The *Learning Stabilizers with Noise* problem

Alexander Poremba[*1,2], Yihui Quek[†1], and Peter Shor[‡1]

[1]Department of Mathematics, Massachusetts Institute of Technology
[2]Computer Science & Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology

Abstract

Random classical codes have good error correcting properties, and yet they are notoriously hard to decode in practice. Despite many decades of extensive study, the fastest known algorithms still run in exponential time. The *Learning Parity with Noise* (LPN) problem, which can be seen as the task of decoding a random linear code in the presence of noise, has thus emerged as a prominent hardness assumption with numerous applications in both cryptography and learning theory.

Is there a natural quantum analog of the LPN problem? In this work, we introduce the *Learning Stabilizers with Noise* (LSN) problem, the task of decoding a random stabilizer code in the presence of local depolarizing noise. We give both polynomial-time and exponential-time quantum algorithms for solving LSN in various depolarizing noise regimes, ranging from extremely low noise, to low constant noise rates, and even higher noise rates up to a threshold. Next, we provide concrete evidence that LSN is hard. First, we show that LSN includes LPN as a special case, which suggests that it is at least as hard as its classical counterpart. Second, we prove a worst-case to average-case reduction for variants of LSN. We then ask: what is the computational complexity of solving LSN? Because the task features quantum inputs, its complexity cannot be characterized by traditional complexity classes. Instead, we show that the LSN problem lies in a recently introduced (distributional and oracle) *unitary synthesis class*. Finally, we identify several applications of our LSN assumption, ranging from the construction of quantum bit commitment schemes to the computational limitations of learning from quantum data.

[*]poremba@mit.edu
[†]yquek@mit.edu
[‡]shor@mit.edu

Contents

1 Introduction

Coding theory has offered many valuable insights into the theory of computation, ranging from structural insights in complexity theory [Din06, ABN23], to the design of cryptographic primitives [Sha79, Ste93, FS96, McE78, Ale03] and even to lower bounds in computational learning theory [BKW03, FGKP06]. The existence of asymptotically good error correcting codes, in particular, is a major cornerstone in the field. Thanks to the probabilistic method, we know that a random linear code already attains the so-called Gilbert-Varshamov bound [Gil52, Var64] with high probability. This suggests that asymptotically good error correcting codes not only exist in theory, but are in fact also abundant. Despite their remarkable error correcting properties, random linear codes have been found to be notoriously hard to decode in practice, and the fastest known algorithms still run in exponential time [BKW03].

Learning Parity with Noise. The observation that better codes seem harder to decode is captured by the *Learning Parity with Noise* (LPN) problem [BFKL94]. In a nutshell, this assumption says that it is computationally difficult to decode a random linear code under Bernoulli noise. In other words, given as input

$$(\mathbf{A} \sim \mathbb{Z}_2^{n\times k}, \mathbf{A}\cdot\mathbf{x} + \mathbf{e} \pmod 2)$$

it is hard to find the string $\mathbf{x}$ which is chosen uniformly at random in $\mathbb{Z}_2^k$, and where $\mathbf{e} \sim \mathsf{Ber}_p^{\otimes n}$ is a random Bernoulli error for some appropriate noise rate $p \in (0, 1/2)$. Here, $\mathbf{A} \in \mathbb{Z}_2^{n\times k}$ serves as a random *generator matrix* of a linear code, for $n = \mathsf{poly}(k)$.

In practice, LPN is believed to be hard for both classical and quantum algorithms running in time $\mathsf{poly}(k)$ in various noise regimes. For constant noise rates $p \in (0, 1/2)$, the celebrated BKW algorithm [BKW03] solves LPN in both time and sample[1] complexity given by $O(2^{k/\log k})$. The conjectured hardness of LPN has found applications in both cryptography [HB01, Ale03, JW05, ACPS09, JKPT12, DDN14, AHI+17, BLVW18] and learning theory [BFKL94, FGKP06]. The *Learning with Errors* (LWE) problem [Reg09]—a more structured variant of LPN—has since become the basis of modern cryptography and has even led to highly advanced cryptographic primitives, such as fully homomorphic encryption [Gen09, BV11] and the classical verification of quantum computations [Mah22]. In the context of learning theory, it was shown that an efficient algorithm for LPN would allow us to learn important function classes, such as 2-DNF formulas, juntas, and even more general functions with sparse Fourier spectrum [FGKP06].

