. *Quantum*, 6:754, 2022. `doi:10.22331/q-2022-06-30-754`.

17 Niels Gleinig and Torsten Hoefler. An efficient algorithm for sparse quantum state preparation. In *Proceedings of the 58th ACM/IEEE Design Automation Conference*, pages 433–438. IEEE, 2021. `doi:10.1109/DAC18074.2021.9586240`.

18 Javier Gonzalez-Conde, Thomas W Watts, Pablo Rodriguez-Grasa, and Mikel Sanz. Efficient quantum amplitude encoding of polynomial functions. *Quantum*, 8:1297, 2024. `doi:10.22331/q-2024-03-21-1297`.

19 Lov Grover and Terry Rudolph. Creating superpositions that correspond to efficiently integrable probability distributions, 2002. URL: `https://arxiv.org/abs/quant-ph/0208112`, `arXiv:quant-ph/0208112`.

20 Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pages 212–219, 1996. `doi:10.1145/237814.237866`.

21 Marshall Hall. *The theory of groups*. Courier Dover Publications, 2018.

22 Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15):150502, 2009. `doi:10.1103/physrevlett.103.150502`.

23 Adam Holmes and Anne Y Matsuura. Efficient quantum circuits for accurate state preparation of smooth, differentiable functions. In *Proceedings of IEEE International Conference on Quantum Computing and Engineering*, pages 169–179. IEEE, 2020. `doi:10.1109/QCE49297.2020.00030`.

24 Raban Iten, Roger Colbeck, Ivan Kukuljan, Jonathan Home, and Matthias Christandl. Quantum circuits for isometries. *Physical Review A*, 93(3):032318, 2016. `doi:10.1103/PhysRevA.93.032318`.

25 Jiaqing Jiang, Xiaoming Sun, Shang-Hua Teng, Bujiao Wu, Kewen Wu, and Jialin Zhang. Optimal space-depth trade-off of CNOT circuits in quantum logic synthesis. In *Proceedings of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 213–229. SIAM, 2020. `doi:10.1137/1.9781611975994.13`.

26 Sonika Johri, Shantanu Debnath, Avinash Mocherla, Alexandros Singk, Anupam Prakash, Jungsang Kim, and Iordanis Kerenidis. Nearest centroid classification on a trapped ion quantum computer. *npj Quantum Information*, 7(1):122, 2021. `doi:10.1038/s41534-021-00456-5`.

27 Iordanis Kerenidis and Jonas Landman. Quantum spectral clustering. *Physical Review A*, 103(4):042415, 2021. `doi:10.1103/PhysRevA.103.042415`.

28 Iordanis Kerenidis, Jonas Landman, Alessandro Luongo, and Anupam Prakash. q-means: A quantum algorithm for unsupervised machine learning. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL: `https://proceedings.neurips.cc/paper_files/paper/2019/file/16026d60ff9b54410b3435b403afd226-Paper.pdf`.

29 Iordanis Kerenidis and Anupam Prakash. Quantum recommendation systems. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference*, volume 67 of *Leibniz*

International Proceedings in Informatics (LIPIcs), pages 49:1–49:21, 2017. doi:10.4230/LIPIcs.ITCS.2017.49.

30 Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum principal component analysis. *Nature Physics*, 10(9):631–633, 2014. doi:10.1038/nphys3029.

31 Guang Hao Low and Isaac L Chuang. Optimal hamiltonian simulation by quantum signal processing. *Physical Review Letters*, 118(1):010501, 2017. doi:10.1103/physrevlett.118.010501.

32 Guang Hao Low and Isaac L Chuang. Hamiltonian simulation by qubitization. *Quantum*, 3:163, 2019. doi:10.22331/q-2019-07-12-163.

33 Emanuel Malvetti, Raban Iten, and Roger Colbeck. Quantum circuits for sparse isometries. *Quantum*, 5:412, 2021. doi:10.22331/q-2021-03-15-412.

34 Rui Mao, Guojing Tian, and Xiaoming Sun. Toward optimal circuit size for sparse quantum state preparation. *Physical Review A*, 110:032439, Sep 2024. doi:10.1103/PhysRevA.110.032439.

