On the Need for Large Quantum Depth

NAI-HUI CHIA, Department of Computer Science, Rice University, USA
KAI-MIN CHUNG, Institute of Information Science, Academia Sinica, Taiwan
CHING-YI LAI, Institute of Communications Engineering, National Yang Ming Chiao Tung University, Taiwan

Near-term quantum computers are likely to have small depths due to short coherence time and noisy gates. A natural approach to leverage these quantum computers is interleaving them with classical computers. Understanding the capabilities and limits of this hybrid approach is an essential topic in quantum computation. Most notably, the quantum Fourier transform can be implemented by a hybrid of logarithmic-depth quantum circuits and a classical polynomial-time algorithm. Therefore, it seems possible that quantum polylogarithmic depth is as powerful as quantum polynomial depth in the presence of classical computation. Indeed, Jozsa conjectured that "*Any quantum polynomial-time algorithm can be implemented with only $O(\log n)$ quantum depth interspersed with polynomial-time classical computations.*" This can be formalized as asserting the equivalence of BQP and "$\mathsf{BQNC}^{\mathsf{BPP}}$." However, Aaronson conjectured that "*there exists an oracle separation between* BQP *and* $\mathsf{BPP}^{\mathsf{BQNC}}$." $\mathsf{BQNC}^{\mathsf{BPP}}$ and $\mathsf{BPP}^{\mathsf{BQNC}}$ are two natural and seemingly incomparable ways of hybrid classical-quantum computation.

In this work, we manage to prove Aaronson's conjecture and in the meantime prove that Jozsa's conjecture, relative to an oracle, is false. In fact, we prove a stronger statement that for any depth parameter d, there exists an oracle that separates quantum depth d and $2d+1$ in the presence of classical computation. Thus, our results show that relative to oracles, doubling the quantum circuit depth does make the hybrid model more powerful, and this cannot be traded by classical computation.

CCS Concepts: • **Theory of computation → Quantum complexity theory;**

Additional Key Words and Phrases: Small-depth quantum circuit, hybrid quantum-classical computer, near-term quantum computer, d-shuffling Simon's problem, oracle separation

ACM Reference format:
Nai-Hui Chia, Kai-min Chung, and Ching-Yi Lai. 2023. On the Need for Large Quantum Depth. *J. ACM* 70, 1, Article 6 (January 2023), 38 pages.
https://doi.org/10.1145/3570637

Nai-Hui Chia's research is supported by Scott Aaronson's Vannevar Bush Faculty Fellowship from the US Department of Defense. Kai-Min Chung's research is partially supported by the Academia Sinica Career Development Award from the Ministry of Science and Technology (MOST) in Taiwan under Grant 23-17 MOST QC project, and MOST QC project under Grant MOST 107-2627-E-002-002. Ching-Yi Lai was financially supported by MOST in Taiwan, under Grant MOST 110-2628-E-A49-007.

Authors' addresses: N.-H. Chia, Department of Computer Science, Rice University, Houston; email: nc67@rice.edu; K.-M. Chung, Institute of Information Science, Academia Sinica, Taipei, Taiwan; email: kmchung@iis.sinica.edu.tw; C.-Y. Lai, Institute of Communications Engineering, National Yang Ming Chiao Tung University, Hsinchu, Taiwan; email: cylai@nycu.edu.tw.

0004-5411/2023/01-ART6 $15.00
https://doi.org/10.1145/3570637

1 INTRODUCTION

Circuit depth may become an essential consideration when designing algorithms on near-term quantum computers. Quantum computers with more than 50 qubits have been realized recently by Google [20] and IBM [18]; both the quantity and quality of the qubits are continually improving. Furthermore, Google and NASA recently showed that their quantum computer outperforms the best supercomputers on the task of random circuit sampling [8]. However, due to noisy gates and limited coherence time, these quantum computers are only able to operate for a short period. Hence, the effective circuit depths of these quantum computers are limited,* and this seems to be an essential bottleneck for quantum technologies.

Studies from a theoretical perspective indicate that small-depth quantum computers can demonstrate so-called "Quantum Supremacy" [5, 23], which means that a quantum device can solve some computational problems that are intractable for classical computers. Terhal and DiVincenzo first showed that constant-depth quantum circuits can sample certain distributions that are intractable for classical computers under some plausible complexity conjecture [23]. Later, Bremner, Jozsa, and Shepherd [10] introduced a class of commuting quantum computations that are difficult to simulate by classical computers unless the **polynomial hierarchy (PH)** collapses. Combining these ideas, one leads to the conclusion that constant depth quantum circuits can sample from distributions that are intractable for classical computers based on the conjecture that PH is infinite. Then Aaronson and Chen conceived a statistical test that, under a natural average-case hardness assumption, no polynomial-time classical algorithm can pass, but a small-depth quantum circuit can [5].

However, it is worth noting that the computation capability of a constant-depth quantum computer is limited, since it cannot solve certain problems that are classically easy. Specifically, consider the standard setting that the composing gate set of a quantum circuit includes only one- and two-qubit gates. It is obvious that a constant-depth quantum circuit cannot solve any classically intractable decision problem or even some classically easy problems, e.g., computing a parity function, because each output qubit depends on only $O(1)$ input qubits. A quantum circuit with unbounded fan-out gates is allowed to conduct many operations in small depth, such as parity, mod[q], threshold[t], arithmetic operations, phase estimations, and the quantum Fourier transform [17]. However, it seems difficult to implement unbounded fan-out gates in practice and thus they are rarely considered for near-term quantum devices.†

A natural idea to exploit the power of small-depth quantum computers is a hybrid of classical and quantum computation, where small depth quantum computers are interleaved with classical computers so quantum computations may enhance classical computations and vice versa. Many quantum algorithms require only small depths in the quantum part; notably, Cleve and Watrous showed that the quantum Fourier transform can be parallelized to have only logarithmic quantum depth [12], which implies that quantum algorithms for Abelian hidden subgroup problems, such as Shor's factoring algorithm, can be implemented with logarithmic quantum depth. Therefore, "quantum polylogarithmic depth is as powerful as quantum polynomial depth in the presence of classical computation" seems to be a live possibility!

Aware of this possibility, Jozsa [19] conjectured that

> "*Any polynomial time quantum algorithm can be implemented with only $O(\log n)$ quantum depth interspersed with polynomial-time classical computations.*"

*The experiments of Google and NASA consider quantum circuits with depth at most 20.
†Our oracle separation results hold even if the unbounded fan-out gates are allowed.

Nevertheless, there are other opinions in the community. It has been conjectured by Aaronson [1–4] more than a decade ago that

> "*There exists an oracle O relative to which* $\mathsf{BQP} \neq \mathsf{BPP}^{\mathsf{QNC}}$."

Here, $\mathsf{BPP}^{\mathsf{BQNC}}$ is corresponding to one of the hybrid approaches for interleaving classical computers with small-depth quantum circuits. It is worth noting that the models Jozsa and Aaronson considered were related but different, and we will clarify this later in Section 1.2.

In this work, we prove Aaronson's conjecture and show that Jozsa's conjecture, in the relativized world, does not hold. In fact, we prove a stronger conclusion that relative to oracles, a computational model would be more powerful if the quantum circuit depth is doubled, and this cannot be traded by classical computations.

1.1 Main Results

We start by defining two hybrid schemes of classical and quantum computation. The first scheme, called **d-depth quantum-classical scheme (d-QC scheme)**, is based on a d-depth quantum circuit with access to some classical computational resources after each depth and it can be considered as a generalized model for small-depth **measurement-based quantum computers (MBQCs)**. The second scheme, called **d-depth classical-quantum scheme (d-CQ scheme)**, is based on a classical computer with access to some d-depth quantum circuits.

We (informally) define $\mathsf{BQNC}_{\mathrm{d}}^{\mathsf{BPP}}$ to be the set of languages decided by d-QC schemes, and $\mathsf{BPP}^{\mathsf{BQNC}_{\mathrm{d}}}$ to be the set of languages decided by d-CQ schemes. Also, $\mathsf{BQNC}^{\mathsf{BPP}}$ (respectively, $\mathsf{BPP}^{\mathsf{BQNC}}$) refers to the union of $\mathsf{BQNC}_{\log^k \mathrm{n}}^{\mathsf{BPP}}$ (respectively, $\mathsf{BPP}^{\mathsf{BQNC}_{\log^k \mathrm{n}}}$) for constant $k \in \mathbb{N}$. (These definitions will be formally given in Section 3.)

Note that Jozsa's conjecture refers to the equivalence problem of BQP and d-QC schemes, and Aaronson's conjecture refers to the separation between BQP and d-CQ schemes.

In the following, we will introduce an oracle problem that can be used to show separation results for Aaronson's conjecture and Jozsa's conjecture relative to an oracle. Our oracle problem is a variant of *Simon's problem*. Given a function $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ with the promise that there exists $s \in \mathbb{Z}_2^n$ such that $f(x) = f(x \oplus s)$ for all $x \in \mathbb{Z}_2^n$, Simon's problem is to find s [22]. Such two-to-one function f is called a *Simon's function*. Simon's problem is easy for **quantum-polynomial-time (QPT)** algorithms but hard for all **probabilistic-polynomial-time (PPT)** algorithms; however, Simon's problem can be solved by a constant-depth quantum circuit with classical postprocessing.

For our purpose, we have to devise a harder problem. We first represent a Simon's function as a composition of random one-to-one functions $f_0, \ldots, f_{d-1}$ and a two-to-one function f_d such that $f = f_d \circ f_{d-1} \circ \cdots \circ f_0$, as shown in Figure 1, where $f_j : S_j \to S_{j+1}$ has domain S_j and range S_{j+1} for $j = 0, \ldots, d$. As a result, the inputs of the function f are "shuffled" in the sequence of functions. Suppose that f is hidden and only oracle access to the functions f_i is provided.[‡] If $f_0, \ldots, f_d$ must be queried in sequence to evaluate f, then a d-depth quantum circuit cannot solve this variant of Simon's problem, since it can only make at most d sequential queries (but there are $d + 1$ random functions). This idea is called *pointer chasing*.

Nevertheless, a more clever approach that does not query the functions in sequence might exist. To rule out the possibility of such an approach, we further *make it infeasible for the domain of f_i to be accessed before the $(i + 1)$-th parallel queries*. Specifically, f_j is now defined on a larger domain

[‡]Note that multiple parallel queries can be made at each circuit depth.

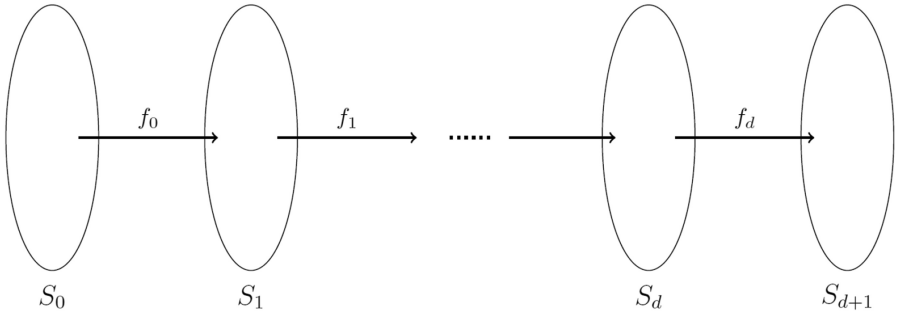

Fig. 1. Composing d random one-to-one function $f_0, f_1 \ldots, f_{d-1}$ and a two-to-one function f_d such that $f = f_d \circ \cdots \circ f_0$.

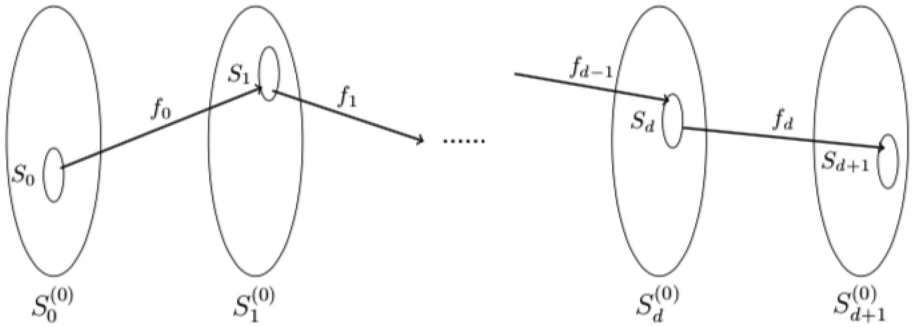

Fig. 2. The shuffling oracle: $f_0, \ldots, f_{d-1}$ are random one-to-one functions on greater domains. For $x \in S_0$, $f(x) = f_d \circ \cdots \circ f_0(x)$.

$S_j^{(0)}$ such that S_j is a subset of $S_j^{(0)}$ and $|S_j|/|S_j^{(0)}|$ is chosen to be negligible for $j = 0, 1, \ldots, d+1$. This is illustrated in Figure 2. Therefore, to evaluate f_i, one must find the target domain S_i from the larger domain $S_i^{(0)}$, and the success probability is negligible by construction. This idea is called *domain hiding*.

It is worth noting that an algorithm can learn S_i after the ith query with probability one, and thus is able to evaluate f after $d+1$ sequential queries. However, using at most d sequential queries is still not enough for the algorithm to evaluate f.

To sum up, the oracle, we consider is as follows: Let f be an arbitrary Simon function or a one-to-one function. Choose a sequence of random one-to-one functions $f_0, \ldots, f_{d-1}$ defined on much larger domains $S_0^{(0)}, \ldots, S_d^{(0)}$, respectively, and let f_d be a function such that $f_d \circ \cdots \circ f_0(x) = f(x)$ for $x \in S_0$. We define the *d-shuffling oracle* of f with respect to $f_0, \ldots, f_d$ to be an oracle that returns a value of $f_0, \ldots, f_{d-1}$ or f_d when queried. A quantum algorithm that has oracle access to the *d-shuffling oracle* of f can query one f_i in one query and can make superposition queries to each f_i. More specific, for $i \in [d]$, $f_i|x\rangle|y\rangle = |x\rangle|y \oplus f_i(x)\rangle$, where x and y are in the domain and range of f_i. We will simply call it the *d-shuffling oracle* of f in the following without mentioning the underlying decomposition functions $f_0, \ldots, f_{d-1}$.

The ***d-Shuffling Simon's Problem*** (d-SSP) is defined as follows:

Definition 1.1 (d-SSP (Informal)). Let f be a one-to-one function or a Simon function and d be a positive integer. Given oracle access to the d-shuffling oracle of f (as in Figure 2), the problem is to decide whether f is a Simon function or not.

We summarize our results in the following theorem:

Theorem 1.2 (Informal). *Let O denote the shuffling oracle of a random Simon function f. For any d, d-SSP $\in (\mathrm{BQNC}^{\mathrm{BPP}}_{2d+1})^O \cap (\mathrm{BPP}^{\mathrm{BQNC}_{2d+1}})^O$, but d-SSP $\notin (\mathrm{BQNC}^{\mathrm{BPP}}_{d})^O \cup (\mathrm{BPP}^{\mathrm{BQNC}_d})^O$.*[§]

Theorem 1.2 states that relative to the shuffling oracles, $(2d+1)$-QC schemes (respectively, $(2d+1)$-CQ schemes) are strictly more powerful than d-QC schemes (respectively, d-CQ schemes). Namely, the hybrid model can be strictly more powerful with quantum depth doubled, and this cannot be traded for classical computation. Theorem 1.2 immediately implies the following corollary, which states that Jozsa's conjecture, relative to an oracle, is false:

Corollary 1.3 (Informal). *For $d = \log^{\omega(1)} n$, d-SSP $\in \mathrm{BQP}^O$, but d-SSP $\notin (\mathrm{BPP}^{\mathrm{BQNC}})^O \cup (\mathrm{BQNC}^{\mathrm{BPP}})^O$, where O is the shuffling oracle of f.*

1.2 Discussion and Open Problems

Jozsa's and Aaronson's conjectures. The hybrid model Jozsa considered in Reference [19] is the small-depth MBQC, which is characterized by the d-QC scheme in this work. However, Aaronson considered the complexity class $\mathrm{BPP}^{\mathrm{BQNC}}$, which corresponds to the d-CQ scheme.

Since we cannot even prove that BPP $\neq$ BQP, an unconditional separation seems unachievable by the state-of-the-art techniques. Instead, we can expect an oracle separation or a conditional lower bound based on certain assumptions in cryptography or complexity. In this work, we obtain an oracle separation between BQP and $\mathrm{BPP}^{\mathrm{BQNC}} \cup \mathrm{BQNC}^{\mathrm{BPP}}$. Consequently, one central question is: Can we achieve a conditional separation based on plausible assumptions, or equivalently, can we instantiate our oracle? A natural approach is to instantiate it based on cryptographic assumptions, such as **virtual-black-box (VBB)** obfuscations. Along this line, we may first use pseudorandom permutations to implement the shuffling oracle and then use VBB obfuscation to obfuscate these pseudorandom permutations. However, this approach may be infeasible given the known negative evidence by Bitansky et al. [9].[¶] It is not clear whether our oracle can be instantiated based on any reasonable cryptographic assumptions.

Our work also addresses another critical question: Can we trade classical computations for quantum circuit depth? If this is possible, then it may be easier for near-term quantum computers to achieve quantum supremacy on practical problems. In this work, we give some negative evidence by showing a fine-grained depth separation result that $(\mathrm{BPP}^{\mathrm{BQNC}_d})^O \neq (\mathrm{BPP}^{\mathrm{BQNC}_{2d+1}})^O$. This result implies that a hybrid model is more powerful with larger quantum depth, which cannot be traded by classical computations in the relativized world. We would like to see whether a sharper separation is possible, such as d versus $d+1$, under plausible assumptions in cryptography or complexity.

1.3 Independent Work

Independent and concurrent to our work, Coudron and Menda also investigated Jozsa's conjecture and proved that the conjecture is false relative to some oracle [14]. However, they did not have the sharp separation between quantum depth d and $2d+1$ as we did. Interestingly, the oracle problem used in their work and their analysis are very different from ours, which may lead to incomparable extensions or applications in the future.

[§]Following the definition of $\mathrm{BQNC}^{\mathrm{BPP}}_{d}$ and $\mathrm{BPP}^{\mathrm{BQNC}_d}$, the gate set is the collection of one- and two-qubit gates. However, our oracle separation actually holds for any gate set even with unbounded fan-out gates. The main point is that the depth of sequential queries is fewer than the depth of the shuffling oracle.

[¶]Bitansky et al. [9] showed the impossibility of VBB obfuscation with auxiliary input for circuit families with super-polynomial pseudo-entropy.

In the following, we discuss the work by Coudron and Menda at a high-level and compare their work to ours. Coudron and Menda used the Welded Tree Problem of Childs et al. [11] as an oracle problem to show that Jozsa's conjecture does not hold in the relativized world, which leverages the separation between quantum walk and classical random walk. To show the separation, they do not need to modify the oracle problem, and the crux is a simulation argument showing that low-depth quantum computation can be simulated by (inefficient) classical computation without blowing up the number of oracle queries too much. Hence, the hardness of the hybrid models follows from the classical hardness. The simulation argument relies on the structure of the welded tree oracle and does not seem to apply to our oracle.

In contrast, our starting point is Simon's problem, which can be viewed as a separation between quantum depth 0 (i.e., classical computation) and 1 in the hybrid models, and our main idea is to lift the separation by pointer chasing and domain hiding. This allows us to prove the sharp separation between quantum depth d and $2d + 1$. To prove the separation, we generalize some techniques in cryptography (from both quantum and classical cryptography literature) to show that if the quantum depth is not large enough, then the algorithm cannot find the hidden domain of a shuffled Simon function. Hence, the hybrid models with quantum depth d are not capable to solve the d-SSP problem.

We mention that the original version of Coudron and Menda [14] only considered the hybrid model d-CQ schemes, but their analysis extends directly to establish hardness of the Welded Tree Problem for d-QC schemes. Furthermore, we believe that the techniques in both works extend to establish hardness of respective oracle problems for the natural hierarchy of hybrid models like $\mathsf{BPP}^{\mathsf{BQNC}^{\mathsf{BPP}^{\cdots}}}$ [15].

2 PROOF OVERVIEW

As mentioned in the previous section, we use *pointer chasing* and *domain hiding* in our shuffling oracle to lift the hardness of Simon's problem. Briefly, pointer chasing is a sequence of (query) instructions, where an accessed data point is required to determine the subsequent data point to be accessed, forcing the algorithm to access data points in sequence to complete tasks. Domain hiding, roughly speaking, is hiding domains with critical information in much larger domains with much additional redundant elements. In the following, we sketch our implementations of pointer chasing and domain hiding, as well as the analysis of the shuffling oracle.