Because the LPN problem is so prevalent in many areas of computer science, a significant effort has been devoted to finding evidence of its hardness. One of these pieces of evidence is a *worst-to-average-case reduction* [BLVW18, YZ21]. Recall that the LPN problem is an average-case problem: the task is to decode a *random* code, secret and error. Ref. [BLVW18] studied a related worst-case problem—the *nearest codeword problem* (NCP)—and showed that it reduces to LPN. This reduction, later improved by [YZ21], showed that LPN is at least as hard as (a mildly hard variant of) NCP in the worst case. [BLVW18] also found the first non-trivial complexity upper bound on the hardness of the LPN problem; specifically, they showed that the LPN problem is contained in $\mathsf{SearchBPP}^{\mathsf{SZK}}$, and is thus unlikely to be NP-hard.

[1] Here, the number of samples refers to the parameter n—the number of *noisy linear equations* on $\mathbf{x}$.

The hardness of decoding random stabilizer codes. Just like random linear codes, random *quantum* stabilizer codes[2] also possess remarkable error correcting properties [Smi06, Got24]. They are ubiquitous in quantum information science; for example, random stabilizer codes appear in the context of quantum authentication schemes and the verification of quantum computations [ABEM17], quantum cryptography [DS18], the theory of quantum communication [Smi06, Wil13], and even in black-hole physics and quantum gravity [HP07, YK17, HH13]. Characterizing the hardness and complexity of decoding random stabilizer codes is therefore not only important from the perspective of quantum error correction, but could also shed a new light on the computational limitations of quantum information processing as a whole.

And yet, this subject has seen little theoretical treatment. While prior work has shown that decoding quantum stabilizer codes is *worst-case* hard [HL11, IP15]—via reduction from a purely classical decoding problem—the average-case complexity of decoding stabilizer codes as an inherently *quantum problem* was left as an open problem [IP15]. We re-formulate this as a question that bears on all of the areas mentioned above:

Can we find a natural quantum analog of the Learning Parity with Noise problem; in particular, what would its hardness imply for quantum information science as a whole?

Given the success of constructing cryptographic primitives from the hardness of LPN in a classical world, could such a quantum analog of LPN allow us to directly construct cryptographic protocols in a quantum world? This follows a recent line of work arguing that one should build quantum cryptography from inherently quantum, rather than classical cryptographic hardness assumptions [Kre21, MY22, BEM+23, MPSY24, BHHP24]. Finally—and perhaps, even more interestingly—such a quantum assumption may turn out to be even harder to break than its classical counterpart.

2 Overview

We now give an overview of our contributions in this work, summarized in Table 1.

2.1 Learning Stabilizers with Noise

In this work, we introduce a natural quantum analog of LPN—the *Learning Stabilizers with Noise* (LSN) problem. In studying the LSN problem, we thoroughly characterize the hardness and complexity of decoding random stabilizer codes in different noise regimes. Similar to the LPN problem, which has found numerous applications in both cryptography and learning theory, we believe that our LSN assumption has the potential to occupy a similar role in quantum information more broadly. Before we introduce our LSN task formally, we first revisit LPN and draw a connection to quantum error correction.

A quantum analog of LPN? Let $n, k \in \mathbb{N}$ be integers with $n = \mathsf{poly}(n)$, and let $p \in (0, 1/2)$ be a parameter. Recall that an instance of the LPN problem[3] consists of a generator matrix for a random linear code, together with a noisy codeword for a uniformly random string; specifically, we consider samples of the form

$$(\mathbf{A} \sim \mathbb{Z}_2^{n\times k}, \mathbf{A}\cdot\mathbf{x} + \mathbf{e} \pmod 2)$$

[2] The stabilizer formalism was first developed by Gottesman [Got97] and incorporates the majority of quantum error correcting codes we know today [AF24].

[3] For a more formal definition, see Definition 5.1.