35 Gabriel Marin-Sanchez, Javier Gonzalez-Conde, and Mikel Sanz. Quantum algorithms for approximate function loading. *Physical Review Research*, 5(3):033114, 2023. doi:10.1103/PhysRevResearch.5.033114.

36 Ketan Markov, Igor Patel, and John Hayes. Optimal synthesis of linear reversible circuits. *Quantum Information and Computation*, 8(3&4):0282–0294, 2008. doi:10.26421/QIC8.3-4-4.

37 Sam McArdle, András Gilyén, and Mario Berta. Quantum state preparation without coherent arithmetic, 2022. URL: https://arxiv.org/abs/2210.14892, arXiv:2210.14892.

38 Cristopher Moore and Martin Nilsson. Parallel quantum computation and quantum codes. *SIAM Journal on Computing*, 31(3):799–815, 2001. doi:10.1137/S0097539799355053.

39 Mudassir Moosa, Thomas W Watts, Yiyou Chen, Abhijat Sarma, and Peter L McMahon. Linear-depth quantum circuits for loading fourier approximations of arbitrary functions. *Quantum Science and Technology*, 9(1):015002, 2023. doi:10.1088/2058-9565/acfc62.

40 Fereshte Mozafari, Giovanni De Micheli, and Yuxiang Yang. Efficient deterministic preparation of quantum states using decision diagrams. *Physical Review A*, 106(2):022617, 2022. doi:10.48550/arXiv.2206.08588.

41 Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*, volume 2. Cambridge university press Cambridge, 2001. doi:10.1145/505482.505499.

42 Daniel K Park, Francesco Petruccione, and June-Koo Kevin Rhee. Circuit-based quantum random access memory for classical data. *Scientific Reports*, 9(1):3949, 2019. doi:10.1038/s41598-019-40439-3.

43 Martin Plesch and Časlav Brukner. Quantum-state preparation with universal gate decompositions. *Physical Review A*, 83(3):032302, 2011. doi:10.1103/PhysRevA.83.032302.

44 Debora Ramacciotti, Andreea I Lefterovici, and Antonio F Rotundo. Simple quantum algorithm to efficiently prepare sparse states. *Physical Review A*, 110(3):032609, 2024. doi:10.1103/physreva.110.032609.

45 Arthur G. Rattew and Bálint Koczor. Preparing arbitrary continuous functions in quantum registers with logarithmic complexity, 2022. URL: https://arxiv.org/abs/2205.00519, arXiv:2205.00519.

46 Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *Physical Review Letters*, 113(13):130503, 2014. doi:10.1103/PhysRevLett.113.130503.

47 Mehdi Saeedi and Igor L Markov. Synthesis and optimization of reversible circuits—a survey. *ACM Computing Surveys*, 45(2):1–34, 2013. doi:10.1145/2431211.2431220.

48 Mehdi Saeedi, Mehdi Sedighi, and Morteza Saheb Zamani. A library-based synthesis methodology for reversible logic. *Microelectronics Journal*, 41(4):185–194, 2010. doi:10.1016/J.MEJO.2010.02.002.

49 Mehdi Saeedi, Morteza Saheb Zamani, Mehdi Sedighi, and Zahra Sasanian. Reversible circuit synthesis using a cycle-based approach. *ACM Journal on Emerging Technologies in Computing Systems*, 6(4):1–26, 2010. doi:10.1145/1877745.1877747.

50 Yuval R Sanders, Guang Hao Low, Artur Scherer, and Dominic W Berry. Black-box quantum state preparation without arithmetic. *Physical Review Letters*, 122(2):020502, 2019. doi:10.1103/PhysRevLett.122.020502.

51 Vivek V Shende, Stephen S Bullock, and Igor L Markov. Synthesis of quantum logic circuits. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, pages 272–275, 2005. doi:10.1145/1120725.1120847.

52 Vivek V Shende, Igor L Markov, and Stephen S Bullock. Smaller two-qubit circuits for quantum communication and computation. In *Proceedings Design, Automation and Test in Europe Conference and Exhibition*, volume 2, pages 980–985. IEEE, 2004. doi:10.1109/DATE.2004.1269020.

53 Vivek V Shende, Aditya K Prasad, Igor L Markov, and John P Hayes. Synthesis of reversible logic circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 22(6):710–722, 2003. doi:10.1109/TCAD.2003.811448.