Pointer Chasing & Domain Hiding. In our context, we aim to allow a random Simon function to be easily evaluated by using $d + 1$ sequential queries but make it hard by using at most d sequential queries. To implement this idea, we decompose a Simon's function f as $f_0, \ldots, f_d$, where $f_j : S_j \to S_{j+1}$ for $j = 0, \ldots, d-1$ are random shufflings of the original function f and f_d is a function such that $f = f_d \circ \cdots \circ f_0$. Ideally, to evaluate f, a standard approach is to evaluate the functions in sequence. However, it is unclear whether there are clever methods. To tackle this, our second idea is hiding the original domains $S_0, \ldots, S_d$ in larger domains $S_0^{(0)}, \ldots, S_d^{(0)}$, respectively. Intuitively, to evaluate f, one needs to learn the values of f_d on S_d, and this needs to find S_d in $S_d^{(0)}$. Note that using i rounds of queries provides sufficient information to learn S_i. Thus, the goal of the shuffling oracle is to obtain the (tight) hardness that prevents the algorithm from learning S_d before the dth round queries so one cannot evaluate f_d in d sequential queries.

Obtaining Indistinguishability from Hiding. The intuition behind our analysis is similar to the **One-way-to-Hiding (O2H)** Lemma [6, 25] in quantum cryptography, and our analysis can be viewed as a generalization of the lemma to the setting of the shuffling oracle against both hybrid models. Briefly, the O2H Lemma shows that, given any two random functions that have same

function values on most elements, except for some small unknown *hidden subset*, the probability that these two functions can be distinguished is upper bounded by the probability that the hidden set is queried. This probability, which will be called the *finding probability*, will be negligible if the hidden subset is a random subset that is small enough. The O2H Lemma implies that, given a random function, we can arbitrarily program the mappings in the hidden subset, and it cannot be distinguished by any quantum polynomial-time distinguisher. In our context, a natural approach is to argue that S_d is a hidden subset in $S_d^{(0)}$ by construction and hence, we can program the mappings in S_d in a way such that the programmed oracle has no information about f. Different from the typical use of O2H Lemma in cryptography, in the shuffling oracle, the algorithm, however, can find S_d by using d sequential queries starting from S_0 (and then S_d can be queried at the $d+1$st query). To argue that S_d is a hidden subset by construction, we need another trick as follows:

Russian-nesting-doll Trick. To bypass the difficulty mentioned above, we decompose the large set $S_d^{(0)}$ into a series of smaller subsets $S_d^{(1)}, \ldots, S_d^{(d)}$, such that $S_d^{(\ell)}$ is an exponentially small subset of $S_d^{(\ell-1)}$ for $\ell \in [d]$. For the ease of illustrating ideas, we assume $S_d \subset S_d^{(d)}$.[‖] For $f_0, \ldots, f_d$, we decompose their domains in this manner. Then, we use the O2H Lemma to show that after the first-round queries, the algorithm learns almost nothing about $f_1, \ldots, f_d$ on $S_1^{(1)}, \ldots, S_d^{(1)}$. For the second-round queries, we start from $f_1, \ldots, f_d$ on $S_1^{(1)}, \ldots, S_d^{(1)}$ and use the O2H Lemma again to argue that the algorithm learns little about $f_2, \ldots, f_d$ on $S_2^{(2)}, \ldots, S_d^{(2)}$. Finally, for the dth round of queries, the algorithm has little information about f_d on $S_d^{(d)}$ and thus has no information about f. We can think of this approach as a Russian-nesting-doll trick: The mappings f_0 on $S_0^{(0)}$ and $f_1, \ldots, f_d$ on domains out of $S_1^{(1)}, \ldots, S_d^{(1)}$ are the largest doll outside, there are d dolls, and f_d on $S_d^{(d)}$ is the smallest one we want to see. For an algorithm to see S_d, it needs to open all the dolls, and each requires one round of queries. Therefore, the algorithm cannot see the smallest doll before the dth round of queries. We illustrate this idea of the Russian-nesting-doll trick as in Figure 3.

Several ideas above are implemented by hybrid arguments, such as the O2H Lemma and the Russian-nesting-doll trick.

Next, we briefly describe the proof that d-SSP is hard for $\mathrm{QNC_d}$ circuits, d-QC schemes, and d-CQ schemes by employing the ideas above.

2.1 $\mathrm{QNC_d}$ Circuits

This is the warm-up case. Let U be a d-depth quantum circuit with components $U_1, \ldots, U_{d+1}$ at each depth. Assume that superposed queries can be made to the shuffling oracle $\mathcal{F} := (f_0, \ldots, f_d)$ after U_i for $i \in [d]$. We represent the computation as

$$U_{d+1} \circ \mathcal{F} \circ U_d \circ \cdots \circ \mathcal{F} \circ U_1.$$

Following the Russian-nesting-doll trick, we decompose the domains $S_0^{(0)}, \ldots, S_d^{(0)}$ into d subsets. Then, for the ith queries, we construct a *shadow* oracle $\mathcal{G}_i$ of $\mathcal{F}$ such that $\mathcal{G}_i$ agrees with $\mathcal{F}$ on elements out of $S_i^{(i)}, \ldots, S_d^{(i)}$ but maps the rest to $\perp$ (a specific symbol for no information). Here, we need to choose the subsets in a specific way that will be described in the formal proof. By using hybrid arguments and O2H Lemma, we inductively prove that $U_{d+1} \circ \mathcal{F} \circ U_d \circ \cdots \circ \mathcal{F} \circ U_1$ is indistinguishable from $U_{d+1} \circ \mathcal{G}_d \circ U_d \circ \cdots \circ \mathcal{G}_1 \circ U_1$.

[‖] In the formal proof of the hardness of d-SSP, there will be a small subset in S_d that is not in $S_d^{(d)}$, since a subset of S_d is learned by the classical part of the d-CQ and d-QC schemes. However, it suffices to prove the hardness result using the ideas provided here with careful analysis.

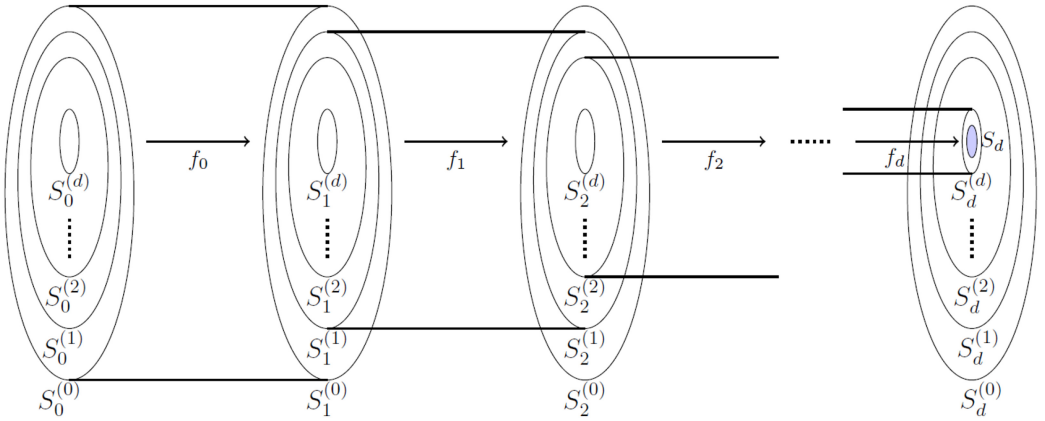

Fig. 3. The domains under the Russian-nesting-doll trick: f_0 on $S_0^{(0)}$ and $f_1, \ldots, f_d$ on $S_1^{(0)} \setminus S_1^{(1)}, \ldots, S_d^{(0)} \setminus S_d^{(1)}$ can be viewed as the "first doll," which will be opened by the first-round queries. Along this line, f_{i-1} on $S_{i-1}^{(i-1)}$ and $f_i, \ldots, f_d$ on $S_i^{(i-1)} \setminus S_i^{(i)}, \ldots, S_d^{(i-1)} \setminus S_d^{(i)}$ can be viewed as the ith doll, which will be opened by the ith-round queries.

2.2 d-QC Schemes

For a d-QC scheme, since the classical algorithm can make at most polynomially many queries, the set S_d can be found by the classical algorithm. However, the classical algorithm can only make polynomially many queries, and the remaining points that are not queried are still uniform. Therefore, we can remove whatever points learned by classical queries from the hidden sets $S_k^{(j)}$ and use O2H Lemma and the hybrid arguments as before. At the end, the algorithm does not learn S_d except for polynomially many points, and it is unlikely to learn the shift (as in the proof of classical hardness for Simon's problem).

2.3 d-CQ Schemes

Arguing that S_d is a hidden subset in $S_d^{(0)}$ against the d-CQ scheme is the most challenging part of our analysis. The main obstacle is that the measurement outcome of the quantum circuit can contain global information about the shuffling oracle, and this makes it hard to reason that the shuffling oracle is still uniformly random conditioned on the measurement outcome. However, since the measurement outcome can only be a "short classical advice," intuitively, it cannot contain too much information about the shuffling oracle.

To formalize this intuition, we (partially) generalize a presampling argument in (classical) cryptography [13, 24]. Informally, the argument states that a random function conditioned on a short classical advice string is indistinguishable from a convex combination of random functions that are fixed on a few elements for any classical algorithm that makes polynomially many queries. Ideally, we would like to generalize it to the shuffling oracle (which is more complicated than a random oracle) and show indistinguishability to a convex combination of shuffling oracles with a few points fixed for quantum algorithms with a bounded number of queries. Unfortunately, we do not know how to achieve this. Instead, we prove something that is weaker but sufficient for our purpose: We show that, given a short classical advice string, the shuffling oracle is indistinguishable from a convex combination of "*almost-uniform*" shuffling oracles with few points fixed for any quantum algorithm making *one round of quantum queries*.

This weaker statement suffices to prove our oracle separation. While there are some further subtleties, the main idea is that we apply it inductively after each round of quantum queries, and

the short advice string only fixes polynomially many random points in the shuffling oracle. Given this, we are able to show that S_d remains a hidden set against the d-CQ scheme and establish hardness of d-SSP against $\mathsf{BPP}^{\mathsf{BQNC_d}}$.

3 PRELIMINARIES

In this section, we first introduce the distance measures of quantum states. Then, we give formal definitions of the d-CQ and d-QC schemes.

3.1 State Distance

Definition 3.1. Let $\mathcal{H}$ be a Hilbert space. For any two pure states $|\psi\rangle, |\phi\rangle \in \mathcal{H}$, we define

- (Fidelity) $F(|\psi\rangle, |\phi\rangle) := |\langle\psi|\phi\rangle|$;
- (Two-norm distance) $\||\psi\rangle - |\phi\rangle\|$.

Then, we define distance measures between mixed states.

Definition 3.2. Let $\mathcal{H}$ be a Hilbert space. For any two mixed states $\rho, \rho' \in \mathcal{H}$,

- (Fidelity) $F(\rho, \rho') := tr(\sqrt{\sqrt{\rho}\rho'\sqrt{\rho}})$;
- (Trace distance) $TD(\rho, \rho') := \frac{1}{2}\operatorname{Tr}|\rho - \rho'|$;
- (Bures distance) $B(\rho, \rho') := \sqrt{2 - 2F(\rho, \rho')}$.

The probability for a quantum procedure to distinguish two states can be bounded by the Bures distance between the two states.

Claim 1. *For any two mixed states ρ and ρ', for any quantum algorithm $\mathcal{A}$ and for any classical string s,*

$$|\Pr[\mathcal{A}(\rho) = s] - \Pr[\mathcal{A}(\rho') = s]| \leq B(\rho, \rho').$$

Proof. It is well-known that $|\Pr[\mathcal{A}(\rho) = s] - \Pr[\mathcal{A}(\rho') = s]| \leq TD(\rho, \rho')$. Then,

$$\begin{aligned} TD(\rho, \rho') &\leq \sqrt{1 - F(\rho, \rho')^2} \\ &= \sqrt{\frac{1 + F(\rho, \rho')}{2}}\sqrt{2 - 2F(\rho, \rho')} \\ &\leq B(\rho, \rho'). \end{aligned}$$

□

3.2 Computational Models

The quantum oracle access to functions is defined as follows:

Definition 3.3 (Quantum Oracle Access). Let $f : S \to T$ be a function. The quantum oracle access to f is a unitary U_f defined as $U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$, where $x \in S$ and $y \in T$. Note that a quantum algorithm can make superposition queries to f. Thus, a general form for quantum queries is $U_f \sum_{x\in S, y\in T} \alpha_{x,y}|x\rangle|y\rangle = \sum_{x\in S, y\in T} \alpha_{x,y}|x\rangle|y \oplus f(x)\rangle$, where $\sum_{x\in S, y\in T} |\alpha_{x,y}|^2$.

We here define two schemes that interleave low-depth quantum circuits and classical computers. The first scheme is called **d-depth quantum-classical scheme (d-QC scheme)** and the second scheme is **d-depth classical-quantum scheme (d-CQ scheme)**.

We say a set of gates forms a layer if all the gates in the set operate on disjoint qubits. Gates in the same layer can be parallelly applied. We define the number of layers in a circuit as the *d*epth of the circuit. In the following, we define circuit families that have circuit depth d as in References [21, 23]:

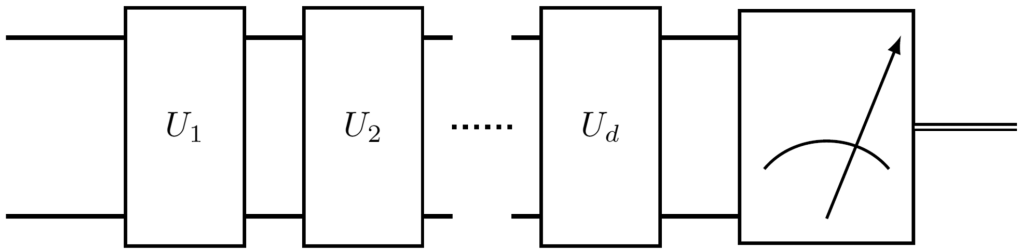

Fig. 4. The $\mathrm{QNC_d}$ circuit: The single-line wires are quantum wires, and the double-line wire is a classical wire.

Definition 3.4 ($d(n)$-depth Quantum Circuit $\mathrm{QNC_d}$). Let $d(n)$ be a function of n. A $\mathrm{QNC_d}$ quantum circuit family $\{C_n : n > 0\}$ is defined as below:

- There exists a polynomial p such that for all $n > 0$, C_n operates on n input qubits and $p(n)$ ancilla qubits;
- for $n > 0$, C_n has the initial state $|0^{n+p(n)}\rangle$, which consists of $d(n)$ layers of one- and two-qubit gates,[**] and measures all qubits after the last layer.

Here, the depth $d(n)$ is a function of the inputs size n. In this work, we are mainly interested in the case where $d(n) = \mathrm{polylog}\, n$. (The input size is significantly greater than the circuit depth.) For simplicity, we will just use d in the rest of the article.

We illustrate a $\mathrm{QNC_d}$ circuit as in Figure 4, where U_i for $i \in [d]$ is a unitary that can be implemented by one layer of one- and two-qubit gates, and the last computational unit is a qubit-wise measurement in the standard basis.

Remark 1 ($\mathrm{QNC_d}$ with Oracle Access to f). A d-depth quantum circuit with oracle access to some classical function $f : \{0,1\}^n \to \{0,1\}^m$ is a sequence of unitaries

$$U_{d+1} U_f U_d \cdots U_f U_2 U_f U_1, \tag{1}$$

where U_i's are depth-1 quantum circuit and U_f is a unitary such that $U_f|x\rangle|b\rangle = |x\rangle|b \oplus f(x)\rangle$ for all $x \in \{0,1\}^n$ and $b \in \{0,1\}^m$ (as in Definition 3.3). We add an additional depth-1 unitary U_{d+1} to the computational model to process the quantum state after the last U_f. It is worth noting that each U_f can operate on multiple quantum queries.

In some studies, QNC is also used to refer the set of languages decided by the quantum circuits. For clarity, we define the set of languages decided by $\mathrm{QNC_d}$ as $\mathrm{BQNC_d}$ as follows:

Definition 3.5 ($\mathrm{BQNC_d}$). The set of languages $L = \{L_n : n > 0\}$ for which there exists a circuit family $\{C_n : n > 0\} \in QNC_d$ such that for $n > 0$, for any x where $|x| = n$,

- if $x \in L_n$, then $\Pr[C_n(x) = 1] \geq 2/3$;
- otherwise, $\Pr[C_n(x) = 1] \leq 1/3$.

Then, we define the quantum analogue of the NC class as in Reference [16]. One can check the textbook [7] for the formal definition of NC.

Definition 3.6 ($\mathrm{BQNC^k}$). The set of languages $L = \{L_n : n > 0\}$ for which there exists a circuit family $\{C_n : n > 0\} \in \mathrm{QNC}_d$ for $d = O(\log^k n)$ such that for $n > 0$, for any x where $|x| = n$,

[**]Here, we choose the gateset to be all one- and two-qubit gates. One can also define $\mathrm{QNC_d}$ by considering other universal gate sets.

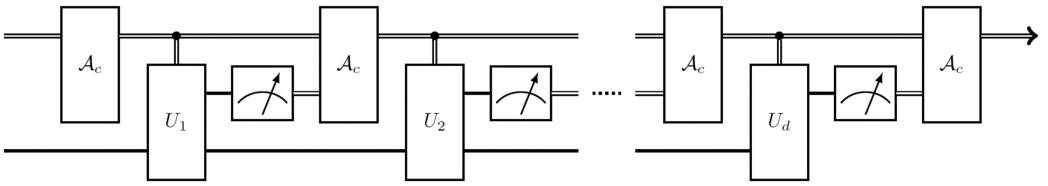

Fig. 5. d-depth quantum-classical (d-QC) scheme: The single-line wires stands for quantum wires, and the double-line wires are classical wires.

- if $x \in L_n$, then $\Pr[C_n(x) = 1] \geq 2/3$;
- otherwise, $\Pr[C_n(x) = 1] \leq 1/3$.

For simplicity, we define BQNC as the set of languages that can be decided by poly log-depth quantum circuit, that is $\mathsf{BQNC} := \bigcup_k \mathsf{BQNC}^k$.

We represent the process in Equation (1) as follows: Let $|0,0\rangle_{QA}|0\rangle_W$ be the initial state, where registers Q and A consist of qubits used to interact with the oracle and register W consists of the rest of the qubits the process uses as the working space. Let

$$U_1|0,0\rangle_{QA}|0\rangle_W = \sum_{x,y} c(x,y)|x,y\rangle_{QA}|z(x,y)\rangle_W$$

be the quantum message before applying the first U_f. After applying U_f,

$$U_f \sum_{x,y} c(x,y)|x,y\rangle_{QA}|z(x,y)\rangle_W = \sum_{x,y} c(x,y)|x,y \oplus f(x)\rangle_{QA}|z(x,y)\rangle_W.$$

We represent the state $U_{d+1}U_fU_d \cdots U_fU_1|0,0\rangle_{QA}|0\rangle_W$ in this way, and for simplicity, we rewrite $U_{d+1}U_fU_d \cdots U_fU_1$ as $U_{d+1}fU_d \cdots fU_1$ in the rest of the article.

3.2.1 d-QC Schemes and $\mathsf{BQNC}_{\mathrm{d}}^{\mathsf{BPP}}$. The first family of hybrid quantum-classical algorithms we consider is d-QC schemes, which is a generalized model for d-depth MBQC and can be represented as the following sequence:

$$\mathcal{A}_c \xrightarrow{c} (\Pi_{0/1} \otimes I) \circ U^{(1)} \xrightarrow{c} \mathcal{A}_c \xrightarrow{c/q} \cdots \xrightarrow{c/q} (\Pi_{0/1} \otimes I) \circ U^{(d)} \xrightarrow{c} \mathcal{A}_c, \tag{2}$$

where $\mathcal{A}_c$ is a randomized algorithm, $U^{(i)}$ is a depth-one quantum circuit, and $\Pi_{0/1}$ is a projective measurement in the standard basis on a subset of the qubits. The arrows $\xrightarrow{c}$ and $\xrightarrow{q}$ indicate the classical and quantum messages transmitted, and $\xrightarrow{c/q}$ indicates that there can be both quantum and classical messages transmitted. We illustrate the scheme as in Figure 5.

In this model, after applying $U^{(i)}$, one measures a subset of qubits, uses the measurement outcome to generate $U^{(i+1)}$ by the classical polynomial algorithm $\mathcal{A}_c$, and then applies $U^{(i+1)}$ to the rest of the qubits. For example, in the MBQC, the measurement outcome after $U^{(i)}$ can decide the angles of rotations to be applied in $U^{(i+1)}$ by some simple mappings. Let $L^{(i)}$ be the procedure $\mathcal{A}_c \to c(\Pi_{0/1} \otimes I) \circ U^{(i)}$. Then, we rewrite Equation (2) as

$$L^{(1)} \xrightarrow{c/q} \cdots \xrightarrow{c/q} L^{(d)} \xrightarrow{c} \mathcal{A}_c. \tag{3}$$

Then, we define the languages that can be decided by d-QC schemes.