	Learning Parity with Noise	Learning Stabilizers with Noise (this problem)
Worst-case hardness	✓ NP-complete [BMT78] as a decisional syndrome decoding task. Variant of the *(Promise) Nearest Codeword Problem* (NCP) [BLVW18]	As a classical syndrome decoding task: NP-complete [HL11, KL12] or #P-complete [IP15] depending on the decoding problem
Average-case hardness	✓ $\mathsf{SearchBPP}^{\mathsf{SZK}}$ [BLVW18]	This paper (Section 8)
Worst-to-average-case reductions	✓[BLVW18, YZ21]	This paper (Section 7)
Algorithms for average-case problem	✓ $2^{O(k/\log k)}$ time/sample complexity [BKW00] in constant-noise regime.	This paper (Section 6)

Table 1: Comparison of LPN and LSN in terms of hardness and complexity

where $\mathbf{A} \in \mathbb{Z}_2^{n\times k}$ is a random generator matrix, where $\mathbf{A}\cdot\mathbf{x}+\mathbf{e} \pmod 2$ is a noisy codeword which encodes uniformly random string $\mathbf{x} \sim \mathbb{Z}_2^k$, and where $\mathbf{e} \sim \mathrm{Ber}_p^{\otimes n}$ is a random Bernoulli error. Without loss of generality[4], we assume that the matrix $\mathbf{A}$ has full column-rank, i.e., the columns of $\mathbf{A}$ generate all of $\mathbb{Z}_2^k$. We now make a simple observation; namely, we can interpret the LPN instance $\mathbf{A}\cdot\mathbf{x}+\mathbf{e} \pmod 2$ as a particular noisy quantum codeword on n qubits[5], since

$$\begin{aligned} |\mathbf{A}\cdot\mathbf{x}+\mathbf{e} \pmod 2\rangle &= X^{\mathbf{e}}\,|\mathbf{A}\cdot\mathbf{x} \pmod 2\rangle \\ &= X^{\mathbf{e}} U_{\mathbf{A}} \left(|0^{n-k}\rangle \otimes |\mathbf{x}\rangle\right), \end{aligned} \tag{1}$$

where $X^{\mathbf{e}} = X^{e_1}\otimes\cdots\otimes X^{e_n}$ is a product of low-weight Pauli-X operators and where the unitary operator $U_{\mathbf{A}}$ is defined to be the matrix multiplication operation

$$U_{\mathbf{A}} : |0^{n-k}\rangle \otimes |\mathbf{x}\rangle \to |\mathbf{A}\cdot\mathbf{x} \pmod 2\rangle\,. \tag{2}$$

Because $\mathbf{A}$ has full column-rank, $U_{\mathbf{A}}$ corresponds to a linear reversible circuit which can be described solely in terms of CNOT gates [PMH08]. Therefore, $U_{\mathbf{A}}$ is a Clifford operator and thus maps Pauli operators to Pauli operators via conjugation.

We may also observe that $U_{\mathbf{A}}$ is the encoding circuit for a stabilizer code. The stabilizer group associated with this code is precisely the group of k commuting Pauli operators under which $U_{\mathbf{A}}(|0^{n-k}\rangle \otimes |\mathbf{x}\rangle)$ remains invariant. These are easily seen to be the Pauli operators

$$U_{\mathbf{A}} Z_i U_{\mathbf{A}}^\dagger, \quad \text{for } i \in [n-k]\,,$$

where Z_i denotes a Pauli operator which acts as a Pauli-Z operator on the i-th qubit, and is equal to the identity everywhere else. In other words, the Clifford encoding unitary $U_{\mathbf{A}}$ (derived from an instance of an LPN problem) gives rise to the quantum stabilizer code[6] given by $S_{\mathbf{A}} = \langle U_{\mathbf{A}} Z_1 U_{\mathbf{A}}^\dagger, \ldots, U_{\mathbf{A}} Z_{n-k} U_{\mathbf{A}}^\dagger\rangle$. This shows that every instance of LPN can be

[4]This happens with overwhelming probability for $\mathbf{A} \sim \mathbb{Z}_2^{n\times k}$ provided that $n \gg k$ (see Section 5.2).
[5]Strictly speaking, we should think of $|\mathbf{A}\cdot\mathbf{x}+\mathbf{e} \pmod 2\rangle$ as encoding the row vector $\mathbf{x}^\intercal\mathbf{A}+\mathbf{e}^\intercal \pmod 2$.
[6]See Section 3.4 for additional background on stabilizer codes.

mapped to an instance of decoding stabilizer codes. We now generalize this significantly, ultimately leading us to the *Learning Stabilizers with Noise* (LSN) problem—the natural quantum analog of LPN:

- (Random stabilizer code:) Note that the encoding Clifford unitary in Equation (2) generates a specific stabilizer code of the form $S_{\mathbf{A}} = \langle U_{\mathbf{A}} Z_1 U_{\mathbf{A}}^\dagger, \ldots, U_{\mathbf{A}} Z_{n-k} U_{\mathbf{A}}^\dagger \rangle$. We consider stabilizer subgroups of the Pauli group which are chosen uniformly at random from the set of all stabilizer subgroups with $n-k$ generators, denoted by $\mathsf{Stab}(n,k)$. In fact, as we later prove in Theorem 3.9, this is equivalent to choosing random stabilizer codes which are described by $\langle CZ_1C^\dagger, \ldots, CZ_{n-k}C^\dagger \rangle$, where $C \sim \mathsf{Cliff}_n$ is a random n-qubit Clifford operator.
- (Local depolarizing noise:) Recall that the noisy codeword $X^{\mathbf{e}} U_{\mathbf{A}}(|0^{n-k}\rangle \otimes |\mathbf{x}\rangle)$ in Equation (1) is only affected by low-weight bit-flip errors $X^{\mathbf{e}}$, where $\mathbf{e} \sim \mathsf{Ber}_p^{\otimes n}$ comes from a Bernoulli distribution. In quantum systems, however, noise may also come in the form of phase errors. This leads us to consider a quantum noise model in the form of local depolarizing noise $\mathcal{D}_p^{\otimes n}$. Similar to the Bernoulli distribution, local depolarizing noise also produces low-weight errors with high probability, which therefore naturally generalizes the classical noise model.

In other words, we consider the task of decoding a random quantum stabilizer code in the presence of local depolarizing noise. Because the codeword is a *stabilizer state*, we call this the *Learning Stabilizers with Noise* (LSN) problem—in analogy to the classical LPN problem. We now give a formal definition of the problem.

Learning Stabilizers with Noise. The *Learning Stabilizers with Noise* (LSN) problem (formally defined in Definition 4.1) is to find $x \in \{0,1\}^k$ given as input a sample

$$\left(S \in \mathsf{Stab}(n,k),\ E\,|\overline{\psi_x}\rangle^S\right) \quad \text{with} \quad |\overline{\psi_x}\rangle^S := U_{\mathrm{Enc}}^S(|0^{n-k}\rangle \otimes |x\rangle)\,,$$

where $S \sim \mathsf{Stab}(n,k)$ is a uniformly random stabilizer subgroup, where $E \sim \mathcal{D}_p^{\otimes n}$ is a Pauli error from a local depolarizing channel, where $x \sim \{0,1\}^k$ is a random string, and where U_{Enc}^S is some canonical encoding circuit for the stabilizer code associated with S. As mentioned before, the encoding circuit is typically given in the form of a random n-qubit Clifford operator.

At first sight, it may not be clear why the LSN problem is even well-defined, since a unique solution to the decoding problem may not exist in certain parameter regimes. The intuition behind our argument for the existence of a unique solution is as follows. Suppose that $p \in (0, 1/2)$ is a sufficiently small constant. Then, the quantum Gilbert-Varshamov bound (see Section 3.4) tells us that a random stabilizer code is non-degenerate and has distance at least $d = 3np + 1$ with overwhelming probability, in which case for any pair of codewords with $x, y \in \{0,1\}^k$, we have

$$\langle \overline{\psi_x}|\, E_a^\dagger E_b\, |\overline{\psi_y}\rangle = 0$$

by the *Knill-Laflamme conditions*—provided that the errors E_a, E_b have weight at most $|E_a|, |E_b| \leq \frac{3}{2}np$. Fortunately, a simple Chernoff bound analysis reveals that this is the case with overwhelming probability for the local depolarizing channel $\mathcal{D}_p^{\otimes n}$. Therefore, Pauli errors which originate from a local depolarizing noise channel take orthogonal codewords to orthogonal codewords, and hence there must exist a measurement that perfectly distinguishes between them. This observation is also at the core of our algorithms for the LSN problem, which we describe next.

Algorithms for Learning Stabilizers with Noise. In the previous section, we discussed why random stabilizer codes give rise to a single-shot decoding problem which exhibits unique solutions with high probability. This suggests that LSN can be solved at least information-theoretically. Can we find efficient algorithms for solving the LSN problem? Not surprisingly, the answer depends on the specific noise regime of the error distribution. In Section 6, we give both polynomial-time and exponential-time algorithms for solving LSN in various noise-regimes.