54 Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 124–134. IEEE, 1994. doi:10.1109/SFCS.1994.365700.

55 Xiaoming Sun, Guojing Tian, Shuai Yang, Pei Yuan, and Shengyu Zhang. Asymptotically optimal circuit depth for quantum state preparation and general unitary synthesis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 42(10):3301–3314, 2023. doi:10.1109/TCAD.2023.3244885.

56 Ewin Tang. A quantum-inspired classical algorithm for recommendation systems. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 217–228, 2019. doi:10.1145/3313276.3316310.

57 Ewin Tang. Quantum principal component analysis only achieves an exponential speedup because of its state preparation assumptions. *Physical Review Letters*, 127(6):060503, 2021. doi:10.1103/PhysRevLett.127.060503.

58 Carlo A Trugenberger. Probabilistic quantum memories. *Physical Review Letters*, 87(6):067901, 2001. doi:10.1103/PhysRevLett.87.067901.

59 Shengbin Wang, Zhimin Wang, Guolong Cui, Shangshang Shi, Ruimin Shang, Lixin Fan, Wendong Li, Zhiqiang Wei, and Yongjian Gu. Fast black-box quantum state preparation based on linear combination of unitaries. *Quantum Information Processing*, 20(8):270, 2021. doi:10.1007/s11128-021-03203-z.

60 Robert Wille and Rolf Drechsler. Bdd-based synthesis of reversible logic for large functions. In *Proceedings of the 46th Annual Design Automation Conference*, pages 270–275, 2009. doi:10.1145/1629911.1629984.

61 Xian Wu and Lvzhou Li. Asymptotically optimal synthesis of reversible circuits. *Information and Computation*, 301:105235, 2024. URL: https://www.sciencedirect.com/science/article/pii/S0890540124001007, doi:10.1016/j.ic.2024.105235.

62 Dmitry V Zakablukov. On asymptotic gate complexity and depth of reversible circuits without additional memory. *Journal of Computer and System Sciences*, 84:132–143, 2017. doi:10.1016/j.jcss.2016.09.010.

63 Xiao-Ming Zhang, Tongyang Li, and Xiao Yuan. Quantum state preparation with optimal circuit depth: Implementations and applications. *Physical Review Letters*, 129(23):230504, 2022. doi:10.1103/PhysRevLett.129.230504.

64 Xiao-Ming Zhang and Xiao Yuan. Circuit complexity of quantum access models for encoding classical data. *npj Quantum Information*, 10(1):42, 2024. doi:10.1038/s41534-024-00835-8.

A Proof of Lemma 9

▶ **Lemma 23** (Restatement of Lemma 9). *Given two positive integers n and r such that r divides n, and a set of size d $\{(q_i,\alpha_i)\}_{0\le i\le d-1}$ such that $\sum_i|\alpha_i|^2=1$, $q_i\in\{0,1\}^n$ for all $0\le i\le d-1$ and $q_i\neq q_j$ for any $i\neq j$, there exists a quantum circuit preparing the following $\frac{n2^r}{r}$-qubit state*

$$\sum_{i=0}^{d-1}\alpha_i\left|e_{q_i(n-r,n)}\right\rangle\left|e_{q_i(n-2r,n-r)}\right\rangle\dots\left|e_{q_i(0,r)}\right\rangle=\sum_{i=0}^{d-1}\alpha_i\left(\bigotimes_{j=0}^{\frac{n}{r}-1}\left|e_{q_i(jr,(j+1)r)}\right\rangle\right). \tag{25}$$

The circuit is of size $O(\frac{nd}{r})$, using one ancillary qubits.

Proof. The algorithm is presented in Algorithm 3. The basic idea is similar to that of [13, 12, 34], which aimed to prepare a sparse quantum state directly using binary encoding. Let $|0\rangle_{\mathtt{anc}}$ be the ancillary qubit, and let $\mathtt{M}$ denote the $\frac{n2^r}{r}$-qubit working register, which is further divided into $\frac{n}{r}$ sub-register $\{\mathtt{M}_j\}_{0\le j\le\frac{n}{r}-1}$, each consisted of 2^r qubits. For any $\alpha\in\mathbb{C}$ and $\beta\in\mathbb{R}_+$ such that $0\le|\alpha|\le\beta\le1$, define one-qubit gate $G(\alpha,\beta)$ as follows:

$$G(\alpha,\beta):=\frac{1}{\beta}\begin{pmatrix}\sqrt{\beta^2-|\alpha|^2} & \alpha\\ \alpha^\dagger & \sqrt{\beta^2-|\alpha|^2}\end{pmatrix}. \tag{26}$$