Definition 3.7 ($\mathsf{BQNC}_{\mathrm{d}}^{\mathsf{BPP}}$). The set of languages $\mathcal{L} = \{\mathcal{L}_n : n > 0\}$ for which there exists a family of d-QC schemes $\{\mathcal{A}_n : n > 0\}$ such that for $n > 0$, for any x where $|x| = n$,

- if $x \in \mathcal{L}_n$, then $\Pr[\mathcal{A}_n(x) = 1] \geq 2/3$;
- otherwise, $\Pr[\mathcal{A}_n(x) = 1] \leq 1/3$.

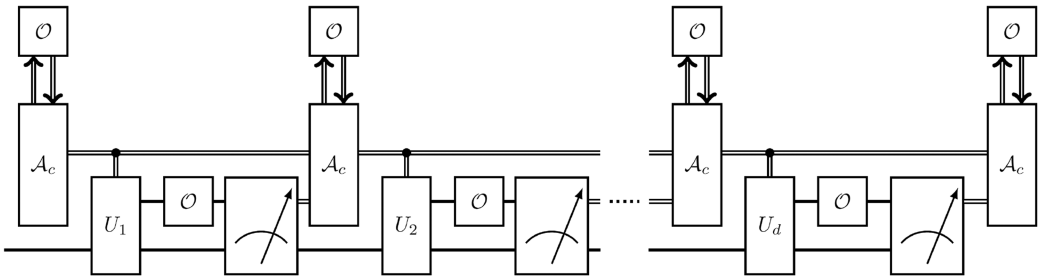

Fig. 6. d-QC scheme with access to an oracle O.

Let $\mathcal{A}$ be a d-QC scheme with access to oracle O. We represent $\mathcal{A}^O$ as a sequence of operators:

$$(L^{(1)})^O \xrightarrow{c/q} \cdots \xrightarrow{c/q} (L^{(d)})^O \xrightarrow{c} \mathcal{A}_c^O, \tag{4}$$

where $(L^{(i)})^O := \mathcal{A}_c^O \xrightarrow{c} (\Pi_{0/1} \otimes I) \circ OU^{(i)}$. We illustrate the scheme as in Figure 6.

Remark 2. Each $U^{(i)}$ can prepare a set of queries to O; however, these queries cannot depend on each other's outputs. So, one can also say that $U^{(i)}$ can make parallel quantum queries to O. Since we are considering d-depth quantum circuits, there can be $d \cdot \text{poly}(n)$ quantum queries in total.

Definition 3.8 ($(\text{BQNC}_{\text{d}}^{\text{BPP}})^O$*).* The set of languages $\mathcal{L}^O := \{\mathcal{L}_n^O : n > 0\}$ for which there exists a family of d-QC schemes $\{\mathcal{A}_n^O : n > 0\}$ such that for $n > 0$, for any x where $|x| = n$,

- if $x \in \mathcal{L}_n^O$, then $\Pr[\mathcal{A}_n^O(x) = 1] \geq 2/3$;
- otherwise, $\Pr[\mathcal{A}_n^O(x) = 1] \leq 1/3$.

Similar to the definition of BQNC, we define BQNC^{BPP} as a set of languages that can be decided by a family of d-QC schemes with $d = O(\text{poly} \log n)$.

3.2.2 d-CQ Schemes and $\text{BPP}^{\text{BQNC}_{\text{d}}}$. We define d-CQ schemes as a family of hybrid quantum-classical algorithms with classical algorithms that have access to QNC_d circuits during the computation. We represent a d-CQ scheme as follows:

$$\mathcal{A}_{c,1} \xrightarrow{c} \Pi_{0/1} \circ C \xrightarrow{c} \cdots \xrightarrow{c} \mathcal{A}_{c,m} \xrightarrow{c} \Pi_{0/1} \circ C \xrightarrow{c} \mathcal{A}_{c,m+1}, \tag{5}$$

where $m = \text{poly}(n)$, $A_{c,i}$ is a randomized algorithm, C is a d-depth quantum circuit, and $\Pi_{0/1}$ is the standard-basis measurement. We illustrate the scheme as in Figure 7.

The classical algorithm can perform queries to the QNC_d circuit and then use the measurement outcomes from the circuit as part of the input to the following procedures: We let $L^{(i)} := \mathcal{A}_{c,i} \xrightarrow{c} \Pi_{0/1} \circ C$ and rewrite Equation (5) as

$$L^{(1)} \xrightarrow{c} L^{(2)} \xrightarrow{c} \cdots \xrightarrow{c} L^{(m)} \xrightarrow{c} \mathcal{A}_{c,m+1}.$$

Definition 3.9 ($\text{BPP}^{\text{BQNC}_{\text{d}}}$*).* The set of languages $\mathcal{L} = \{\mathcal{L}_n : n > 0\}$ for which there exists a family of d-CQ schemes $\{\mathcal{A}_n : n > 0\}$ such that for $n > 0$, for any x where $|x| = n$,

- if $x \in \mathcal{L}_n$, then $\Pr[\mathcal{A}_n(x) = 1] \geq 2/3$;
- otherwise, $\Pr[\mathcal{A}_n(x) = 1] \leq 1/3$.

Let $\mathcal{A}$ be a d-CQ scheme with access to some oracle O. We represent $\mathcal{A}^O$ as

$$(L^{(1)})^O \xrightarrow{c} (L^{(2)})^O \xrightarrow{c} \cdots \xrightarrow{c} (L^{(m-1)})^O \xrightarrow{c} \left(\mathcal{A}_c^{(m)}\right)^O, \tag{6}$$

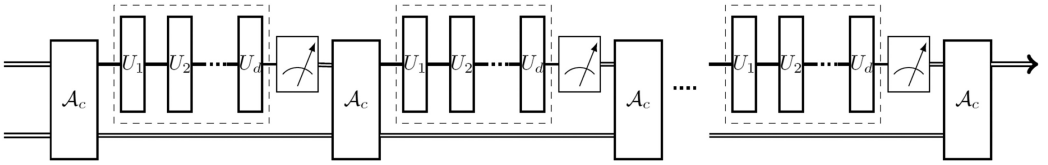

Fig. 7. The d-depth classical-quantum (d-CQ) scheme.

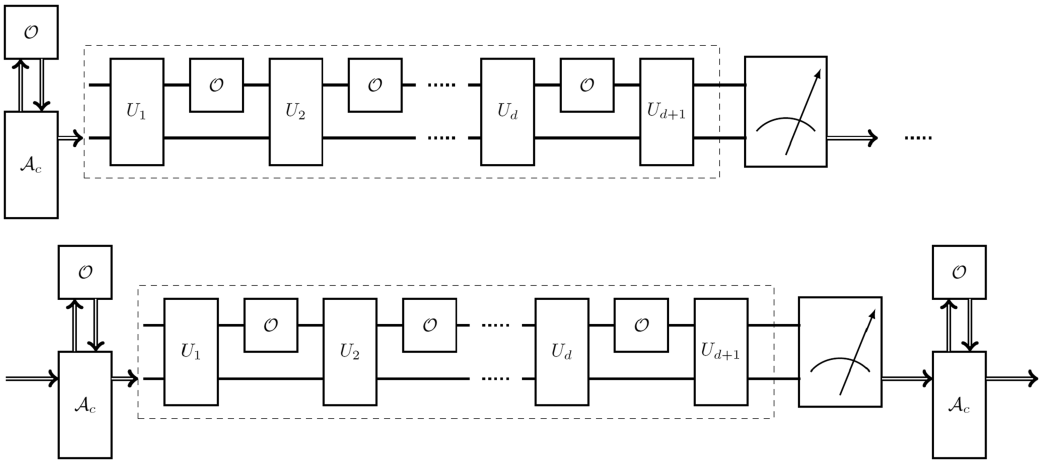

Fig. 8. d-CQ scheme with access to an oracle O.

where $(L^{(i)})^O := (\mathcal{A}_{c,i})^O \xrightarrow{c} \Pi_{0/1} \circ (U^{(d)} O U^{(d-1)} \cdots O U^{(1)})$. We illustrate the scheme as in Figure 8.

Definition 3.10 (($\mathsf{BPP}^{\mathsf{BQNC_d}}$)^O). The set of languages $\mathcal{L}^O = \{\mathcal{L}_n^O : n > 0\}$ for which there exists a family of d-CQ schemes $\{\mathcal{A}_n^O : n > 0\}$ such that for $n > 0$, for any x where $|x| = n$,

- if $x \in \mathcal{L}_n^O$, then $\Pr[\mathcal{A}_n^O(x) = 1] \geq 2/3$;
- otherwise, $\Pr[\mathcal{A}_n^O(x) = 1] \leq 1/3$.

We define $\mathsf{BPP}^{\mathsf{BQNC}}$ as a set of languages that can be decided by a family of d-CQ schemes with $d = O(\mathrm{polylog}\, n)$.

The main differences between d-CQ and d-QC schemes are that (1) a d-QC scheme can transmit quantum messages from one layer to the next, but a d-CQ scheme can only send classical messages, and (2) a d-QC scheme has at most d layers, but a d-CQ scheme may have $m \times d$ layers. According to these observations, these two schemes seem to be incomparable.

The notation $\mathsf{BQNC}^{\mathsf{BPP}}$ and $\mathsf{BPP}^{\mathsf{BQNC}}$ may not be standard in quantum complexity theory. For instance, one may expect that the quantum circuit can make superposed queries to the BPP part, but we only allow classical queries. The reason we chose these non-standard names is that $\mathsf{BPP}^{\mathsf{BQNC}}$ and $\mathsf{BQNC}^{\mathsf{BPP}}$ are more intuitive for readers to capture the ideas of the two models and can help readers to figure out the differences between these two models, as we have mentioned in the previous paragraph.

4 THE D-SHUFFLING SIMON'S PROBLEM (D-SSP)

In this section, we define the oracle and the corresponding oracle problem that separates poly(n)-depth quantum algorithms from d-QC and d-CQ schemes. We summarize the notation in this section in Table 1.

4.1 The Shuffling Oracle

For any two sets X and Y, let $\mathcal{P}(X, Y)$ be the set of one-to-one functions from X to Y. For example, $\mathcal{P}(\mathbb{Z}_2^n, \mathbb{Z}_2^n)$ is the set of all permutations over $\mathbb{Z}_2^n$. Let $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ be an arbitrary function and $d \in \mathbb{N}$. We define (d, f)-Shufflings as follows:

Definition 4.1 ((d, f)-Shuffling). Let $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ be an arbitrary function and $d \in \mathbb{N}$. A (d, f)-shuffling is defined by a set of functions $\mathcal{F} := (f_0, \ldots, f_d)$, where $f_0, \ldots, f_{d-1} \in P(\mathbb{Z}_2^{(d+2)n}, \mathbb{Z}_2^{(d+2)n})$ can be any permutations over $\mathbb{Z}_2^{(d+2)n}$, and then f_d is chosen to be the function satisfying the following properties: Let $S_d := \{f_{d-1} \circ \cdots \circ f_0(x') : x' = 0^{(d+1)n}0^n, 0^{(d+1)n}0^{n-1}1, \ldots, 0^{(d+1)n}1^n\}$.

- For $x \in S_d$, let $f_{d-1} \circ \cdots \circ f_0(x') = x$, and we choose the function $f_d : S_d \to [0, 2^n - 1]$ satisfying that $f_d \circ f_{d-1} \circ \cdots \circ f_0(x') = f(x')$.
- For $x \notin S_d$, we let $f_d(x) = \perp$.

We let $\mathbf{SHUF}(d, f)$ be the set of all (d, f)-shufflings of f. For simplicity, we denote f_d on the subdomain S_d as f_d^*.

In this article, we consider the case that a (d, f)-shuffling is given randomly. One of the most natural ways is sampling a shuffling $\mathcal{F}$ uniformly at random from $\mathbf{SHUF}(d, f)$. We describe the sampling procedure as below.

Definition 4.2 ($\mathcal{D}(f, d)$). Draw $f_0, \ldots, f_{d-1}$ uniformly at random from $\mathcal{P}(\mathbb{Z}_2^{(d+2)n}, \mathbb{Z}_2^{(d+2)n})$ and then choose f_d^* such that $f_d^* \circ \cdots \circ f_0(x) = f(x)$ for $x \in \mathbb{Z}_2^n$.

Fix $d \in \mathbb{N}$ and a function $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$, we define d-shuffling oracle as a random oracle that chooses a (d, f)-shuffling uniformly randomly from $\mathbf{SHUF}(d, f)$ according to $\mathcal{D}(f, d)$.

Definition 4.3 (d-Shuffling oracle $O_{\mathrm{unif}}^{f,d}$). Let f be an arbitrary function from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$. Let $d \in \mathbb{N}$. We define $O_{\mathrm{unif}}^{f,d}$ as randomly choosing a (d, f)-shuffling $\mathcal{F}$ from $\mathbf{SHUF}(d, f)$ according to $\mathcal{D}(f, d)$.

If we sample $\mathcal{F}$ according to the distribution in Definition 4.2, then only f_d^* encodes the information of f, while $(f_0, \ldots, f_{d-1})$ are just random one-to-one functions. We call f_d^* the **core function** of f.

For simplifying our proofs, we here define paths in the shuffling oracle. The concept of paths will be used when we consider d-QC and d-CQ schemes.

Definition 4.4 (Path in $O_{\mathrm{unif}}^{f,d}$). Let $\mathcal{F} \in \mathbf{SHUF}(d, f)$. We say $(x_0, \ldots, x_{d+1})$ is a **path** in $\mathcal{F}$ if $f_i(x_i) = x_{i+1}$ for $i = 0, \ldots, d$.

Now, we describe the oracle access to the shuffling oracle $O_{\mathrm{unif}}^{f,d}$. Let $|\phi\rangle$ be the input state to $O_{\mathrm{unif}}^{f,d}$, which we represent in the form

$$|\phi\rangle := \sum_{\mathbf{X}_0, \ldots, \mathbf{X}_d} c(\mathbf{X}_0, \ldots, \mathbf{X}_d) \left(\bigotimes_{i=0}^{d} |i, \mathbf{X}_i\rangle \right)_{\mathbf{R}_Q} \otimes |0\rangle_{\mathbf{R}_N} \otimes |w(\mathbf{X}_0, \ldots, \mathbf{X}_d)\rangle_{\mathbf{R}_W}, \tag{7}$$

where $|w(\mathbf{X}_0, \ldots, \mathbf{X}_d)\rangle$'s are some arbitrary states and $\mathbf{X}_i$ is a set of elements in the domain of f_i. The queries in the register $\mathbf{R}_Q$ and the ancillary qubits in the register $\mathbf{R}_N$ will be processed by the oracle, while the remaining local working qubits in the register $\mathbf{R}_W$ are unchanged and hold by the algorithm; the state $|i, \mathbf{X}_i\rangle$ denotes the set of parallel queries to function f_i. Note that the algorithms (d-CQ and d-QC schemes) we consider can make multiple quantum queries to $O_{\mathrm{unif}}^{f,d}$ in each round, as we discussed in Remark 2 and Remark 1.

We let $\mathcal{F} \in \mathbf{SHUF}(d, f)$ be sampled according to $\mathcal{D}(f, d)$, then

$$\mathcal{F}|\phi\rangle := \sum_{\mathbf{X}_0,\ldots,\mathbf{X}_d} c(\mathbf{X}_0,\ldots,\mathbf{X}_d)\left(\bigotimes_{i=0}^{d}|i,\mathbf{X}_i\rangle|f_i(\mathbf{X}_i)\rangle\right)_{\mathbf{R}_Q,\mathbf{R}_N} \otimes |w(\mathbf{X}_0,\ldots,\mathbf{X}_d)\rangle_{\mathbf{R}_W}.$$

Applying $O^{f,d}_{\text{unif}}$ to $|\phi\rangle$ gives a state $\mathcal{F}|\phi\rangle$ with $\mathcal{F}$ sampled from $\mathbf{SHUF}(d, f)$ according to $\mathcal{D}(f, d)$. Note that after $\mathcal{F}$ is sampled by the first query, the rest of the queries will be consistent with the previous queries, i.e., $O^{f,d}_{\text{unif}}$ will not resample $\mathcal{F}$.

4.2 Shuffling Simon's Problem

We recall the definitions of Simon's function and Simon's problem.

Definition 4.5 (Simon's function). A two-to-one function $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ for $n \in \mathbb{N}$ is a Simon's function if there exists an $s \in \mathbb{Z}_2^n$ such that $f(x) = f(x + s)$ for all $x \in \mathbb{Z}_2^n$.

Definition 4.6 (Simon's problem). Let $n \in \mathbb{N}$. Let $\mathbf{F}$ be the set of all Simon's functions from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$. Given f chosen uniformly randomly from $\mathbf{F}$, the problem is to find the hidden shift s of f.

The decision version of Simon's problem is as follows:

Definition 4.7 (Decision Simon's problem). Let $n \in \mathbb{N}$. Let $\mathbf{F}$ be the set of all Simon's functions from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$. Choose f to be either a random Simon's function from $\mathbf{F}$ or a random one-to-one function from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$ with equal probability; the problem is to decide which case f is.

Both problems have been shown to be hard classically and can be solved in quantum polynomial time. We define the d-SSP by combining Simon's problem and the shuffling oracle.

*Definition 4.8 (d-**Shuffling Simon's Problem** (d-SSP)).* Let $d \in \mathbb{N}$. Let $f : \{0,1\}^n \to \{0,1\}^n$ be a random Simon's function or a random one-to-one function with equal probability. Given access to the d-shuffling oracle $O^{f,d}_{\text{unif}}$ of f, the problem is to decide which case f is.

The search version of the d-SSP is as follows:

Definition 4.9 (Search d-SSP). Let $d \in \mathbb{N}$. Let $f : \{0,1\}^n \to \{0,1\}^n$ be a random Simon's function. Given access to the d-shuffling oracle $O^{f,d}_{\text{unif}}$ of f, the problem is to find the hidden shift of f.

Then, we define an oracle O and a language $\mathcal{L}(O)$ corresponding to the d-SSP.

Definition 4.10. Let $\{f_i : \mathbb{Z}_2^i \to \mathbb{Z}_2^i;\ i > 0\}$ be the set of functions satisfying the promise of the decision Simon's problem. Let $O := \{O^{f_i,d(i)}_{unif} : i > 0\}$, where $O^{f_i,d(i)}_{unif}$ is the $d(i)$-shuffling oracle for the function f_i and $d(i) = i$. The language is defined as follows:

$$\mathcal{L}(O) := \{1^n : f_n \text{ is a Simon's function.}\}$$

Remark 3. If the d-SSP is intractable for any d-CQ or d-QC scheme for any d, then

$$\mathcal{L}(O) \notin (\mathsf{BPP}^{\mathsf{BQNC}})^O \cup (\mathsf{BQNC}^{\mathsf{BPP}})^O.$$

This is because $d(n) = \Theta(n)$ is asymptotically greater than $\log^k n$ for any constant k. We can prove it by contradiction. If there exist d-CQ or d-QC algorithms that can decide $\mathcal{L}(O)$, then we can use that algorithm to solve the d-SSP.

We can show that the d-SSP can be solved with a $2d+1$-depth quantum circuit. The idea is using Simon's algorithm and erasing the queries on the path.

Theorem 4.11. *The d-SSP and the search d-SSP can be solved by a QNC_{2d+1} circuit with classical post-processing.*

Proof. The d-SSP can be solved via Simon's algorithm. We show the proof here. Let f be a Simon function or a one-to-one function. Given d-shuffling oracle $O^{f,d}_{\text{unif}}$ of f, let $(f_0, \ldots, f_d)$ be the (d, f)-shuffling sampled by $O^{f,d}_{\text{unif}}$. Then, we can have the following algorithm that decides whether f is a Simon function and finds the hidden shift if f is a Simon function:

$$
\begin{aligned}
\sum_{x\in\mathbb{Z}_2^n} |x\rangle|0,\ldots,0\rangle &\xrightarrow{f_0} \sum_{x\in\mathbb{Z}_2^n} |x\rangle|f_0(x),\ldots,0\rangle \\
&\xrightarrow{f_1} \sum_{x\in\mathbb{Z}_2^n} |x\rangle|f_0(x), f_1(f_0(x)),\ldots,0\rangle \\
&\xrightarrow{f_d} \sum_{x\in\mathbb{Z}_2^n} |x\rangle|f_0(x), f_1(f_0(x)),\ldots,f(x)\rangle \\
&\xrightarrow{\text{measure}} \frac{1}{\sqrt{2}}(|x\rangle|f_0(x),\ldots,f_{d-1}(\cdots f_0(x))\rangle \\
&\quad +|x+s\rangle|f_0(x+s),\ldots,f_{d-1}(\cdots f_0(x+s)))\rangle)|f(x)\rangle \\
&\xrightarrow{\text{uncompute } f_0,\ldots,f_{d-1}} \frac{1}{\sqrt{2}}\left(|x\rangle + |x+s\rangle\right)|f(x)\rangle \\
&\xrightarrow{QFT} \frac{1}{\sqrt{2^n}}\sum_{j\in\mathbb{Z}_2^n}((-1)^{x\cdot j} + (-1)^{(x+s)j})|j\rangle.
\end{aligned}
$$

When $s \cdot j = 0$, measuring the first register outputs j with non-zero probability. However, when $s \cdot j = 1$, the probability that measurement outputs j is zero. Therefore, by sampling $O(n)$ copies of j's, one can find s, which is orthogonal to all j's with high probability.

This algorithm can be implemented by a QNC_{2d+1} circuit with classical post-processing. In particular, the algorithm only requires $2d + 1$ sequential quantum queries to f, and one can defer the intermediate measurement to the end of the algorithm. Note that $2d + 1$ sequential quantum queries to f can be implemented by a QNC_{2d+1} circuit (see Remark 1). □

Theorem 4.11 directly implies that the language defined in Definition 4.10 is in BQP relative to O.

5 ANALYZING THE SHUFFLING ORACLE

In this section, we are going to prove some properties related to the shuffling oracle. We first define a sequence of subsets that are in the domains of $f_0, \ldots, f_d$.

Definition 5.1 ($\overline{S}$). Let $S_0 = \{0, \ldots, 2^n - 1\}$. For $j = 0, \ldots, d-1$, let $S_{j+1} = f_j \circ f_{j-1} \circ \cdots \circ f_0(S_0)$. We define $\overline{S} := (S_0, \ldots, S_d)$.

We define a sequence of hidden sets S corresponding to the shuffling oracle.

Definition 5.2 (The hidden sets S*).* Let $d, n \in \mathbb{N}$, $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$, and $\mathcal{F}$ be a (d, f)-shuffling of f. Then, we define the sequence of hidden sets $\mathrm{S} = (\overline{S}^{(0)}, \ldots, \overline{S}^{(d)})$ as follows:

(1) Let $S_j^{(0)} = \mathbb{Z}_2^{(d+2)n}$ for $j = 0, \ldots, d$. We define $\overline{S}^{(0)} = (S_0^{(0)}, \ldots, S_d^{(0)})$.
(2) For $\ell = 1, \ldots, d$, we define $\overline{S}^{(\ell)} = (S_\ell^{(\ell)}, \ldots, S_d^{(\ell)})$ as follows: Choose $S_\ell^{(\ell)}$ a uniformly random subset in $S_\ell^{(\ell-1)}$ such that $\frac{|S_j^{(\ell)}|}{|S_j^{(\ell-1)}|} \leq \frac{1}{2^n}$. Then, $S_{j+1}^{(\ell)} = f_j(S_j^{(\ell)})$ for $j = \ell, \ldots, d-1$.

Note that S is a concept that we will use to show that a d-depth quantum circuit cannot successfully evaluate f_d^* with high probability. Hence, we will choose S in the ways such that some properties are satisfied, depending on the computational models we are considering. We will see how to construct S in the following sections.

With the concept of S, we can introduce the notation *Shadow*, which we will use to analyze the shuffling oracle.

Definition 5.3 (Shadow function). Let $\mathcal{F} := (f_0, \ldots, f_d)$ be a (d, f)-shuffling of $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$. Fix the hidden sets $\mathrm{S} := (\overline{S}^{(0)}, \ldots, \overline{S}^{(d)})$. The shadow $\mathcal{G}$ of $\mathcal{F}$ in $\overline{S}^{(\ell)} = (S_\ell^{(\ell)}, \ldots, S_d^{(\ell)})$ is as follows: For $j = \ell, \ldots, d$, let g_j be the function such that if $x \in S_j^{(\ell)}$, $g_j(x) = \perp$; otherwise, $g_j(x) = f_j(x)$. We let $\mathcal{G} := (f_0, \ldots, f_{\ell-1}, g_\ell, \ldots, g_d)$.

We can also represent a (d, f)-shuffling $\mathcal{F}$ in terms of mappings corresponding to elements in $\overline{S}^{(\ell)}$ of S.

Definition 5.4 ($\mathcal{F}^{(\ell)}$ and $\hat{\mathcal{F}}^{(\ell)}$). For $\ell = 1, \ldots, d$, we let $f_j^{(\ell)}$ be f_j on $S_j^{(\ell-1)} \setminus S_j^{(\ell)}$ and $\hat{f}_j^{(\ell)}$ be f_j on $S_j^{(\ell)}$. Then, we define

$$\mathcal{F}^{(1)} := \left(f_0, f_1^{(1)}, \ldots, f_d^{(1)}\right), \quad \mathcal{F}^{(d+1)} := \left(\hat{f}_d^{(d)}\right), \text{ and}$$
$$\mathcal{F}^{(\ell)} := \left(\hat{f}_{\ell-1}^{(\ell)}, f_\ell^{(\ell)}, \ldots, f_d^{(\ell)}\right)$$

for $\ell = 2, \ldots, d$. Also, we define

$$\hat{\mathcal{F}}^{(0)} := (f_0, \ldots, f_d) \quad \text{and} \quad \hat{\mathcal{F}}^{(\ell)} := \left(\hat{f}_\ell^{(\ell)}, \ldots, \hat{f}_d^{(\ell)}\right)$$

for $\ell = 1, \ldots, d$.

We can say that $\hat{\mathcal{F}}^{(\ell)}$ is the mapping of elements in $\overline{S}^{(\ell)}$, and $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(\ell)}$ are the mappings out of $\overline{S}^{(\ell)}$ for $\ell \in [d]$.

We rewrite $\mathcal{F}$ in the following form:

$$\mathcal{F} := (\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(d+1)}).$$

This representation will be convenient for our analysis. There are several facts corresponding to this representation.

Observation 1. *Let $\mathcal{F}$ be a uniform (d, f)-shuffling of some function f.*

- *f is determined by $\mathcal{F}^{(d+1)} := (\hat{f}_d^{(d)})$.*
- *Let $\mathcal{G} := (f_1, \ldots, f_{\ell-1}, g_\ell, \ldots, g_d)$ be the shadow of $\mathcal{F}$ in $\overline{S}^{(\ell)}$, $\mathcal{G}$ must be consistent with $\mathcal{F}$ on $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(\ell)}$, and maps the elements in $\overline{S}^{(\ell)}$ to $\perp$. In other words, $\hat{\mathcal{F}}^{(\ell)}$ is totally blocked by $\perp$ when given $\mathcal{G}$.*
- *For $\ell = 1, \ldots, d$, conditioned on $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(\ell)}$, the function $\hat{f}_j^{(\ell)}$ is still drawn uniformly randomly from $\mathcal{P}(S_j^{(\ell)}, S_{j+1}^{(\ell)})$ for $j = \ell, \ldots, d-1$ according to the definition of $\mathcal{D}(f, d)$.*

Remark 4. In this work, we will say a quantum state ρ or a classical bit string $\bar{s}$ is **uncorrelated** to $\mathcal{F}^{(\ell)}$ if we replace $\mathcal{F}^{(\ell)}$ by any other function, the process that outputs ρ or $\bar{s}$ will not change the output distribution. However, if a quantum state ρ or a bit string $\bar{s}$ is correlated to $\mathcal{F}^{(\ell)}$, then we will assume a process that is given ρ (or $\bar{s}$) knows everything about $\mathcal{F}^{(\ell)}$ without loss of generality.

We summarize the notations above in Table 1.

5.1 Semi-classical Shuffling Oracle

In this section, we combine the concepts of "semi-classical" oracle introduced in Reference [6] and the shuffling oracle together. Roughly, the semi-classical oracle applies an additional unitary that identifies x's that are in a specific subset before applying the oracle. This concept helps us to quantify the information the algorithm gains for the specific subset from the quantum queries.

Definition 5.5 ($U^{\mathcal{F}\setminus\overline{S}^{(\ell)}}$). Let f be an arbitrary function and $\mathcal{F}$ be a random (d, f)-shuffling of f. Let $\mathbf{S} := (\overline{S}^{(0)}, \ldots, \overline{S}^{(d)})$ be a sequence of hidden sets. Let U be single-depth quantum circuit. For $\ell \in [d]$, we define $U^{\mathcal{F}\setminus\overline{S}^{(\ell)}}$ to be a unitary operating on registers $(\mathbf{R}, \mathbf{I})$ where $\mathbf{I}$ is a single-qubit register. $U^{\mathcal{F}\setminus\overline{S}^{(\ell)}}$ simulates $\mathcal{F}U$ and that:

> Before applying $\mathcal{F}$, $U^{\mathcal{F}\setminus\overline{S}^{(\ell)}}$ first applies $U_{\overline{S}^{(\ell)}}$ on $(\mathbf{R}, \mathbf{I})$ and then performs $\mathcal{F}$. Here, $U_{\overline{S}^{(\ell)}}$ is defined by:

$$U_{\overline{S}^{(\ell)}}|(\ell, \mathbf{X}_\ell), \ldots, (d, \mathbf{X}_d)\rangle_{\mathbf{R}}|b\rangle_{\mathbf{I}} := \begin{cases} |(\ell, \mathbf{X}_\ell), \ldots, (d, \mathbf{X}_d)\rangle|b\rangle \text{ if every } \mathbf{X}_i \cap S_i^{(\ell)} = \phi, \\ |(\ell, \mathbf{X}_\ell), \ldots, (d, \mathbf{X}_d)\rangle|b+1 \mod 2\rangle \text{ otherwise.} \end{cases}$$

In other words, for any state $|\psi\rangle_{\mathbf{R},\mathbf{I}}$,

$$U^{\mathcal{F}\setminus\overline{S}^{(\ell)}}|\psi\rangle := \mathcal{F}U_{\overline{S}^{(\ell)}}U|\psi\rangle.$$

In the following, we define a quantity that is the probability that the parallel queries are in a particular hidden set $\overline{S}^{(\ell)}$:

Definition 5.6 ($\Pr(\text{find } \overline{S}^{(k+1)} : U^{\mathcal{F}\setminus\overline{S}^{(k+1)}}, \rho)$). Let $k, d \in \mathbb{N}$ and $k+1 < d$. Let U be a single-depth quantum circuit and ρ be any input state. We define

$$\Pr[\text{find } \overline{S}^{(k+1)} : U^{\mathcal{F}\setminus\overline{S}^{(k+1)}}, \rho] := \mathbb{E}\left[\mathrm{Tr}\left((I_{\mathbf{R}} \otimes (I - |0\rangle\langle 0|))_{\mathbf{I}} \circ U^{\mathcal{F}\setminus\overline{S}^{(k+1)}} \circ \rho\right)\right].$$

Following Definition 5.6, we let $|\psi\rangle$ be a pure state and U be a single-depth quantum circuit. Then,

$$U^{\mathcal{F}\setminus\overline{S}^{(\ell)}}|\psi\rangle_{\mathbf{R}}|0\rangle_{\mathbf{I}} := |\phi_0\rangle_{\mathbf{R}}|0\rangle_{\mathbf{I}} + |\phi_1\rangle_{\mathbf{R}}|1\rangle_{\mathbf{I}}$$

and $\Pr[\text{find } \overline{S}^{(\ell)} : U^{\mathcal{F}\setminus\overline{S}^{(\ell)}}, |\psi\rangle] = \mathbb{E}[\||\phi_1\rangle_{\mathbf{R}}\|^2]$. Note that $|\phi_0\rangle$ and $|\phi_1\rangle$ are orthogonal by the fact that $|\phi_0\rangle$ involves no query to $\overline{S}^{(\ell)}$ but $|\phi_1\rangle$ does. Therefore,

$$\mathcal{F}U|\psi\rangle = |\phi_0\rangle + |\phi_1\rangle.$$

5.2 One-way-to-hiding (O2H) Lemma for the Shuffling Oracle

Here, we extend the **One-way-to-hiding lemma (O2H lemma)** by Ambainis et al. [6] to the setting of the shuffling oracle. Briefly, the O2H lemma shows that for any two functions g and h, the probability for a quantum algorithm to distinguish them is bounded by the probability that the quantum algorithm ever "finds" an element in the input domain on which g and h disagree with each other times the depth of the quantum algorithm.

Lemma 5.7 (O2H lemma for the shuffling oracle). *Let $k, d \in \mathbb{N}$ satisfying that $k < d$. Let U be any single depth quantum circuit and ρ be any input state. Let f be any function from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$.*

Let $\mathcal{F}$ be a random (d, f)-shuffling of f. Let $\mathbf{S} := (\overline{S}^{(0)}, \ldots, \overline{S}^{(d)})$ be a sequence of random hidden sets as defined in Definition 5.2. Let $\mathcal{G}$ be the shadow of $\mathcal{F}$ in $\overline{S}^{(k)}$. Then, for any binary string t,

$$\begin{aligned} |\Pr[\Pi_{0/1} \circ \mathcal{F}U(\rho) = t] - \Pr[\Pi_{0/1} \circ \mathcal{G}U(\rho) = t]| &\leq B(\mathcal{F}U(\rho), \mathcal{G}U(\rho)) \\ &\leq \sqrt{2\Pr[\mathit{find}\ \overline{S}^{(k)} : U^{\mathcal{F}\setminus\overline{S}^{(k)}}, \rho]}, \end{aligned}$$

where $\Pi_{0/1}$ is the measurement in the standard basis. Here, the probability is over $\mathcal{F}$, $\mathbf{S}$, and the randomness of the quantum mechanism.

Proof. We will prove the case where the initial state is a pure state and then the general case directly follows from the concavity of the mixed state. For simplicity, we denote $\Pr[\text{find } \overline{S}^{(k)} : U^{\mathcal{F}\setminus\overline{S}^{(k)}}, \rho]$ as P_{find}.

Fix $\mathcal{F}$ and $\overline{S}^{(k)}$. We let $|\psi\rangle$ be any initial state and

$$\mathcal{F}U_{\overline{S}^{(k)}}U|\psi\rangle_{\mathbf{R}}|0\rangle_{\mathbf{I}} := |\phi_0\rangle_{\mathbf{R}}|0\rangle_{\mathbf{I}} + |\phi_1\rangle_{\mathbf{R}}|1\rangle_{\mathbf{I}},$$

where $|\phi_0\rangle$ and $|\phi_1\rangle$ are two unnormalized states. $|\phi_0\rangle$ and $|\phi_1\rangle$ are orthogonal due to the fact that all queries $|\phi_0\rangle$ consists of are not in $\overline{S}^{(k)}$,while the queries $|\phi_1\rangle$ performs are elements in $U_{\overline{S}^{(k)}}$. This, therefore, implies that

$$|\psi_f\rangle := \mathcal{F}U|\psi\rangle = |\phi_0\rangle + |\phi_1\rangle.$$

Similarly, we let

$$\mathcal{G}UU_{\overline{S}^{(k)}}|\psi\rangle_{\mathbf{R}}|0\rangle_{\mathbf{I}} := |\phi_0\rangle_{\mathbf{R}}|0\rangle_{\mathbf{I}} + |\phi_1^{\perp}\rangle_{\mathbf{R}}|1\rangle_{\mathbf{I}},$$

and due to the same fact that $|\phi_0\rangle$ and $|\phi_1^{\perp}\rangle$ are orthogonal, we have

$$|\psi_g\rangle := \mathcal{G}U|\psi\rangle = |\phi_0\rangle + |\phi_1^{\perp}\rangle.$$

Here, $|\phi_1^{\perp}\rangle$ and $|\phi_1\rangle$ are orthogonal, since $\mathcal{G}$ maps all elements in $\overline{S}^{(k)}$ to $\perp$.

Let $P_{find}(\mathcal{F}, \overline{S}^{(k)})$ be the probability that a standard-basis measurement in the register $\mathbf{I}$ of $\mathcal{F}U_{\overline{S}^{(k)}}U|\psi\rangle_{\mathbf{R}}|0\rangle_{\mathbf{I}}$ returns 1, which is equal to $\||\phi_1\rangle\|^2$. Consider the two-norm distance between $|\psi_f\rangle$ and $|\psi_g\rangle$,

$$\begin{aligned} \||\psi_f\rangle - |\psi_g\rangle\|^2 &= \||\phi_1\rangle - |\phi_1^{\perp}\rangle\|^2 \\ &= \||\phi_1\rangle\|^2 + \||\phi_1^{\perp}\rangle\|^2 \\ &\leq 2\||\phi_1\rangle\|^2 = 2P_{find}(\mathcal{F}, \overline{S}^{(k)}). \end{aligned}$$

The second equality follows from the fact that $|\phi_1\rangle$ and $|\phi_1^{\perp}\rangle$ are orthogonal. The inequality is because $\||\phi_1\rangle\|^2 = \||\phi_1^{\perp}\rangle\|^2 = 1 - \||\phi_0\rangle\|^2$.

Then, consider the case that $\overline{S}^{(k)}$ and $\mathcal{F}$ are random. The output states of $\mathcal{F}U$ and $\mathcal{G}U$ becomes

$$\rho_f := \sum_{\mathcal{F}, \overline{S}^{(k)}} \Pr[\mathcal{F} \wedge \overline{S}^{(k)}]|\psi_f\rangle\langle\psi_f|, \text{ and}$$

$$\rho_g := \sum_{\mathcal{F}, \overline{S}^{(k)}} \Pr[\mathcal{F} \wedge \overline{S}^{(k)}]|\psi_g\rangle\langle\psi_g|.$$

Consider the fidelity of these two mixed states,

$$\begin{aligned}
F(\rho_f, \rho_g) &= F\left(\sum_{\mathcal{F},\overline{S}^{(k)}} \Pr[\mathcal{F} \wedge \overline{S}^{(k)}]|\psi_f\rangle\langle\psi_f|, \sum_{\mathcal{F},\overline{S}^{(k)}} \Pr[\mathcal{F} \wedge \overline{S}^{(k)}]|\psi_g\rangle\langle\psi_g|\right) \\
&\geq \sum_{\mathcal{F},\overline{S}^{(k)}} \Pr[\mathcal{F} \wedge \overline{S}^{(k)}] F\left(|\psi_f\rangle\langle\psi_f|, |\psi_g\rangle\langle\psi_g|\right) \\
&\geq 1 - \frac{1}{2} \cdot \sum_{\mathcal{F},\overline{S}^{(k)}} \Pr[\mathcal{F} \wedge \overline{S}^{(k)}] \||\psi_f\rangle - |\psi_g\rangle\|^2 \\
&\geq 1 - \frac{1}{2} \cdot \sum_{\mathcal{F},\overline{S}^{(k)}} \Pr[\mathcal{F} \wedge \overline{S}^{(k)}] 2P_{find}(\mathcal{F}, \overline{S}^{(k)}). \\
&\geq 1 - P_{find}.
\end{aligned}$$

Then, we obtain the Bures distance between the two states,

$$\begin{aligned}
B(\rho_f, \rho_g) &= \sqrt{2 - 2F(\rho_f, \rho_g)} \\
&\leq \sqrt{2 - 2(1 - P_{find})} = \sqrt{2P_{find}}.
\end{aligned}$$

Finally, by Claim 1,

$$|\Pr[\Pi_{0/1} \circ \mathcal{F}U(\rho) = t] - \Pr[\Pi_{0/1} \circ \mathcal{G}U(\rho) = t]| \leq \sqrt{2P_{find}}.$$

□

5.3 Bounding the Finding Probability

We have just shown that the probability of distinguishing $\mathcal{F}$ and its shadow can be bounded by the probability of finding the "shadow." Then, we would like to show how to bound the finding probability.

Following the previous section, we let $\mathcal{F}$ be a random (d, f)-shuffling of f and $\mathbf{S} := (\overline{S}^{(0)}, \ldots, \overline{S}^{(d)})$ be a sequence of random hidden sets as defined in Definition 5.2 (which could be chosen according to an arbitrary distribution). We show that the finding probability of $\overline{S}^{(k)}$ is bounded.