The algorithm proceeds in an iterative manner, and we argue that before the beginning of the iteration (corresponding to $i=-1$) and at the end of the i-th iteration for all $0\le i\le d-1$, the state of the whole register is as follows:

$$|0\rangle_{\mathtt{anc}}\otimes\sum_{k=0}^{i}\alpha_k\left(\bigotimes_{j=0}^{\frac{n}{r}-1}\left|e_{q_k(jr,(j+1)r)}\right\rangle_{\mathtt{M}_j}\right)+\sqrt{\sum_{k>i}|\alpha_k|^2}\,|1\rangle_{\mathtt{anc}}\otimes\left(\bigotimes_{j=0}^{\frac{n}{r}-1}|0\rangle_{\mathtt{M}_j}^{\otimes2^r}\right). \tag{27}$$

We prove Equation (27) by induction. Before the beginning of the iteration, the initial state is $|1\rangle_{\mathtt{anc}}\otimes\left(\bigotimes_{j=0}^{\frac{n}{r}-1}|0\rangle_{\mathtt{M}_j}^{\otimes2^r}\right)$, satisfying Equation (27) for $i=-1$.

In the $(i+1)$-the iteration, the left part of Equation (27), i.e., the partial state with the ancillary qubit being $|0\rangle$, remains unchanged. After Step 3, the right part of Equation (27) is transformed to

$$\sqrt{\sum_{k>i}|\alpha_k|^2}\,|1\rangle_{\mathtt{anc}}\otimes\left(\bigotimes_{j=0}^{\frac{n}{r}-1}\left|e_{q_{i+1}(jr,(j+1)r)}\right\rangle_{\mathtt{M}_j}\right), \tag{28}$$

which is transformed to, after Step 4,

$$\alpha_{i+1}|0\rangle_{\mathtt{anc}}\otimes\left(\bigotimes_{j=0}^{\frac{n}{r}-1}\left|e_{q_{i+1}(jr,(j+1)r)}\right\rangle_{\mathtt{M}_j}\right)+\sqrt{\sum_{k>i+1}|\alpha_k|^2}\,|1\rangle_{\mathtt{anc}}\otimes\left(\bigotimes_{j=0}^{\frac{n}{r}-1}\left|e_{q_{i+1}(jr,(j+1)r)}\right\rangle_{\mathtt{M}_j}\right). \tag{29}$$

After Step 5, the second part of Equation (29) is restored to $\sqrt{\sum_{k>i+1}|\alpha_k|^2}\,|1\rangle_{\mathtt{anc}}\otimes\left(\bigotimes_{j=0}^{\frac{n}{r}-1}|0\rangle_{\mathtt{M}_j}^{\otimes2^r}\right)$. Therefore, we get the desired state as Equation (27) for $i\leftarrow i+1$.

Finally, we turn to the circuit size of the algorithm. In each iteration, in addition to a total of $\frac{2n}{r}$ CNOT gates in Step 3 and Step 5, one $G(\alpha,\beta)$ gate conditioned on $\frac{n}{r}$ qubits are applied, which can be implemented with two one-qubit gates and one $(\frac{n}{r}+1)$-qubit Toffoli gate [3]. Therefore, the circuit size of the circuit is $O(\frac{nd}{r})$. ◀

Algorithm 3 SQSP using (n, r)-unary encoding

```
Input: $\mathcal{P} = \{(\alpha_i, q_i)\}_{0\le i\le d-1}$.
Output: A quantum circuit for preparing state $\sum_{i=0}^{d-1}\alpha_i\left(\bigotimes_{j=0}^{\frac{n}{r}-1}|e_{q_i(jr,(j+1)r)}\rangle\right)$.
1: Prepare the initial state $|1\rangle_{\mathtt{anc}}\otimes\left(\bigotimes_{j=0}^{\frac{n}{r}-1}|0\rangle_{\mathtt{M}_j}^{\otimes 2^r}\right)$;
2: for $i = 0$ to $d-1$ do
3:     Apply a total of $\frac{n}{r}$ CNOT gates to transform $\bigotimes_{j=0}^{\frac{n}{r}-1}|0\rangle_{\mathtt{M}_j}^{\otimes 2^r}$ to $\bigotimes_{j=0}^{\frac{n}{r}-1}|e_{q_i(jr,(j+1)r}\rangle_{\mathtt{M}_j}$
       conditioned on the state of the ancillary qubit being $|1\rangle$;
4:     Apply $G(\alpha_i, \sum_{j=i}^{d-1}|\alpha_j|^2)$ on the ancillary qubit conditioned on the state of $\mathtt{M}_j$ being
       $|e_{q_i(jr,(j+1)r}\rangle$ for all $0\le j\le\frac{n}{r}-1$;
5:     Repeat Step 3;
6: end for
```

The following is a concrete example illustrating the workflow of Algorithm 3.

▶ **Example 24.** Let $d = 3$, $n = 8$, $r = 2$, with $q_0 = 11011000$, $q_1 = 00011001$, and $q_2 = 10001111$. The initial state is $|1\rangle|0\rangle^{\otimes 16}$. In the first iteration, the state evolves as follows:

$$\begin{aligned}|1\rangle|0\rangle^{\otimes 16} &\to |1\rangle|0001\ 0100\ 0010\ 1000\rangle &(30)\\ &\to \sqrt{1-|\alpha_0|^2}\,|1\rangle|0001\ 0100\ 0010\ 1000\rangle + \alpha_0|0\rangle|0001\ 0100\ 0010\ 1000\rangle &(31)\\ &\to \sqrt{1-|\alpha_0|^2}\,|1\rangle|0\rangle^{\otimes 16} + \alpha_0|0\rangle|0001\ 0100\ 0010\ 1000\rangle. &(32)\end{aligned}$$

In the second iteration, the state is further transformed:

$$\begin{aligned}\textit{Equation }(32) \to &\sqrt{1-|\alpha_0|^2}\,|1\rangle|1000\ 0100\ 0010\ 0100\rangle\\ &+\alpha_0|0\rangle|0001\ 0100\ 0010\ 1000\rangle &(33)\\ \to &|\alpha_2|\,|1\rangle|1000\ 0100\ 0010\ 0100\rangle\\ &+\alpha_1|0\rangle|1000\ 0100\ 0010\ 0100\rangle\\ &+\alpha_0|0\rangle|0001\ 0100\ 0010\ 1000\rangle &(34)\\ \to &|\alpha_2|\,|1\rangle|0\rangle^{\otimes 16}\\ &+\alpha_1|0\rangle|1000\ 0100\ 0010\ 0100\rangle\\ &+\alpha_0|0\rangle|0001\ 0100\ 0010\ 1000\rangle. &(35)\end{aligned}$$

The same argument applies to the third iteration.

B Proof of Lemma 18

▶ **Lemma 25** (Restatement of Lemma 18). *There exists a quantum circuit that converts the unary encoding of n-bit integers to the binary encoding, that is, achieving the following transformation for any $0 \le i \le 2^n - 1$:*

$$|e_i\rangle|0\rangle^n \to |0\rangle^{\otimes 2^n}|i\rangle \tag{36}$$

The circuit has a size of $O(n2^n)$.

Proof. The algorithm is presented in Algorithm 4. It is direct to verify the correctness of the algorithm. In each iteration, at most n CNOT gates and one $n+1$-qubit Toffoli gate are applied. Therefore, the circuit size of the circuit is $O(n2^n)$. ◀

Algorithm 4 A Quantum Circuit Converting the Unary Encoding to the Binary Encoding

```
Input: An integer $n > 0$.
Output: A quantum circuit performing $|e_i\rangle |0\rangle^{\otimes n} \to |0\rangle^{\otimes 2^n} |i\rangle$.
for $i = 0$ to $2^n - 1$ do
    Transform the second register from $|0\rangle^{\otimes n}$ to $|i\rangle$ conditioned on the state of the $i$-th qubit of the first register being $|1\rangle$ using at most $n$ CNOT gates;
    Apply a $(n+1)$-qubit Toffoli gate conditioned on the state of the second register being $|i\rangle$ and targeted on the the $i$-th qubit of the first register;
end for
```