Lemma 5.8. *Suppose* $\Pr[x \in S_i^{(k)} | x \in S_i^{(k-1)}] \leq p$ *for* $i = k, \ldots, d$. *Then for any single-depth quantum circuit* U *and initial state* ρ, *which are promised to be uncorrelated to* $\hat{\mathcal{F}}^{(k-1)}$ *and* $\overline{S}^{(k)}$,[††]

$$\Pr[\mathit{find}\, \overline{S}^{(k)} : U^{\mathcal{F}\setminus\overline{S}^{(k)}}, \rho] \leq q \cdot p,$$

where q *is the number of queries* U *performs.*

Proof. It suffices to prove the case where ρ is a pure state. Let $|\psi\rangle$ be the initial state and be uncorrelated to $\hat{\mathcal{F}}^{(k-1)}$. We represent $\mathcal{F}U_{\overline{S}^{(k)}}U|\psi\rangle$ as

$$\sum_{\mathbf{X}_0,\ldots,\mathbf{X}_d} c(\mathbf{X}_0, \ldots, \mathbf{X}_d) \left(\bigotimes_{i=0}^{d} |i, \mathbf{X}_i, f_i(\mathbf{X}_i)\rangle\right) |w(\mathbf{X}_0, \ldots, \mathbf{X}_d)\rangle |b(\mathbf{X}_1, \ldots, \mathbf{X}_d)\rangle_{\mathbf{I}},$$

[††]Here, ρ and U can be arbitrarily correlated to $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k-1)}$, and $\mathcal{F}$ and $\overline{S}^{(k)}$ can be sampled arbitrarily.

where $b(\mathbf{X}_1,\ldots,\mathbf{X}_d) = 1$ if there exists $i \in [\ell,d]$ such that $\mathbf{X}_i \cap S_i^{(k)} \neq \{\phi\}$; otherwise, $b(\mathbf{X}_1,\ldots,\mathbf{X}_d) = 0$. We can assume all queries are in $\overline{S}^{(k-1)}$ without loss of generality. Since both U and $|\psi\rangle$ are uncorrelated to $\overline{S}^{(k)}$, the probability that $b(\mathbf{X}_1,\ldots,\mathbf{X}_d) = 1$ is at most $p \cdot (\sum_{i=k}^{d} |\mathbf{X}_i|)$ for all $\mathbf{X}_1,\ldots,\mathbf{X}_d$ by union bound. Therefore,

$$\begin{aligned}
&\Pr[\text{find } \overline{S}^{(k)} : U^{\mathcal{F}\setminus\overline{S}^{(k)}}, |\psi\rangle] \\
&= \mathbb{E}\left[\left\| \sum_{\mathbf{X}_0,\ldots,\mathbf{X}_d : b(\mathbf{X}_0,\ldots,\mathbf{X}_d)=1} c(\mathbf{X}_0,\ldots,\mathbf{X}_d)\left(\bigotimes_{i=0}^{d} |i,\mathbf{X}_i,f_i(\mathbf{X}_i)\rangle\right)|w(\mathbf{X}_0,\ldots,\mathbf{X}_d)\rangle \right\|^2\right] \\
&= \sum_{\mathbf{X}_0,\ldots,\mathbf{X}_d : b(\mathbf{X}_0,\ldots,\mathbf{X}_d)=1} |c(\mathbf{X}_0,\ldots,\mathbf{X}_d)|^2 \cdot \Pr[\bigvee_{i=k}^{d} (\mathbf{X}_i \cap S_i^{(k)} \neq \{\phi\})] \\
&\leq q \cdot p
\end{aligned}$$

for q the number of queries U performs. The second equality follows from the fact that for different set of queries, $|i,\mathbf{X}_i,f_i(\mathbf{X}_i)\rangle$'s are orthogonal. The last inequality follows from the union bound. □

6 THE D-SSP IS HARD FOR $\mathrm{QNC_D}$

We start by showing that the d-SSP is intractable for any $\mathrm{QNC_d}$ circuit as a warm-up. We first prove the main theorem in this section.

Theorem 6.1. *Let $n,d \in \mathbb{N}$. Let $(\mathcal{A},\rho)$ be any d-depth quantum circuit and initial state. Let f be a random Simon's function from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$ with hidden shift s. Give $\mathcal{A}$ the access to the shuffling oracle $\mathcal{O}_{\mathrm{unif}}^{f,d}$. Let $\mathcal{F}$ be the (d,f)-shuffling sampled from $\mathcal{O}_{\mathrm{unif}}^{f,d}$, then*

$$\Pr[\mathcal{A}^{\mathcal{F}}(\rho) = s] \leq d \cdot \sqrt{\frac{\mathrm{poly}(n)}{2^n}} + \frac{1}{2^n}.$$

Proof. We choose $\mathbf{S} = (\overline{S}^{(0)},\ldots,\overline{S}^{(d)})$ according to Procedure 1 and represent $\mathcal{F}$ in form $(\mathcal{F}^{(1)},\ldots,\mathcal{F}^{(d+1)})$ regarding to $\mathbf{S}$. Let $\mathcal{G}_\ell$ be the shadow function of $\mathcal{F}$ in $\overline{S}^{(\ell)}$ for $\ell \in [d]$. We define

$$\mathcal{A}^{\mathcal{G}} := U_{d+1}\mathcal{G}_d U_d \cdots \mathcal{G}_1 U_1.$$

Then, for any initial state ρ_0 that is uncorrelated to $\mathcal{F}$,

$$\begin{aligned}
&|\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = s] - \Pr[\mathcal{A}^{\mathcal{G}}(\rho_0) = s]| \\
&= |\Pr[U_{d+1}\mathcal{F}U_d\cdots\mathcal{F}U_1(\rho_0) = s] - \Pr[U_{d+1}\mathcal{G}_dU_d\cdots\mathcal{G}_1U_1(\rho_0) = s]| \\
&\leq |\Pr[U_{d+1}\mathcal{F}U_d\cdots\mathcal{F}U_2\mathcal{F}U_1(\rho_0) = s] - \Pr[U_{d+1}\mathcal{F}U_d\cdots\mathcal{F}U_2\mathcal{G}_1U_1(\rho_0) = s]| \\
&+|\Pr[U_{d+1}\mathcal{F}U_d\cdots\mathcal{F}U_2\mathcal{G}_1U_1(\rho_0) = s] - \Pr[U_{d+1}\mathcal{G}_dU_d\cdots\mathcal{G}_2U_2\mathcal{G}_1U_1(\rho_0) = s]| \\
&\leq \sum_{i=1}^{d} B(\mathcal{F}U_i(\rho_{i-1}),\rho_i) \\
&\leq \sum_{i=1}^{d} \sqrt{2\Pr[\text{find } \overline{S}^{(i)} : U_i^{\mathcal{F}\setminus\overline{S}^{(i)}}, \rho_{i-1}]},
\end{aligned}$$

where $\rho_i := \mathcal{G}_i U_i \rho_{i-1} U_i^\dagger \mathcal{G}_i^\dagger$ for $i \geq 1$.

PROCEDURE 1: The hidden sets for $\mathrm{QNC_d}$

Let $d, n \in \mathbb{N}$ and f a random Simon's problem from $\mathbb{Z}_2^n \to \mathbb{Z}_2^n$. Consider $\mathcal{F} \sim \mathcal{D}(f, d)$, we construct S as follows:

- Let $\vec{S}^{(0)} := (S_0^{(0)}, \ldots, S_d^{(0)})$, where $S_j^{(0)} := \mathbb{Z}_2^{(d+2)n}$ for $j = 0, \ldots, d$.
- For $\ell = 1, \ldots, d$,
  (1) let $S_\ell^{(\ell)}$ be a subset chosen uniformly at random with the promise that $|S_\ell^{(\ell)}|/|S_\ell^{(\ell-1)}| = \frac{1}{2^n}$ and $S_\ell \subset S_\ell^{(\ell)}$;
  (2) for $j = \ell + 1, \ldots, d$, let $S_j^{(\ell)} := \{f_{j-1} \circ \cdots \circ f_\ell(S_\ell^{(\ell)})\}$;
  (3) let $\overline{S}^{(\ell)} := (S_\ell^{(\ell)}, \ldots, S_d^{(\ell)})$.
- We then let $\mathrm{S} := (\overline{S}^{(0)}, \ldots, \overline{S}^{(d)})$.

It is not hard to see that $\Pr[\mathcal{A}^{\mathcal{G}}(\rho_0) = s]$ is at most $\frac{1}{2^n}$. This follows from the fact that $\mathcal{G}_1, \ldots, \mathcal{G}_d$ does not contain information of f_d^* and therefore $\mathcal{A}^{\mathcal{G}}(\rho_0)$ can do no better than guess. The rest to show is that $\Pr[find\ \overline{S}^{(i)} : U_i^{\mathcal{F}\setminus\overline{S}^{(i)}}, \rho_{i-1}]$ is at most $\frac{\mathrm{poly}(n)}{2^n}$ for all $i \in [d]$. To prove it, we show that $\Pr[x \in S_j^{(\ell)} | x \in S_j^{(\ell-1)}] = \frac{1}{2^n}$ for $\ell = 1, \ldots, d$ and $j = \ell, \ldots, d$. We prove it by induction on ℓ.

For the base case $\ell = 1$, for all $j \in [d]$, and $x \in S_j^{(0)}$,

$$\begin{aligned}
\Pr[x \in S_j^{(1)}] &= \Pr[x \in S_j] \Pr\left[x \in S_j^{(1)} \middle| x \in S_j\right] + \Pr[x \notin S_j] \Pr\left[x \in S_j^{(1)} \middle| x \notin S_j\right] \\
&= \Pr[x \in S_j] + (1 - \Pr[x \in S_j]) \Pr\left[x \in S_j^{(1)} \middle| x \notin S_j\right] \\
&= \frac{1}{2^{(d+1)n}} + \left(1 - \frac{1}{2^{(d+1)n}}\right) \frac{2^{(d+1)n} - 2^n}{2^{(d+2)n} - 2^n} = \frac{1}{2^n}.
\end{aligned}$$

The second equality is because $x \in S_j$ implies $x \in S_j^{(1)}$ and the third inequality follows from the fact that $f_0, \ldots, f_{d-1}$ are uniformly random one-to-one functions.

Suppose that the randomness holds for $\ell = k$. Note that ρ_k could be correlated to $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}$, and therefore, $\mathcal{A}$ could have much information about $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}$ given ρ_k. So, we just assume that $\mathcal{A}$ knows $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}$ without loss of generality, as in Remark 4. Note that this potentially makes the algorithm stronger and thus only makes the impossibility result stronger.

Then, given $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}$, for $j \geq k + 1$ and $x \in S_j^{(k)}$

$$\begin{aligned}
\Pr\left[x \in S_j^{(k+1)}\right] &= \Pr[x \in S_j] \Pr\left[x \in S_j^{(k)} \middle| x \in S_j\right] \\
&\quad + \Pr[x \notin S_j] \Pr\left[x \in S_j^{(k)} \middle| x \notin S_j\right] \\
&= \Pr[x \in S_j] + (1 - \Pr[x \in S_j]) \Pr\left[x \in S_j^{(k)} \middle| x \notin S_j\right] \\
&= \frac{2^n}{2^{(d+2-k)n}} + \left(1 - \frac{2^n}{2^{(d+2-k)n}}\right) \frac{2^{(d+1-k)n} - 2^n}{2^{(d+2-k)n} - 2^n} \\
&= \frac{1}{2^n}.
\end{aligned}$$

The second last equality follows from the fact that, given $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}$, $\hat{f}_j^{(k)}$ is still a uniformly random one-to-one function for $j = k, \ldots, d - 1$.

Finally, U_i and ρ_{i-1} are uncorrelated to $\hat{\mathcal{F}}^{(i)}$. By Lemma 5.8, $\Pr[\text{find } \overline{S}^{(i)} : U_i^{\mathcal{F}\setminus\overline{S}^{(i)}}, \rho_{i-1}]$ is at most $q_i \cdot \frac{1}{2^n}$, where q_i is the number of queries U_i performs. Therefore,

$$\begin{aligned}
&|\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = s] - \Pr[\mathcal{A}^{\mathcal{G}}(\rho_0) = s]| \\
&\leq \sum_{i=1}^{d} \sqrt{2\Pr[\text{find } \overline{S}^{(i)} : U_i^{\mathcal{F}\setminus\overline{S}^{(i)}}, \rho_{i-1}]} \\
&\leq \sum_{i=1}^{d} \sqrt{\frac{2q_i}{2^n}} = d \cdot \sqrt{\frac{\text{poly}(n)}{2^n}}.
\end{aligned}$$

□

Theorem 6.1 shows that the search d-SSP is hard for any QNC_d circuit. By following the same proof, we can show that for any QNC_d circuit, the d-SSP is also hard.

Theorem 6.2. *The d-SSP cannot be decided by any QNC_d circuit with probability greater than $\frac{1}{2} + d \cdot \sqrt{\frac{\text{poly}(n)}{2^n}}$.*

Proof. We also consider the same shadow $\mathcal{G}$ in the proof of Theorem 6.1. Following that proof, for any ρ and $\mathcal{A}$,

$$|\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = 0] - \Pr[\mathcal{A}^{\mathcal{G}}(\rho_0) = 0]| \leq d \cdot \sqrt{\frac{\text{poly}(n)}{2^n}}.$$

Then, the rest to check is that $\mathcal{A}^{\mathcal{G}}$ cannot solve the d-SSP with high probability. Similar to the case of the search d-SSP, since $\mathcal{G}_1, \ldots, \mathcal{G}_d$ have the core function f_d^* be blocked, $\mathcal{A}^{\mathcal{G}}$ has no information about f and thus cannot do better than guess. This implies that $\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = 0] \leq \frac{1}{2} + d \cdot \sqrt{\frac{\text{poly}(n)}{2^n}} < 2/3$. □

Therefore, the language defined in Definition 4.10 is also hard for QNC circuit.

Corollary 6.3. *Let O and $\mathcal{L}(O)$ be defined as in Definition 4.10. $\mathcal{L}(O) \notin \text{BQNC}^O$.*

7 THE D-SSP IS HARD FOR D-QC SCHEMES

The main theorem we are going to show in this section is that the search d-SSP is hard for all d-QC schemes.

Theorem 7.1. *Let $d, n \in \mathbb{N}$. For any d-QC scheme $\mathcal{A}$ and initial state ρ, let f be a random Simon's function from $\mathbb{Z}_2^N$ to $\mathbb{Z}_2^n$ with hidden shift s, and $\mathcal{F} \sim \mathcal{D}(f, d)$, then*

$$\Pr[\mathcal{A}^{\mathcal{F}}(\rho) = s] \leq d \cdot \sqrt{\frac{\text{poly}(n)}{2^n}}.$$

Before proving Theorem 7.1, we first recall the classical lower bound for the Simon's problem.

Lemma 7.2. *Let $\mathcal{A}_c$ be any PPT algorithm. Let $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ be a uniformly random Simon function. Let $q \in \mathbb{N}$ be the number of queries $\mathcal{A}_c$ performs, and $S \subset \mathbb{Z}_n$ be the set where $f(x)$ is known for $x \in S$ and $f(x) \neq f(x')$ for $x \neq x'$. Then the probability that $\mathcal{A}_c^f(S, f(S))$ outputs the hidden shift correctly is at most*

$$\frac{(q + 1 + |S|)^2}{2^{n+1} - (q + 1 + |S|)^2},$$

which is $O(\text{poly}(n)/2^n)$ when q and $|S|$ are polynomial in n.

Proof. We only need to consider the case where $\mathcal{A}_c$ is a deterministic algorithm. A probabilistic algorithm can be seen as a convex combination of deterministic algorithm; therefore, the success probability of a probabilistic algorithm must be an average over deterministic algorithms.

Let $S \subset \mathbb{Z}_2^n$ and $f(x) \neq f(y)$ for $x, y \in S$ and $x \neq y$. The probability that $\mathcal{A}_c$ finds a collision is

$$\Pr\left[\text{collision} : \mathcal{A}_c^f(S)\right] \leq \sum_{i=1}^{q} \frac{i + |S| - 1}{2^n - (i + |S|)^2/2} \leq \frac{(q + |S|)^2}{2^{n+1} - (q + |S|)^2}.$$

For any algorithm that can find s with probability p by performing q queries, it can find a collision with the same probability by performing $q + 1$ queries. The probability of finding a collision by using $q + 1$ queries is at most $\frac{(q+|S|+1)^2}{2^{n+1}-(q+|S|+1)^2}$. Therefore,

$$\Pr\left[\mathcal{A}_c^f(S) = s\right] \leq \frac{(q + |S| + 1)^2}{2^{n+1} - (q + |S| + 1)^2}.$$

□

7.1 Proof of Theorem 7.1

Recall that we represent a d-QC scheme $\mathcal{A}$ with access to $\mathcal{F} \sim \mathcal{D}(f, d)$ as

$$\mathcal{A}_c^{\mathcal{F}} \circ \left(\Pi_{0/1} \circ \mathcal{F}U_d \circ \mathcal{A}_c^{\mathcal{F}}\right) \circ \cdots \circ \left(\Pi_{0/1} \circ \mathcal{F}U_1 \circ \mathcal{A}_c^{\mathcal{F}}\right).$$

We denote $\Pi_{0/1} \circ \mathcal{F}U_i \circ \mathcal{A}_c^{\mathcal{F}}$ as $L_i^{\mathcal{F}}$ for $i = 1, \ldots, d$ and rewrite the representation above as $\mathcal{A}_c^{\mathcal{F}} \circ L_d^{\mathcal{F}} \circ \cdots \circ L_1^{\mathcal{F}}$. We let q_i be the number of quantum queries and r_i be the number of classical queries the algorithm performs in L_i. We let $q := \sum_{i=1}^d q_i$ and $r := \sum_{i=1}^d r_i$.

For the ease of the analysis, we allow $\mathcal{A}_c$ to learn the whole path from f_0 to f_d by just one query, which we called the "**path query**." It is worth noting that $\mathcal{A}_c$ that can make path queries can be simulated by the original model. This follows from the fact that the original model can achieve the same thing by using d times as many queries as $\mathcal{A}_c$.

To prove the theorem, we define an $\mathbf{S}$ which has property that $\Pr[x \in S_j^{(\ell)}] \leq p$ as described in Lemma 5.8 for some p that is small enough. In the following, we show that $\mathbf{S} = (\overline{S}^{(1)}, \ldots, \overline{S}^{(d)})$ in Procedure 2 satisfies this property with $p = \frac{1}{2^n}$:

Claim 2. *Let $d, n \in \mathbb{N}$. Let $\ell \in [d]$. Let $\mathcal{A}_c$ be any randomized algorithm. Let $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ be any function. Let $\mathcal{F} \sim \mathcal{D}(f, d)$. Let $\overline{S}^{(1)}, \ldots, \overline{S}^{(\ell)}$ and $\overline{T}^{(1)}, \ldots, \overline{T}^{(\ell)}$ be defined as in Procedure 2 regarding to $\mathcal{A}_c$. Given $(\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(\ell-1)})$ and $(\overline{T}^{(1)}, \ldots, \overline{T}^{(\ell)})$, then*

$$\Pr_{\mathcal{F},\mathbf{S}}\left[x \in S_j^{(\ell)} \middle| x \in S_j^{(\ell-1)} \setminus T_j^{(\ell)}\right] = \frac{1}{2^n} \text{ for } j = \ell, \ldots, d.$$

Proof. We prove it via induction on the depth of $\mathcal{F}$. For the base case where $\ell = 1$, given $\overline{T}^{(1)}$ and f_j on $T_j^{(1)}$ for $j = 1, \ldots, d$, for all $i \in [d]$ and $x_i \in S_i^{(0)} \setminus T_i^{(1)}$,

$$\begin{aligned}
\Pr\left[x_i \in S_i^{(1)}\right] &= \Pr\left[x_i \in S_i \setminus T_i^{(1)}\right] + \Pr\left[x_i \in S_i^{(1)} \setminus S_i\right] \\
&= \frac{2^n - |T_i^{(1)}|}{2^{(d+2)n} - |T_i^{(1)}|} + \frac{|S_i^{(1)}| - 2^n + |T_i^{(1)}|}{2^{(d+2)n} - |T_i^{(1)}|} \\
&= \frac{|S_i^{(1)}|}{2^{(d+2)n} - |T_i^{(1)}|} = \frac{1}{2^n}.
\end{aligned}$$

PROCEDURE 2: The hidden sets for d-QC scheme

Let $d, n \in \mathbb{N}$ and f a random Simon's problem from $\mathbb{Z}_2^n \to \mathbb{Z}_2^n$. Given $\mathcal{F} \sim \mathcal{D}(f, d)$ and $\mathcal{A}$ a d-QC scheme, We construct S as follows:

- Let $\vec{S}^{(0)} := (S_0^{(0)}, \ldots, S_d^{(0)})$, where $S_j^{(0)} := \mathbb{Z}_2^{(d+2)n}$ for $j = 0, \ldots, d$.
- For $\ell = 1, \ldots, d$:
  (1) After the ℓth $\mathcal{A}_c^{\mathcal{F}}$ is applied, let $\overline{T}^{(\ell)} = (T_\ell^{(\ell)}, \ldots, T_d^{(\ell)})$ be the set of points the ℓth $\mathcal{A}_c^{\mathcal{F}}$ queried. As we have mentioned before, we allow $\mathcal{A}_c$ to query the whole path by one query. Hence, $f_j(T_j^{(\ell)}) = T_{j+1}^{(\ell)}$ for $j = \ell, \ldots, d$.
  (2) Let $W_\ell^{(\ell-1)} := S_\ell^{(\ell-1)} \setminus T_\ell^{(\ell)}$. Then, we choose $S_\ell^{(\ell)}$ uniformly randomly from $W_\ell^{(\ell-1)}$ with the promise that $|S_\ell^{(\ell)}|/|W_\ell^{(\ell-1)}| = 1/2^n$ and $S_\ell \setminus (T_\ell^{(1)} \cup \cdots \cup T_\ell^{(\ell)}) \subset S_\ell^{(\ell)}$.
  (3) For $j = \ell + 1, \ldots, d$, let $S_j^{(\ell)} := \{f_{j-1} \circ \cdots \circ f_\ell(S_\ell^{(\ell)})\}$.
  (4) Let $\overline{S}^{(\ell)} = (S_\ell^{(\ell)}, \ldots, S_d^{(\ell)})$.
- $\mathrm{S} := (\overline{S}^{(0)}, \ldots, \overline{S}^{(d)})$.

We now suppose that when $\ell = k - 1$, given $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k-2)}$ and $\overline{T}^{(1)}, \ldots, \overline{T}^{(k-1)}$, for $i = k-1, \ldots, d$,

$$\Pr\left[x \in S_i^{(k-1)} \middle| x \in S_i^{(k-2)} \setminus T_i^{k-1}\right] = \frac{1}{2^n}.$$

Then, for $\ell = k$, given $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k-1)}$ and $\overline{T}^{(1)}, \ldots, \overline{T}^{(k)}$, for $i = k, \ldots, d$ and $x \in S_i^{(k-1)} \setminus T_i^{(k)}$,

$$\begin{aligned}
\Pr\left[x \in S_i^{(k)}\right] &= \Pr\left[x \in S_i \setminus \left(\cup_{m=1}^k \left(T_i^{(m)}\right)\right)\right] + \Pr\left[x \in S_i^{(k)} \setminus S_i\right] \\
&= \frac{2^n - \sum_{m=1}^k |T_i^{(m)}|}{|S_i^{(k-1)}| - |T_i^{(k)}|} + \frac{|S_i^{(k)}| - 2^n + \sum_{m=1}^k |T_i^{(m)}|}{|S_i^{(k-1)}| - |T_i^{(k)}|} \\
&= \frac{1}{2^n}.
\end{aligned}$$

□

Now, we are ready to prove Theorem 7.1.

Proof of Theorem 7.1. We choose the hidden set $\mathrm{S} = (\overline{S}^{(0)}, \ldots, \overline{S}^{(d)})$ according to Procedure 2. In the procedure, we choose $\overline{S}^{(\ell)}$ after the $\mathcal{A}_c$ in L_ℓ has performed. We represent $\mathcal{F}$ as $(\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(d+1)})$ according to S. Let $\mathcal{G}_\ell$ be the shadow of $\mathcal{F}$ in $\overline{S}^{(\ell)}$ for $\ell \in [d]$. We define

$$\begin{aligned}
\mathcal{A}^{\mathcal{G}} &:= \mathcal{A}_c^{\mathcal{F}} \circ \left(\Pi_{0/1} \circ \mathcal{G}_d U_d \circ \mathcal{A}_c^{\mathcal{F}}\right) \circ \cdots \circ \left(\Pi_{0/1} \circ \mathcal{G}_1 U_1 \circ \mathcal{A}_c^{\mathcal{F}}\right) \\
&:= \mathcal{A}_c^{\mathcal{F}} \circ L_d^{\mathcal{G}_d} \circ \cdots \circ L_1^{\mathcal{G}_1}.
\end{aligned}$$

$\mathcal{A}^{\mathcal{G}}$ succeeds to output the hidden shift with probability at most $\frac{(r+1)^2}{2^n - (r+1)^2}$, where r is the number of queries the classical algorithms perform. Note that the outputs of $U_1, \ldots, U_d$ are uncorrelated to $\hat{\mathcal{F}}^{(d)}$. This fact implies that, given the measurement outcomes of the ith layer quantum unitaries, $\mathcal{A}_c$ can only learn information about $\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(d)}$. This does not give $\mathcal{A}_c$ more information

about f. Therefore, $\mathcal{A}_c$ at L_i succeeds with probability at most $\frac{(\sum_{j=1}^{i+1} r_j+1)^2}{2^n-(\sum_{j=1}^{i+1} r_j+1)^2}$ and $\mathcal{A}^{\mathcal{G}}$ succeeds with probability at most $\frac{\mathrm{poly}(n)}{2^n}$.

Let ρ_0 be the initial state and ρ_i be the output state of $(L_i^{\mathcal{G}_i} \circ \cdots \circ L_1^{\mathcal{G}_1})(\rho_0)$ for $i = 1, \ldots, d$, we can show that

$$
\begin{aligned}
&|\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = s] - \Pr[\mathcal{A}^{\mathcal{G}}(\rho_0) = s]| \\
&\leq \left|\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = s] - \Pr\left[(L_d \circ \cdots \circ L_2)^{\mathcal{F}}\left(L_1^{\mathcal{G}}(\rho_0)\right) = s\right]\right| \\
&+ \quad \left|\Pr\left[(L_d \circ \cdots \circ L_2)^{\mathcal{F}}\left(L_1^{\mathcal{G}}(\rho_0)\right) = s\right] - \Pr\left[(L_d \circ \cdots \circ L_2)^{\mathcal{G}}\left(L_1^{\mathcal{G}}(\rho_0)\right) = s\right]\right| \\
&\leq B\left(L_1^{\mathcal{F}}(\rho_0), L_1^{\mathcal{G}}(\rho_0)\right) \\
&+ \quad |\Pr[(L_d \circ \cdots \circ L_2)^{\mathcal{F}}(\rho_1) = s] - \Pr[(L_d \circ \cdots \circ L_2)^{\mathcal{G}}(\rho_1) = s]| \\
&\leq \sum_{i=1}^{d} B\left(\rho_i, L_i^{\mathcal{F}}(\rho_{i-1})\right) \qquad (8) \\
&\leq \sum_{i=1}^{d} \sqrt{2\Pr[\text{find } \overline{S}^{(i)} : U_i^{\mathcal{F}\setminus\overline{S}^{(i)}}, \rho_{i-1}]}. \qquad (9)
\end{aligned}
$$

Equation (8) is by the hybrid argument and Equation (9) is from Lemma 5.7.

Then, by Lemma 5.8 and Claim 2,

$$\Pr\left[\text{find } \overline{S}^{(i)} : U_i^{\mathcal{F}\setminus\overline{S}^{(i)}}, \rho_{i-1}\right] \leq \frac{q_i}{2^n}.$$

This implies that

$$\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = s] \leq \frac{\mathrm{poly}(n)}{2^n} + d \cdot \sqrt{\frac{\mathrm{poly}(n)}{2^n}} \leq \sqrt{\frac{\mathrm{poly}(n)}{2^n}}.$$

□

7.2 On Separating the Depth Hierarchy of d-QC Scheme

We show that the d-SSP is also hard for any d-QC scheme.

Theorem 7.3. *The d-SSP cannot be decided by any d-QC scheme with probability greater than $\frac{1}{2} + \sqrt{\frac{\mathrm{poly}(n)}{2^n}}$.*

Proof. We consider the same shadow $\mathcal{G}$ in the proof of Theorem 7.1. Following that proof, for any ρ_0 and $\mathcal{A}$,

$$|\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = 1] - \Pr[\mathcal{A}^{\mathcal{G}}(\rho_0) = 1]| \leq d \cdot \sqrt{\frac{\mathrm{poly}(n)}{2^n}}.$$

Then, the rest to check is that $\mathcal{A}^{\mathcal{G}}$ cannot solve the d-SSP with high probability. In case that f is a random Simon's function, $\mathcal{A}^{\mathcal{G}}$ finds s with probability at most $\frac{\mathrm{poly}(n)}{2^n}$. Therefore, $\Pr[\mathcal{A}^{\mathcal{G}}(\rho_0) = 1]$ is at most $1/2 + \mathrm{poly}(n)/2^n$. This implies that $\Pr[\mathcal{A}^{\mathcal{F}}(\rho_0) = 1] \leq \frac{1}{2} + d \cdot \sqrt{\frac{\mathrm{poly}(n)}{2^n}} + \frac{\mathrm{poly}(n)}{2^n}$.

□

Corollary 7.4. *For any $d \in \mathbb{N}$, there is a $(2d+1)$-QC scheme that can solve the d-SSP with high probability, but there is no d-QC scheme that can solve the d-SSP.*

Proof. This corollary follows from Theorems 7.3 and 4.11 directly. □

Finally, we can conclude that

Corollary 7.5. *Let O and $\mathcal{L}(O)$ be defined as in Definition 4.10. $\mathcal{L}(O) \in \mathrm{BQP}^O$ and $\mathcal{L}(O) \notin (\mathrm{BQNC}^{\mathrm{BPP}})^O$.*

Proof. Note that for each $n \in \mathbb{N}$, $O_{unif}^{f_n, d(n)} \in O$ has depth equal to the input size. A quantum circuit with depth $\mathrm{poly}(n)$ can decide if 1^n is in $\mathcal{L}(O)$ by solving the d-SSP by Theorem 4.11. However, for d-QC scheme that only has quantum depth $d = \mathrm{poly}\log n$, it cannot decide the language by Theorem 7.3. □

8 THE D-SSP IS HARD FOR D-CQ SCHEMES

Theorem 8.1. *Let $d, n \in \mathbb{N}$. Let $\mathcal{A}$ be any d-CQ scheme. Let f be a random Simon's function from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$ with hidden shift s. Let $\mathcal{F} \sim \mathcal{D}(f, d)$. Then*

$$\Pr[\mathcal{A}^{\mathcal{F}}() = s] \leq d \cdot \sqrt{\frac{\mathrm{poly}(n)}{2^n}}.$$

Recall that we represent a d-CQ scheme $\mathcal{A}$ as

$$\begin{aligned}&\mathcal{A}_{c,m+1}^{\mathcal{F}} \circ \left(\Pi_{0/1} \circ U_{d+1} \mathcal{F} \cdots \mathcal{F} U_1 \circ \mathcal{A}_{c,m}^{\mathcal{F}}\right) \circ \cdots \circ \left(\Pi_{0/1} \circ U_{d+1} \mathcal{F} \cdots \mathcal{F} U_1 \circ \mathcal{A}_{c,1}^{\mathcal{F}}\right)\\ &:= \mathcal{A}_{c,m+1}^{\mathcal{F}} \circ L_m^{\mathcal{F}} \circ \cdots \circ L_1^{\mathcal{F}}.\end{aligned}$$

The main difficulty for proving Theorem 8.1 is that $L_i^{\mathcal{F}}$ can send some short classical advice to the proceeding processes. This classical advice can have information about mappings in $\mathcal{F}$. Therefore, conditioned on the short advice, the distribution of the shuffling oracle may not be uniform enough for us to follow the same proofs in the previous sections. To deal with this difficulty, we show that conditioned on short classical advice, by fixing the shuffling function on a few paths, the rest of the shuffling oracle is still *almost-uniform*.

8.1 The Presampling Argument for the Shuffling Oracle

Here, we are going to show that for $\mathcal{F} \sim \mathcal{D}(f, d)$ and $\bar{a}$ a short classical string correlated to $\mathcal{F}$, we can approximate $\mathcal{F}|\bar{a}$ ($\mathcal{F}$ given $\bar{a}$) by a **convex combination** of $(p, 1+\delta)$-uniform shuffling functions. In the following, we first define $(p, 1+\delta)$-uniform shuffling functions:

Let X and Y be two sets of elements such that $|X| = |Y|$. Recall that $P(X, Y)$ is the set of all one-to-one functions from X to Y.

Definition 8.2 (Random Variable $\mathcal{H}_{\vec{X}}$). Let $k, N \in \mathbb{N}$ and $\vec{X} := (X_1, \ldots, X_{k+1})$ be a set of sets with size N. Let h_i be random permutations distributed in $P(X_i, X_{i+1})$ for $i = 1, \ldots, k$. Then, we define $\mathcal{H}_{\vec{X}} := (h_1, \ldots, h_k)$ as a random variable consisting of all possible h_i's.

$\mathcal{H}_{\vec{X}}$ is a sequence of random one-to-one functions for which distribution could be arbitrary. In the following, we introduce a distribution that will be used shortly:

Definition 8.3 (Almost-uniform Shuffling). Let $k, N, p' \in \mathbb{N}$ and $0 < \delta < 1$. Let $\vec{X} = (X_1, \ldots, X_{k+1})$ be a set of sets with size N. Consider $\mathcal{H}_{\vec{X}} = (h_1, \ldots, h_k)$.

- $\mathcal{H}_{\vec{X}}$ is ***(1 + δ)-uniform*** if for all subset of paths $P = (\overrightarrow{P}_1, \ldots, \overrightarrow{P}_m)$

$$\Pr[P \text{ is in } \mathcal{H}_{\vec{X}}] \leq (1+\delta)^m \left(\frac{(N-m)!}{N!}\right)^k,$$

  where $\overrightarrow{P}_i = (x_{i,1}, \ldots, x_{i,k+1})$ for $i = 1, \ldots, m$ and $x_{i,j} \in X_j$ for $j = 1, \ldots, k$. We call P is in $\mathcal{H}_{\vec{X}}$ if $h_j(x_{i,j}) = x_{i,j+1}$ for all $i = 1, \ldots, m$ and $j = 1, \ldots, k$.

- $\mathcal{H}_{\vec{X}}$ is $(p', 1+\delta)$-***uniform*** if there exists a set of paths P' with size at most p' such that $\mathcal{H}_{\vec{X}}|P'$ is $(1+\delta)$-uniform, i.e., let P' be fixed, then for all subset of unfixed paths $P = (\overrightarrow{P}_1, \ldots, \overrightarrow{P}_m)$,

$$\Pr[P \text{ is in } \mathcal{H}_{\vec{X}}|P' \text{ is in } \mathcal{H}_{\vec{X}}] \leq (1+\delta)^m \left(\frac{(N-m-p')!}{(N-p')!}\right)^k.$$

A convex combination of $(p', 1+\delta)$-uniform shuffling functions can be defined by the following formula:

$$\mathbf{H} := \sum_{t=1}^{T} p_t \mathcal{H}_t,$$

where $\mathcal{H}_1, \ldots, \mathcal{H}_T$ are $(p', 1+\delta)$-uniform shuffling functions and $p_1, \ldots, p_T$ are the probabilities for how much each $\mathcal{H}_t$ contributes to $\mathbf{H}$. An algorithm $\mathcal{A}$ that has access to a convex combination of shuffling functions, e.g., $\mathbf{H} := \sum_{t=1}^{T} p_t \mathcal{H}_t$, can be represented as

$$\mathcal{A}^{\mathbf{H}} = \sum_{t=1}^{T} p_t \mathcal{A}^{\mathcal{H}_t}.$$

In the following, we show that if the shuffling is $(p', 1+\delta)$-uniform, then the probability to find the hidden sets in the shuffling is still bounded, as we need for Lemma 5.8. In our context, we will only consider a convex combination of finitely many objects, i.e., T is finite.

We will show in Claim 3 that $\mathcal{F}$ along with short classical advice is close to a convex combination of finitely many $(p', 1+\delta)$-uniform shufflings. Briefly, short classical advice can only "fix limited number of paths" in the shuffling oracle and the rest of the oracle can still be close to uniform. Here, p' is the number of paths fixed by the short classical advice and δ represents the uniformity of the rest of the oracle. Furthermore, we can also show that this is still true even if the original oracle is only almost-uniform.

Claim 3. *Let $0 < \gamma, \delta < 1$. Let $p, k, N \in \mathbb{N}$. Let $\vec{X} = (X_1, \ldots, X_{k+1})$ be a set of sets with size N. Let $\mathcal{H}_{\vec{X}} = (h_1, \ldots, h_k)$ be a $(p', 1+\delta')$-uniform shuffling. Let β be a function that takes the shuffling oracle as inputs and outputs a s-bit string y and a set of paths P on $h_1, \ldots, h_k$. Let $\mathcal{H}^{\beta}_{\vec{X}}$ be $\mathcal{H}_{\vec{X}}$ conditioned on β.[‡‡] Then, there exists a convex combination $\mathbf{H}^{\beta}_{\vec{X}}$ of $(p'+p, 1+\delta'+\delta)$-uniform shufflings such that*

$$\mathcal{H}^{\beta}_{\vec{X}} = \mathbf{H}^{\beta}_{\vec{X}} + \gamma' \mathcal{H}',$$

where $p \leq \frac{s-2\log\gamma}{\log(1+\delta+\delta')-\log(1+\delta)} + |P|$, $\gamma' \leq \gamma$, and $\mathcal{H}'$ is an arbitrary random shuffling.

Note that, in our context, $\beta(\cdot)$ can be viewed as an arbitrary d-CQ scheme generating a p-bit advice string and P with polynomial size. Hence, we can assume $\beta(\cdot)$ outputs P with a fixed size at most some polynomial without loss of generality.

Proof. It is obvious that P does not affect the uniformity of the rest permutations in $\mathcal{H}_{\vec{X}}$. So, we only need to take care of the s-bit advice string y. We consider the following two cases: $\Pr[y \in \beta(\mathcal{H}_{\vec{X}})] > \gamma \cdot 2^{-s}$ and $\Pr[y \in \beta(\mathcal{H}_{\vec{X}})] \leq \gamma \cdot 2^{-s}$. The probability is over $\mathcal{H}_{\vec{X}}$ and the randomness of

[‡‡]Here, $\mathcal{H}_{\vec{X}}$ conditioned on β means that $\mathcal{H}_{\vec{X}}$ conditioned on the event that β gives a specific output given the randomness of the shuffling oracle.

β. Along this line, we can write

$$\begin{aligned}\mathcal{H}_{\vec{X}}^{\beta} &= \sum_{y\in\{0,1\}^s,\, P} \Pr[\beta(\mathcal{H}_{\vec{X}}) = (P,y)] \cdot \mathcal{H}_{\vec{X}}^{P,y} \\ &= \sum_{y\in\{0,1\}^s} \Pr[y \in \beta(\mathcal{H}_{\vec{X}})] \cdot \left(\sum_{P} \Pr[P \in \beta(\mathcal{H}_{\vec{X}} \mid y \in \beta(\mathcal{H}_{\vec{X}}))]\right) \cdot \mathcal{H}_{\vec{X}}^{P,y}, \end{aligned} \tag{10}$$

where $\mathcal{H}_{\vec{X}}^{P,y}$ is $\mathcal{H}_{\vec{X}}$ conditioned on $\beta(\mathcal{H}_{\vec{X}}) = (P,y)$.

For the case where $\Pr[y \in \beta(\mathcal{H}_{\vec{X}})] \le \gamma \cdot 2^{-s}$, we can just assume that $\mathcal{H}_{\vec{X}}^{P,y}$ is not $(p'+p, 1+\delta+\delta')$-uniform, since this event happens with probability at most $\gamma \cdot 2^{-s}$, which contributes at most γ to Equation (10). In particular, we can write

$$\begin{aligned}\mathcal{H}_{\vec{X}}^{\beta} = &\sum_{y:\, \Pr[y\in\beta(\mathcal{H}_{\vec{X}})]>\gamma\cdot 2^{-s}} \Pr[y \in \beta(\mathcal{H}_{\vec{X}})] \cdot \left(\sum_{P} \Pr[P \in \beta(\mathcal{H}_{\vec{X}} \mid y \in \beta(\mathcal{H}_{\vec{X}}))]\right) \cdot \mathcal{H}_{\vec{X}}^{P,y} \\ &+ \sum_{y:\, \Pr[y\in\beta(\mathcal{H}_{\vec{X}})]\le\gamma\cdot 2^{-s}} \Pr[y \in \beta(\mathcal{H}_{\vec{X}})] \cdot \left(\sum_{P} \Pr[P \in \beta(\mathcal{H}_{\vec{X}} \mid y \in \beta(\mathcal{H}_{\vec{X}}))]\right) \cdot \mathcal{H}_{\vec{X}}^{P,y} \\ = &\sum_{y:\, \Pr[y\in\beta(\mathcal{H}_{\vec{X}})]>\gamma\cdot 2^{-s}} \Pr[y \in \beta(\mathcal{H}_{\vec{X}})] \cdot \left(\sum_{P} \Pr[P \in \beta(\mathcal{H}_{\vec{X}} \mid y \in \beta(\mathcal{H}_{\vec{X}}))]\right) \cdot \mathcal{H}_{\vec{X}}^{P,y} + \gamma''\mathcal{H}_{\vec{X}}'', \end{aligned} \tag{11}$$

where $\gamma'' \le \gamma$.

We then analyze the case where $\Pr[y \in \beta(\mathcal{H}_{\vec{X}})] > \gamma \cdot 2^{-s}$. Consider all P's where $\Pr[\beta(\mathcal{H}_{\vec{X}}) = (P,y)] > 0$. Let P_1 be the maximal set of paths satisfying that

$$\Pr[(P_1 \text{ in } \mathcal{H}_{\vec{X}}) \mid (P,y)] \ge (1+\delta'+\delta)^{|P_1|} \left(\frac{(N-|P_1|-|P|)!}{(N-|P|)!}\right)^d. \tag{12}$$

Conditioned on $(P_1,\ P,\ y)$, we show that $\mathcal{H}_{\vec{X}}$ is $(1+\delta+\delta')$-uniform by contradiction. Suppose that there exists another set of paths P' such that

$$\Pr[(P' \text{ in } \mathcal{H}_{\vec{X}}) \mid (P_1,P,y)] \ge (1+\delta+\delta')^{|P'|} \left(\frac{(N-|P_1|-|P'|-|P|)!}{(N-P_1-|P|)!}\right)^d.$$

Then,

$$\Pr[(P_1 \cup P' \text{ in } \mathcal{H}_{\vec{X}}) \mid (P,y)] \ge (1+\delta+\delta')^{|P'|+|P_1|} \left(\frac{(N-|P_1|-|P'|-|P|)!}{(N-|P|)!}\right)^d,$$

which contradicts the maximallity of P_1. This proves that $\mathcal{H}_{\vec{X}}^{P,P_1,y}$ is $(1+\delta+\delta')$-uniform.

The size of P_1 is bounded as follows: Since $\Pr[y \in \beta(\mathcal{H}_{\vec{X}})] > \gamma \cdot 2^{-s}$,

$$\begin{aligned}\Pr[(P_1 \text{ in } \mathcal{H}_{\vec{X}}) \mid (P,y)] &\le \frac{\Pr[((P_1 \text{ in } \mathcal{H}_{\vec{X}}) \mid P) \wedge (y \in \beta(\mathcal{H}_{\vec{X}}))]}{\Pr[y \in \beta(\mathcal{H}_{\vec{X}})]} \\ &\le \Pr[(P_1 \text{ in } \mathcal{H}_{\vec{X}}) | P] \cdot 2^s \cdot \gamma^{-1} \\ &\le (1+\delta')^{|P_1|} \left(\frac{(N-|P|-|P_1|)!}{(N-|P|)!}\right)^d \cdot 2^s \cdot \gamma^{-1}. \end{aligned} \tag{13}$$

Combining Equations (12) and (13), we obtain $|P_1| \le \frac{s-\log\gamma}{\log(1+\delta+\delta')-\log(1+\delta)}$.

Now, we can write $\mathcal{H}_{\vec{X}}^{P,y}$ as

$$\Pr[P_1 \text{ in } \mathcal{H}_{\vec{X}}|(P,y)] \cdot \mathcal{H}_{\vec{X}}^{P,P_1,y} + \Pr[\neg(P_1 \text{ in } \mathcal{H}_{\vec{X}})|(P,y)] \cdot \mathcal{H}_{\vec{X}}^{P,\neg P_1,y}.$$

Note that $\mathcal{H}_{\vec{X}}^{P,\neg P_1,y}$ might not be $(p+p', 1+\delta+\delta')$-uniform. Here, $\neg P_1$ is the event that $\mathcal{H}_{\vec{X}}$ does not contain all paths in P_1. We will use this notation for $\neg P_i$.

As long as $\mathcal{H}_{\vec{X}}^{P,\neg P_1,y}$ is not $(p+p', 1+\delta+\delta')$-uniform and

$$\Pr\left[\mathcal{H}_{\vec{X}}^{P,y} \in \text{support}\left(\mathcal{H}_{\vec{X}}^{P,\neg P_1,y}\right)\right] \geq \gamma,$$

we keep decomposing $\mathcal{H}_{\vec{X}}^{P,\neg P_1,y}$ as follows: We find another maximal set of paths P_2 satisfying that

$$\Pr[P_2 \text{ in } \mathcal{H}_{\vec{X}}|(P,\neg P_1,y)] \geq (1+\delta+\delta')^{|P_2|}\left(\frac{(N-|P_2|-|P|)!-(N-|P|-|P_2|-|P_1|)!}{(N-|P|)!-(N-|P|-|P_1|)!}\right)^d.$$

It is obvious that, conditioned on P_2, $\mathcal{H}_{\vec{X}}^{P,\neg P_1,y}$ is $(1+\delta+\delta')$-uniform following the same argument. Furthermore,

$$\begin{aligned}\Pr[P_2 \text{ in } \mathcal{H}_{\vec{X}}|(P,\neg P_1,y)] &= \Pr\left[P_2 \text{ in } \mathcal{H}_{\vec{X}}^{P,y} \,\middle|\, \mathcal{H}_{\vec{X}}^{P,y} \in \text{support}\left(\mathcal{H}_{\vec{X}}^{P,\neg P_1,y}\right)\right] \\ &\leq 2^s \cdot \gamma^{-2} \cdot (1+\delta')^{|P_2|} \cdot \left(\frac{(N-|P_2|-|P|)!-(N-|P|-|P_2|-|P_1|)!}{(N-|P|)!-(N-|P|-|P_1|)!}\right)^d.\end{aligned}$$

This implies that $|P_2| \leq \frac{s-2\log(\gamma)}{\log(1+\delta+\delta')-\log(1+\delta)}$. We keep decomposing $\mathcal{H}_{\vec{X}}^{P,y}$ until either $\mathcal{H}_{\vec{X}}^{P,\neg P_1,\ldots,\neg P_i,y}$ is $(p+p', 1+\delta+\delta')$-uniform or $\Pr[\mathcal{H}_{\vec{X}}^{P,y} \in \text{support}(\mathcal{H}_{\vec{X}}^{P,\neg P_1,\ldots,\neg P_i,y})] < \gamma$. Then, there must exist an integer ℓ such that $\mathcal{H}_{\vec{X}}^{P,y}$ is γ-close to

$$\mathbf{H}_{\vec{X}}^{P,y} = \sum_{i=1}^{\ell} \Pr[P_i \text{ in } \mathcal{H}_{\vec{X}}|P,\neg P_1,\ldots,\neg P_{i-1},y] \cdot \mathcal{H}_{\vec{X}}^{P,\neg P_1,\ldots,\neg P_{i-1},P_i,y}. \tag{14}$$

Finally, by Equations (11) and (14), we conclude that

$$\mathcal{H}_{\vec{X}}^{\beta} = \sum_{y\in\{0,1\}^s,\, P} \Pr[\beta(\mathcal{H}_{\vec{X}}) = (P,y)] \cdot \mathcal{H}_{\vec{X}}^{P,y} = \gamma'\mathcal{H}' + \sum_{P,\, \Pr[y\in\beta(\mathcal{H}_{\vec{X}})]>\gamma 2^{-s}} \mathbf{H}_{\vec{X}}^{P,y}.$$

□

Claim 4. *Let $p,k,N,N' \in \mathbb{N}$ and $0<\delta<1$. Let $\vec{X} = (X_1,\ldots,X_{k+1})$ be a set of sets with size N. Let $\mathcal{H}_{\vec{X}} = (h_1,\ldots,h_k)$ be $(p,1+\delta)$-uniform as defined in Definition 8.3 and P be the fixed paths. Let $X_i' \subset X_i$ be the set on P for $i \in [k+1]$. We choose an N'-element set Y_1 uniformly randomly from $X_1 \setminus X_1'$, and let $Y_i := h(Y_{i-1})$ for $i = 2,\ldots,k$. Then, for $j \in [k]$, for $x_j \in X_j \setminus X_j'$,*

$$\Pr[x_j \in Y_j] \leq (1+\delta)\cdot\frac{N'}{N-p}.$$

Proof. It is obvious that $\Pr[x_1 \in Y_1] = \frac{N'}{N-p}$, since Y_1 is chosen randomly uniformly from X_1. For $i = 2, \ldots, k$, $\Pr[x_i \in Y_i]$ can be calculated as follows:

$$\begin{aligned}
\Pr[x_i \in Y_i] &= \Pr\left[\bigvee_{Y_1 \subset X_1} (x_i \in (h_{i-1} \circ \cdots \circ h_1(Y_1)) \wedge (Y_1 \text{ is chosen}))\right] \\
&\leq \sum_{Y_1 \subset X_1} \Pr[Y_1 \text{ is chosen}] \cdot \Pr[x_i \in (h_{i-1} \circ \cdots \circ h_1(Y_1)) \mid (Y_1 \text{ is chosen})] \\
&\leq \sum_{Y_1 \subset X_1} \Pr[Y_1 \text{ is chosen}] \cdot \left(\sum_{y \in Y_1} \Pr[h_{i-1} \circ \cdots \circ h_1(y) = x_i \mid (Y_1 \text{ is chosen})]\right) \\
&\leq (1+\delta)\frac{N'}{N-p}.
\end{aligned}$$

The first two inequalities follow from the union bound, and the last inequality follows from the fact that $\mathcal{H}_{\vec{X}}$ is $(p, 1+\delta)$-uniform. □

8.2 Proof of Theorem 8.1

We aim to show that there exists a sequence of shadows that is indistinguishable from $\mathcal{F}$ following the same spirit in previous sections. However, to prove Theorem 8.1, we actually show that there exists a "convex combination" of finitely many shadows, which are indistinguishable from $\mathcal{F}$. In particular, we show that there exists a convex combination $\sum_{t=1}^{T} p_t(\mathcal{G}_t^{(1)}, \ldots, \mathcal{G}_t^{(m)})$ such that

$$\left|\Pr\left[L_m^{\mathcal{F}} \circ \cdots \circ L_1^{\mathcal{F}}() = s\right] - \sum_{t=1}^{T} p_t \cdot \Pr\left[\left(L_m^{\mathcal{G}_t^{(m)}} \circ \cdots \circ L_1^{\mathcal{G}_t^{(1)}}() = s\right]\right| \leq md \cdot \sqrt{\frac{\mathrm{poly}(n)}{2^n}}.$$

We will give the details of the shadows later in Lemma 8.6.

Let f be a random Simon's function from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$. Let $\mathcal{F}$ be the random (d, f)-shuffling of f. We represent a round of quantum-classical computing in the d-CQ scheme as

$$\mathcal{B}^{\mathcal{F}} := \Pi_{0/1} \circ U_{d+1}\mathcal{F} \cdots \mathcal{F}U_1 \circ \mathcal{A}_c, \tag{15}$$

where $U_1, \ldots, U_d$ are single depth quantum circuit and $\mathcal{A}_c$ is a PPT algorithm. As we have mentioned earlier, the output of $\mathcal{B}^{\mathcal{F}}$ can be represented as $(P, \bar{a})$, where P is a set of paths fixed by $\mathcal{A}_c$ and $\bar{a}$ is corresponding to the measurement outcome of the quantum circuit.

For each round of quantum-classical computing in the d-CQ scheme, the algorithm is given some binary strings and paths generated by the previous round. We denote the procedure (i.e., the previous rounds of $\mathcal{B}^{\mathcal{F}}$'s) as $\bar{s}$ and view it as a function that generates advice for the current round. In particular, $\bar{s}$ has input as $\mathcal{F}$ and outputs $P, \bar{a}$.

We say that $\bar{s}$ is *uncorrelated* to f conditioned on P if the quantum circuit producing $\bar{a}$ does not change the output distribution when all mappings in f_d^*, except for mappings in P, are replaced by $\perp$. In the following, we define two kinds of advice, which are ideal and semi-ideal for our analysis:

Definition 8.4 (Ideal advice). Let $\bar{s}(\mathcal{F}) = (P, \bar{a})$. We say that $\bar{s}$ is ideal if

- P does not have a collision in f,
- $|P| = \mathrm{poly}(n)$, and
- $\bar{s}$ is uncorrelated to f conditioned on P.

Definition 8.5 (Semi-ideal advice). Let $\bar{s}(\mathcal{F}) = (P, \bar{a})$. We say that $\bar{s}$ is semi-ideal if

- $|P| = \mathrm{poly}(n)$, and
- $\bar{s}$ is uncorrelated to f conditioned on P.

Note that the only difference between ideal and semi-ideal advice is that the paths fixed in an ideal advice are promised to have no collision to reveal s, while the paths in a semi-ideal advice may have a collision. It is obvious that if the algorithm is given ideal advice, then the algorithm does not gain noticeable advantage for finding the hidden shift in f.

Claim 5. *Let $\mathcal{A}$ be any classical polynomial-time algorithm. Let $\bar{s}$ be an ideal advice corresponding to $\mathcal{F}$. Then,*

$$\Pr[\mathcal{A}^{\mathcal{F}}(\bar{s}) = s] \leq \mathrm{poly}(n)/2^n.$$

Proof. If $\bar{s}$ is ideal, then there is no collision in P, and $\bar{a}$ is uncorrelated to f conditioned on P. Given $(P, \bar{a})$, $\mathcal{A}$ cannot distinguish whether f is a one-to-one function or a Simon function. Therefore, $\Pr[\mathcal{A}(\bar{s}) = s] \leq \frac{\mathrm{poly}(n)}{2^n}$. □

To prove Theorem 8.1, we also need to show that, given a short ideal advice, there exists a convex combination of sequences of shadows such that a round $\mathcal{B}$ of a d-CQ scheme cannot distinguish the shadows from the original oracle. Furthermore, $\mathcal{B}$ is also ideal with high probability.

Lemma 8.6. *Let $\gamma, \delta_0 \in (0,1)$. Let $\mathcal{F}$ be a (d, f)-shuffling sampled from $\mathcal{D}_{f,d}$. Let p_0 be at most $\mathrm{poly}(n)$. Let $\mathcal{H}$ be a $(p_0, 1+\delta_0)$-uniform shuffling of $\mathcal{F}$ and $\bar{s}$ be a semi-ideal advice. For any round of quantum-classical computing $\mathcal{B}$ in the d-CQ scheme as represented in Equation (15), there exists a convex combination $\mathbf{G}^{\bar{s}}$ of sequences of shadows corresponding to $\bar{s}$ such that for any binary string y,*

$$|\Pr[\mathcal{B}^{\mathcal{H}}(\bar{s}) = y] - \Pr[\mathcal{B}^{\mathbf{G}^{\bar{s}}}() = y]| \leq \gamma + d \cdot \sqrt{\frac{\mathrm{poly}(n)}{2^n}},$$

and the output of $\mathcal{B}^{\mathbf{G}^{\bar{s}}}()$ is semi-ideal. Moreover, if $\bar{s}$ is ideal and the paths fixed in $\mathcal{H}$ has no collision in f, then the output of $\mathcal{B}^{\mathbf{G}^{\bar{s}}}()$ is also ideal with probability at least $1 - \mathrm{negl}(n)$.

Proof. For the ease of the analysis, we allow the classical algorithm $\mathcal{A}_c$ to make path queries. We prove the lemma by mathematical induction on the quantum circuit depth.

Given $\bar{s}$ that is semi-ideal, we let $(P, \bar{a})$ be the output of $\bar{s}$. We define P_0 be the set of paths of P, the set of paths queried by $\mathcal{A}_c$, and the set of paths fixed in $\mathcal{H}$. Since p_0, the number of queries made by $\mathcal{A}_c$, and $|P|$ are at most $\mathrm{poly}(n)$, $|P_0|$ is at most $\mathrm{poly}(n)$.

Then, we let $\bar{s}^{(0)} = (P_0, \bar{a})$ and $z = |\bar{a}|$. Note that the main difficulty that makes the previous proofs not work directly is that $\mathcal{H}$ is generally not $(p_0, 1+\delta_0)$-uniform conditioned on $\bar{s}^{(0)}$. Fortunately, $\mathcal{H}$ is close to a convex combination of $(p+p_0, 1+\delta+\delta_0)$-uniform shuffling functions conditioned on $\bar{s}^{(0)}$ by Claim 3. In particular,

$$\mathcal{H} = \mathbf{H}^{\bar{s}^{(0)}}_{\bar{S}^{(0)}} + \gamma' \mathcal{H}' = \sum_{t_1=1}^{T_1} w_{t_1} \cdot \mathcal{H}_{t_1} + \gamma' \mathcal{H}',$$

where $\mathbf{H}^{\bar{s}^{(0)}}_{\bar{S}^{(0)}}$ is a convex combination of $(p+p_0, 1+\delta+\delta_0)$-uniform shuffling functions, $\mathcal{H}'$ is an arbitrary random shuffling, and $\gamma' \leq \gamma$. According to Claim 3, $p \leq \frac{z - 2\log\gamma}{\log(1+\delta+\delta_0) - \log(1+\delta)} + |P_0|$.

Let $\rho^{(0)}$ be the initial state. Then, we can represent $\mathcal{B}$ with oracle access to $\mathcal{H}$ and advice $\bar{s}$ as $\mathcal{B}^{\mathcal{H}}(\bar{s}, \rho^{(0)}) = \sum_{t_1=1}^{T_1} w_{t_1} \cdot \mathcal{B}^{\mathcal{H}_{t_1}}(\rho^{(0)}) + \gamma' \cdot \mathcal{B}^{\mathcal{H}'}(\rho^{(0)})$. For clarity, we write $\mathcal{B}$ according to Equation (15) and obtain

$$\begin{aligned}\Pi_{0/1} \circ U_{d+1} \mathcal{H} \cdots \mathcal{H} U_1(\rho^{(0)}, \bar{s}^{(0)}) = &\sum_{t_1=1}^{T_1} w_{t_1} \cdot \Pi_{0/1} \circ U_{d+1} \mathcal{H}_{t_1} \cdots \mathcal{H}_{t_1} U_1(\rho^{(0)}) \\ &+ \gamma' \Pi_{0/1} \circ U_{d+1} \mathcal{H}' \cdots \mathcal{H}' U_1(\rho^{(0)}). \end{aligned} \tag{16}$$

It is easy to see that Equation (16) implies that,

$$\left|\Pr[\Pi_{0/1}\circ U_{d+1}\mathcal{H}\cdots\mathcal{H}U_1(\rho^{(0)},\bar{s}^{(0)})=y]-\Pr\left[\sum_{t_1=1}^{T_1}w_{t_1}\cdot\Pi_{0/1}\circ U_{d+1}\mathcal{H}_{t_1}\cdots\mathcal{H}_{t_1}U_1(\rho^{(0)})=y\right]\right|\le\gamma. \tag{17}$$

Then, we construct shadows for $\mathcal{H}_{t_1}$ as follows: Let P_{t_1} be P_0 and the set of paths fixed in $\mathcal{H}_{t_1}$ in addition to P_0. We construct the hidden set $\overline{S}_{t_1}^{(1)}$ based on $\mathcal{H}_{t_1}$, P_{t_1}, and $\overline{S}^{(0)}$ as in Procedure 3. Let $\mathcal{G}_{t_1}^{(1)}$ be the shadow of $\mathcal{H}_{t_1}$ in $\overline{S}_{t_1}^{(1)}$, Then,

$$\begin{aligned}B(\mathcal{H}_{t_1}U_1(\rho^{(0)}),\mathcal{G}_{t_1}^{(1)}U_1(\rho^{(0)}))&\le\sqrt{2\Pr[\text{find }\overline{S}_{t_1}^{(1)}:U_1^{\mathcal{H}_{t_1}\setminus\overline{S}_{t_1}^{(1)}},\rho^{(0)}]}\\&\le\sqrt{(1+\delta+\delta_0)\cdot\frac{2q_1}{2^n}},\end{aligned}$$

where q_1 is the number of queries U_1 performs. The first inequality follows from Lemma 5.7 and the last inequality follows from Claim 4 and Lemma 5.8.

PROCEDURE 3: The hidden sets for d-CQ scheme

Given $j\in\mathbb{N}$, $\overline{S}^{(j-1)}:=(S_{j-1}^{(j-1)},\ldots,S_d^{(j-1)})$, $\mathcal{H}_{k,j}=(h_j,\ldots,h_d)$, and P a set of fixed paths.

(1) Let $S_j^{(j-1)}$ be $S_j^{(j-1)}$ except for elements on P.
(2) Let $S_j^{(j)}$ be a subset chosen uniformly at random with the promise that $|S_j^{(j)}|/|S_j^{(j-1)}|=\frac{1}{2^n}$, and $S_j^{(j)}$ includes all elements in S_j except for elements on P.
(3) For $\ell=j+1,\ldots,d$, let $S_\ell^{(j)}:=\{h_{\ell-1}\circ\cdots\circ h_j(S_j^{(j)})\}$.
(4) We let $\overline{S}^{(j)}=(S_j^{(j)},\ldots,S_d^{(j)})$.

Note that we can use Procedure 3 to construct all hidden sets $\bar{S}_{t_1}^{(1)},\ldots,\bar{S}_{t_1}^{(d)}$. Then, by following the same hybrid argument for proving Theorem 6.1, we can obtain a sequence of shadows $\mathcal{G}_{t_1}^{(1)},\ldots,\mathcal{G}_{t_1}^{(d)}$ such that

$$\begin{aligned}&\left|\Pr[\Pi_{0/1}\circ U_{d+1}\mathcal{H}_{t_1}\cdots\mathcal{H}_{t_1}U_1(\rho^{(0)})=y]-\Pr\left[\Pi_{0/1}\circ U_{d+1}\mathcal{G}_{t_1}^{(d)}\cdots\mathcal{G}_{t_1}^{(1)}U_1(\rho^{(0)})=y\right]\right|\\&\le\left|\Pr[\Pi_{0/1}\circ U_{d+1}\mathcal{H}_{t_1}\cdots\mathcal{H}_{t_1}U_1(\rho^{(0)})=y]-\Pr\left[\Pi_{0/1}\circ U_{d+1}\mathcal{H}_{t_1}\cdots\mathcal{H}_{t_1}U_2\mathcal{G}_{t_1}^{(1)}U_1(\rho^{(0)})=y\right]\right|\\&\quad+\left|\Pr\left[\Pi_{0/1}\circ U_{d+1}\mathcal{H}_{t_1}\cdots\mathcal{H}_{t_1}U_2\mathcal{G}_{t_1}^{(1)}U_1(\rho^{(0)})=y\right]-\Pr\left[\Pi_{0/1}\circ U_{d+1}\mathcal{G}_{t_1}^{(d)}\cdots\mathcal{G}_{t_1}^{(1)}U_1(\rho^{(0)})=y\right]\right|\\&\le\sum_{i=1}^{d}\sqrt{2\Pr[\text{find }\overline{S}_{t_1}^{(i)}:U_i^{\mathcal{H}_{t_1}\setminus\overline{S}_{t_1}^{(i)}},\rho^{(i)}]}\\&\le\sum_{i=1}^{d}\sqrt{(1+\delta+\delta_0)\cdot\frac{2q_i}{2^n}}=d\cdot\sqrt{\frac{\mathrm{poly}(n)}{2^n}}.\end{aligned} \tag{18}$$

Here, $\overline{S}_{t_1}^{(i)}$ is the hidden set built from Procedure 3 (by considering $\overline{S}_{t_1}^{(i-1)}$, $\mathcal{H}_{t_1}$, and P_{t_1}), and $\mathcal{G}_{t_1}^{(i)}$ is the shadow of $\mathcal{H}_{t_1}$ in $\overline{S}_{t_1}^{(i)}$.

Therefore, we have the following inequality following Equations (17) and (18):

$$\left|\Pr[\Pi_{0/1} \circ U_{d+1}\mathcal{H}\cdots\mathcal{H}U_1(\rho^{(0)}, \bar{s}^{(0)}) = y] - \Pr\left[\sum_{t_1=1}^{T_1} w_{t_1} \cdot \Pi_{0/1} \circ U_{d+1}\mathcal{H}_{t_1}\cdots\mathcal{H}_{t_1}U_2\mathcal{G}_{t_1}^{(1)}U_1(\rho^{(0)}) = y\right]\right| \leq \gamma + d \cdot \sqrt{\frac{\text{poly}(n)}{2^n}}.$$

Note that when we set $\gamma = \text{poly}(1/2^n)$, $\gamma + d \cdot \sqrt{\frac{\text{poly}(n)}{2^n}}$ is negligible. Furthermore, since $p \leq \frac{z-2\log\gamma}{\log(1+\delta+\delta_0)-\log(1+\delta)} + |P_0|$, $p + p_0$ is at most $\text{poly}(n)$.

Let $\mathcal{B}^{\mathbf{G}^{\bar{s}}} := \sum_{t_1=1}^{T_1} w_{t_1} \cdot \Pi_{0/1} \circ U_{d+1}\mathcal{G}_{t_1}^{(d)}U_d \cdots \mathcal{G}_{t_1}^{(1)}U_1(\rho^{(0)})$. It is obvious that the output of $\mathcal{B}^{\mathbf{G}^{\bar{s}}}$ is semi-ideal, since $\bar{s}^{(0)}$ is uncorrelated to f conditioned on P_0 and $\mathbf{G}^{\bar{s}}$ has no information about f except for P_0.

If $\bar{s} = (P, \bar{a})$ is ideal and the paths fixed in $\mathcal{H}$ has no collision in f, then there is no collision in f from P and the paths fixed in $\mathcal{H}$. Furthermore, since $\bar{s}^{(0)}$ is uncorrelated to f conditioned on P_0, the additional paths fixed in $\mathcal{H}_{t_1}$ are uncorrelated to f given P_0. Hence, the additional paths fixed by $\bar{a}$ and the queries by $\mathcal{A}_c$ are just random paths. This implies that the output is not ideal if and only if the random paths fixed by $\bar{a}$ and $\mathcal{A}_c$ have a collision, which probability is at most $\text{negl}(n)$. □

Now, we are ready to prove Theorem 8.1.

Proof of Theorem 8.1. We first consider L_1. Since L_1 has no advice, we can use the analysis in Theorem 7.1 to show that there exists a sequence of shadow $\mathcal{G}^{(1)}$ such that

$$\left|\Pr[\mathcal{A}_c^{\mathcal{F}} \circ (L_m \circ \cdots \circ L_1)^{\mathcal{F}}() = s] - \Pr\left[\mathcal{A}_c^{\mathcal{F}} \circ (L_m \circ \cdots \circ L_2)^{\mathcal{F}} \circ L_1^{\mathcal{G}^{(1)}}() = s\right]\right| \leq d\sqrt{\frac{\text{poly}(n)}{2^n}}.$$

Note that $L_1^{\mathcal{G}^{(1)}}$ must be semi-ideal and is ideal with probability at least $1 - \frac{\text{poly}(n)}{2^n}$ following from the fact that $\mathcal{G}^{(1)}$ has no information about f except for the paths queried by the classical algorithm.

For $\Pr[(\mathcal{A}_c \circ L_m \circ \cdots \circ L_2)^{\mathcal{F}} \circ L_1^{\mathcal{G}^{(1)}}() = s]$, given the fact that $L_1^{\mathcal{G}^{(1)}}$ is semi-ideal, there exists a convex combination $\mathbf{H}_1$ of $(p, 1+\delta)$-uniform shufflings by Claim 3 such that

$$|\Pr[(\mathcal{A}_c \circ L_m \circ \cdots \circ L_2)^{\mathcal{F}} \circ L_1^{\mathcal{G}^{(1)}}() = s] - \Pr[(\mathcal{A}_c \circ L_m \circ \cdots \circ L_2)^{\mathbf{H}_1}() = s]| \leq \gamma,$$

where $p \leq \frac{\text{poly}(n)-2\log\gamma}{\log(1+\delta)}$. We write $\mathbf{H}_1 = \sum_{t_1}^{T_1} w_{t_1}\mathcal{H}_{t_1}$ such that $(\mathcal{A}_c \circ L_m \circ \cdots \circ L_2)^{\mathbf{H}_1} = \sum_{t_1=1}^{T_1} w_{t_1}(\mathcal{A}_c \circ L_m \circ \cdots \circ L_2)^{\mathcal{H}_{t_1}}$. By Lemma 8.6, we can further replace each $L_2^{\mathcal{H}_{t_1}}$ by $L_2^{\mathcal{G}_{t_1}^{(1)}}$ such that

$$\left|\Pr\left[\sum_{t_1=1}^{T_1} w_{t_1}(\mathcal{A}_c \circ L_m \circ \cdots \circ L_2)^{\mathcal{H}_{t_1}}() = s\right] - \Pr\left[\sum_{t_1=1}^{T_1} w_{t_1}(\mathcal{A}_c \circ L_m \circ \cdots \circ L_3)^{\mathcal{H}_{t_1}} \circ L_2^{\mathcal{G}_{t_1}^{(1)}}() = s\right]\right| \leq d\sqrt{\frac{\text{poly}(n)}{2^n}}.$$

Similarly, given $L_2^{\mathcal{G}_{t_1}^{(1)}}$ as advice, we can apply Claim 3 to replace $\mathcal{H}_{t_1}$ by a convex combination $\mathbf{H}_{t_1} = \sum_{t_2=1}^{T_2} w_{t_2,t_1} \cdot \mathcal{H}_{t_2,t_1}$ of $(p', 1+\delta')$-uniform shuffling with negligible loss (by setting $\gamma = \text{negl}(n)$)

and then use Lemma 8.6 to obtain the following inequality:

$$\left| \Pr\left[\sum_{t_1=1}^{T_1} w_{t_1} (\mathcal{A}_c \circ L_m \circ \cdots \circ L_3)^{\mathcal{H}_{t_1}} \circ L_2^{\mathcal{G}_{t_1}^{(1)}}() = s\right] - \Pr\left[\sum_{t_1=1}^{T_1}\sum_{t_2=1}^{T_2} w_{t_2,t_1} (\mathcal{A}_c \circ L_m \circ \cdots \circ L_4)^{\mathcal{H}_{t_2,t_1}} \circ L_3^{\mathcal{G}_{t_2,t_1}}() = s\right] \right| \leq \gamma + d\sqrt{\frac{\mathsf{poly}(n)}{2^n}},$$

where $\mathcal{G}_{t_2,t_1}$ is the shadow of $\mathcal{H}_{t_2,t_1}$ by Lemma 8.6.

Along this line, we can have the following inequality by applying Lemma 8.6 from L_1 to L_m:

$$\left| \Pr[(\mathcal{A}_c \circ L_m \circ \cdots \circ L_1)^{\mathcal{F}}() = s] - \Pr\left[\sum_{t_m,\ldots,t_1=1}^{T_m,\ldots,T_1} w_{t_m\ldots t_1} \mathcal{A}_c^{\mathcal{H}_{t_m\ldots t_1}} \circ L_m^{\mathcal{G}_{t_m\ldots t_1}}() = s\right] \right| \leq \mathsf{negl}(n). \tag{19}$$

Then, by Lemma 8.6 and the union bound, the output of $L_m^{\mathcal{G}_{t_m\ldots t_1}}()$ is ideal with probability $1 - \mathsf{negl}(n)$, and thus

$$\Pr\left[\sum_{t_m,\ldots,t_1=1}^{T_m,\ldots,T_1} w_{t_m\ldots t_1} \mathcal{A}_c^{\mathcal{H}_{t_m\ldots t_1}} \circ L_m^{\mathcal{G}_{t_m\ldots t_1}}() = s\right]$$

is negligible following Claim 5. This, along with Equation (19), implies that $\Pr[(\mathcal{A}_c \circ L_m \circ \cdots \circ L_1)^{\mathcal{F}}() = s] = \mathsf{negl}(n)$. □

8.3 On Separating the Depth Hierarchy of d-CQ Scheme

We show that the d-SSP is also hard for any d-CQ scheme by the same proof for Theorem 8.1.

Theorem 8.7. *The d-SSP cannot be decided by any d-CQ scheme $\mathcal{A}$ with probability greater than $\frac{1}{2} + \sqrt{\frac{\mathsf{poly}(n)}{2^n}}$.*

Proof. Following the proof for Theorem 8.1, there exists a convex combination $\sum_{t=1}^{T} p_t(\mathcal{G}_t^{(1)}, \ldots, \mathcal{G}_t^{(m)})$ such that

$$\left| \Pr\left[L_m^{\mathcal{F}} \circ \cdots \circ L_1^{\mathcal{F}}() = s\right] - \sum_{t=1}^{T} p_t \cdot \Pr\left[\left(L_m^{\mathcal{G}_t^{(m)}} \circ \cdots \circ L_1^{\mathcal{G}_t^{(1)}}() = s\right)\right] \right| \leq md \cdot \sqrt{\frac{\mathsf{poly}(n)}{2^n}}.$$

Moreover, in case that f is a random Simon function, $\mathcal{A}$ with access to $\sum_{t=1}^{T} p_t(\mathcal{G}_t^{(1)}, \mathcal{G}_t^{(2)}, \ldots, \mathcal{G}_t^{(m)})$ can only find s with probability at most $\sqrt{\frac{\mathsf{poly}(n)}{2^n}}$. Following the analysis similar to Theorem 7.3, we can conclude that $\Pr[\mathcal{A}^{\mathcal{F}}() = 1] \leq 1/2 + \sqrt{\frac{\mathsf{poly}(n)}{2^n}}$. □

Corollary 8.8. *For any $d \in \mathbb{N}$, there is a $(2d+1)$-CQ scheme that can solve the d-SSP with high probability, but there is no d-CQ scheme that can solve the d-SSP.*

Proof. This corollary follows from Theorem 8.7 and Theorem 4.11 directly. □

Finally, we can conclude that

Corollary 8.9. *Let O and $\mathcal{L}(O)$ be defined as in Definition 4.10. $\mathcal{L}(O) \in \mathsf{BQP}^O$ and $\mathcal{L}(O) \notin (\mathsf{BPP}^{\mathsf{BQNC}})^O$.*

Proof. Note that each $n \in \mathbb{N}$, $\mathcal{O}_{unif}^{f_n, d(n)} \in \mathcal{O}$ has depth equal to the input size. A quantum circuit with depth poly(n) can decide whether f_n is a Simon's function by Theorem 4.11 and thus decides if 1^n is in $\mathcal{L}(\mathcal{O})$. However, for d-CQ scheme that only has quantum depth $d = \text{poly} \log n$, it cannot decide the language by Theorem 8.7. □

APPENDIX

A TABLE OF NOTATION

Table 1. Summary of Notations

	Notations	Descriptions
Shuffling oracle for f (Section 4.1)	f	an arbitrary function from $\mathbb{Z}_2^n$ to $\mathbb{Z}_2^n$
	$f_0, \ldots, f_{d-1}$	random permutations in $P(\mathbb{Z}_2^{(d+2)n}, \mathbb{Z}_2^{(d+2)n})$
	f_d	$f_d \circ f_{d-1} \circ \cdots \circ f_0(x) = f(x)$
	$\mathcal{F}$	$(f_0, \ldots, f_d)$
	$\mathbf{SHUF}(d, f)$	set of all $\mathcal{F}$ with fixed d and f
	$\mathcal{O}_{\text{unif}}^{f,d}$	random shuffling oracle picked from $\mathbf{SHUF}(d, f)$
	Path $(x_0, \ldots, x_{d+1})$	$f_i(x_i) = x_{i+1}$ for $i = 0, \ldots, d$
Domains of $f_0, \ldots, f_d$ (Section 5)	S_0	$\{0, \ldots, 2^n - 1\}$
	$S_1, \ldots, S_d$	$S_i = f(S_{i-1})$ for $i \in [d]$
	$\overline{S}$	$(S_0, \ldots, S_d)$
	$S_0^{(0)}, \ldots, S_d^{(0)}$	$\mathbb{Z}_2^{(d+2)n}$
	$\overline{S}^{(0)}$	$(S_0^{(0)}, \ldots, S_d^{(0)})$
	$S_j^{(\ell)}$	$S_j^{(\ell)} \subset S_j^{(\ell-1)}$, $\frac{\lvert S_j^{(\ell)} \rvert}{\lvert S_j^{(\ell-1)} \rvert} \leq \frac{1}{2^n}$, $f_j(S_{j-1}^{(\ell)}) = S_j^{(\ell)}$, and $S_j \subseteq S_j^{(\ell)}$.
	$\overline{S}^{(\ell)}$	$(S_\ell^{(\ell)}, \ldots, S_d^{(\ell)})$
Functions w.r.t. $\overline{S}^{(\ell)}$ and $\mathcal{F}$ (Section 5)	$g_\ell, \ldots, g_d$	if $x \in S_j^{(\ell)}$, $g_j(x) = \perp$; otherwise, $g_j(x) = f_j(x)$
	$\mathcal{G}$ of $\mathcal{F}$ in $\overline{S}^{(\ell)}$	$\mathcal{G} := (f_0, \ldots, f_{\ell-1}, g_\ell, \ldots, g_d)$
	$f_j^{(\ell)}$	f_j on $S_j^{(\ell-1)} \setminus S_j^{(\ell)}$
	$\hat{f}_j^{(\ell)}$	f_j on $S_j^{(\ell)}$
	$\mathcal{F}^{(1)}$	$(f_0, f_1^{(1)}, \ldots, f_d^{(1)})$
	$\mathcal{F}^{(d+1)}$	$(\hat{f}_d^{(d)})$
	$\mathcal{F}^{(\ell)}$ for $\ell = 2, \ldots, d-1$	$(\hat{f}_{\ell-1}^{(\ell)}, f_\ell^{(\ell)}, \ldots, f_d^{(\ell)})$
	$\hat{\mathcal{F}}^{(0)}$	$(f_0, \ldots, f_d)$
	$\hat{\mathcal{F}}^{(\ell)}$ for $\ell = 1, \ldots, d$	$(\hat{f}_\ell^{(\ell)}, \ldots, \hat{f}_d^{(\ell)})$

ACKNOWLEDGMENT

We are grateful to Scott Aaronson for bringing us to the problem of separating BQP and $\mathrm{BPP}^{\mathrm{BQNC}}$. We also thank him for helpful discussions and valuable comments on our manuscript. We would like to thank Richard Jozsa for clarifying his conjecture and Aaronson's conjecture. We acknowledge conversations with Matthew Coudron for explaining their related work to us.

REFERENCES

[1] Scott Aaronson. 2005. Ten Semi-grand Challenges for Quantum Computing Theory. Retrieved from https://www.scottaaronson.com/writings/qchallenge.html.
[2] Scott Aaronson. 2010. BQP and the polynomial hierarchy. In *42nd ACM Symposium on Theory of Computing (STOC'10)*. ACM, New York, NY, 141–150. DOI: https://doi.org/10.1145/1806689.1806711
[3] Scott Aaronson. 2011. Projects Aplenty. Retrieved from https://www.scottaaronson.com/blog/?p=663.
[4] Scott Aaronson. 2019. Personal communication.
[5] Scott Aaronson and Lijie Chen. 2017. Complexity-theoretic foundations of quantum supremacy experiments. In *32nd Computational Complexity Conference (CCC'17)*. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, DEU.
[6] Andris Ambainis, Mike Hamburg, and Dominique Unruh. 2018. Quantum security proofs using semi-classical oracles. *IACR Cryptology ePrint Archive* 2018 (2018), 904.
[7] Sanjeev Arora and Boaz Barak. 2009. *Computational Complexity: A Modern Approach* (1st ed.). Cambridge University Press.
[8] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Burkett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowler, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Mike Lindmark, Erik Lucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xiao Mi, Kristel Michielsen, Masoud Mohseni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis. 2019. Quantum supremacy using a programmable superconducting processor. *Nature* 574, 7779 (2019), 505–510.
[9] Nir Bitansky, Ran Canetti, Henry Cohn, Shafi Goldwasser, Yael Tauman Kalai, Omer Paneth, and Alon Rosen. 2014. The impossibility of obfuscation with auxiliary input or a universal simulator. In *Advances in Cryptology – CRYPTO 2014*. Springer Berlin, 71–89.
[10] Michael J. Bremner, Richard Jozsa, and Dan J. Shepherd. 2010. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proc. R. Soc. A* 467, 2126 (2010), 459–472.
[11] Andrew M. Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A. Spielman. 2003. Exponential algorithmic speedup by a quantum walk. In *35th Annual ACM Symposium on Theory of Computing (STOC'03)*. Association for Computing Machinery, New York, NY, 59–68. DOI: https://doi.org/10.1145/780542.780552
[12] R. Cleve and J. Watrous. 2000. Fast parallel circuits for the quantum Fourier transform. In *41st Annual Symposium on Foundations of Computer Science*. 526–536.
[13] Sandro Coretti, Yevgeniy Dodis, Siyao Guo, and John P. Steinberger. 2018. Random oracles and non-uniformity. In *EUROCRYPT (1)*. Springer, 227–258. DOI: https://doi.org/10.1007/978-3-319-78381-9_9
[14] Matthew Coudron and Sanketh Menda. 2019. Computations with Greater Quantum Depth Are Strictly More Powerful (Relative to an Oracle). arXiv:arXiv:1909.10503.
[15] Matthew Coudron and Sanketh Menda. 2019. Personal communication.
[16] Peter Høyer and Robert Špalek. 2003. Quantum circuits with unbounded fan-out. In *Symposium on Theoretical Aspects of Computer Science*. Springer Berlin, 234–246.
[17] Peter Høyer and Robert Špalek. 2005. Quantum fan-out is powerful. *Theor. Comput.* 1 (01 2005), 81–103.
[18] IBM. 2017. IBM Announces Advances to IBM Quantum Systems & Ecosystem. Retrieved from https://www-03.ibm.com/press/us/en/pressrelease/53374.wss.
[19] Richard Jozsa. 2005. An introduction to measurement based quantum computation. *Quant. Inf. Process.* 199 (09 2005).
[20] Julian Kelly. 2018. A Preview of Bristlecone, Google's New Quantum Processor. Retrieved from https://ai.googleblog.com/2018/03/a-preview-of-bristlecone-googles-new.html.
[21] Cristopher Moore and Martin Nilsson. 1998. Parallel Quantum Computation and Quantum Codes. arXiv:arXiv:quant-ph/9808027.

[22] D. R. Simon. 1994. On the power of quantum computation. In *35th Annual Symposium on Foundations of Computer Science (SFCS'94)*. IEEE Computer Society, 116–123. DOI: https://doi.org/10.1109/SFCS.1994.365701
[23] Barbara M. Terhal and David P. DiVincenzo. 2004. Adaptive quantum computation, constant depth quantum circuits and Arthur-Merlin games. *Quant. Inf. Computat.* 4, 2 (2004), 134–145. Retrieved from http://dblp.uni-trier.de/db/journals/qic/qic4.html#TerhalD04.
[24] Dominique Unruh. 2007. Random oracles and auxiliary input. In *Advances in Cryptology - CRYPTO 2007*. Springer Berlin, 205–223.
[25] Dominique Unruh. 2015. Revocable quantum timed-release encryption. *J. ACM* 62, 6 (Dec. 2015).

Received 16 September 2020; revised 9 October 2022; accepted 24 October 2022