- **extremely low-noise** regime with parameter $p \leq \frac{1}{n} - \frac{1}{n^{1+c}}$, for some $c > 0$. In this regime, we show that a simple projection onto the stabilizer codespace (see Algorithm 1) suffices to solve the LSN problem in time $O(n^3)$ with inverse-polynomial success probability at least $1/n^c$.
- **low constant-noise** regime for a some small constant $p \in (0, 1/2)$. In this regime, we show that with only a single sample, the *Pretty Good Measurement* (PGM) [BK00, Mon07] succeeds with high probability to perfectly recover the secret in the LSN problem. This is inextricably linked to the structure of our problem; although the distance between two arbitrary orthogonal states contracts tremendously—in fact, exponentially in system size (see, e.g. Proposition IV.7 in [HRF23])—under a layer of local depolarizing noise, a good error correcting code encodes orthogonal states into orthogonal subspaces that are "protected" from such destructive contraction even under noise. This means the information in them is still recoverable after noise, and this is the intuition behind our PGM algorithm.
- **higher constant-noise** regime, decoding is still possible at the cost of more samples. We derive the scaling of the sample complexity with noise, up to a certain noise threshold.

Worst-case to average-case reductions. Recall that the LSN decoding problem is stated as an *average-case* problem, where the success probability of an algorithm is measured on average over the random choice of stabilizer $S \in \mathsf{Stab}(n, k)$, secret $x \in \mathbb{Z}_2^k$ and error $E \sim \mathcal{D}_p^{\otimes n}$. While the quantum Gilbert-Varshamov bound does in fact guarantee that an average-case instance of LSN can be solved information-theoretically, our results in Section 6 indicate that the problem becomes computationally intractable for large k—even in a low constant-noise regime. This raises the question of whether we can find concrete evidence for the average-case hardness of the LSN problem, beyond the fact that it subsumes the classical LPN problem.

Recently, a number of works showed that there is in fact evidence of *worst-case* hardness for LPN; specifically, by studying a related worst-case problem—the *nearest codeword problem* (NCP) [BLVW18, YZ21]. Using the *sample amplification* technique [Lyu05], Brakerski, Lyubashevsky, Vaikuntanathan and Wichs [BLVW18] gave a worst-case to average-case reduction from NCP to LPN. Here, an instance to the former problem consists of $(\mathbf{C} \in \mathbb{Z}_2^{m \times k}, \mathbf{t} = \mathbf{C} \cdot \mathbf{s} + \mathbf{w} \pmod 2)$ for some $\mathbf{s} \in \mathbb{Z}_2^k$, with the promise that the generator matrix $\mathbf{C}$ is balanced[7] and that the Hamming weight of the error $\mathbf{w} \in \mathbb{Z}_2^m$ is known. On a high level, the reduction in [BLVW18] proceeds in two steps:

- (Re-randomization of the secret) A random string $\mathbf{u} \sim \mathbb{Z}_2^k$ is chosen, and the worst-case instance $(\mathbf{C}, \mathbf{t})$ gets mapped via an additive shift to

$$(\mathbf{C},\ \mathbf{t} + \mathbf{C} \cdot \mathbf{u} \pmod 2) = (\mathbf{C}, \mathbf{C} \cdot (\mathbf{s} + \mathbf{u}) + \mathbf{w} \pmod 2).$$

[7] Roughly speaking, this means that the minimum and maximum distance of the linear code generated by $\mathbf{C} \in \mathbb{Z}_2^{n \times m}$ is neither too small nor too large.

Note that, whereas the initial secret $\mathbf{s} \in \mathbb{Z}_2^k$ was fixed, the new secret $\mathbf{s} + \mathbf{u}$ is now distributed according to the uniform distribution over $\mathbb{Z}_2^k$.

- (Re-randomization of the code and error) A random $\mathbf{R} \in \mathbb{Z}_2^{n \times m}$ is sampled from a *smoothing distribution* $\mathcal{R}_w^{n \times m}$, and the previous sample gets mapped to

$$(\mathbf{R} \cdot \mathbf{C}, \mathbf{R} \cdot (\mathbf{C} \cdot (\mathbf{s} + \mathbf{u}) + \mathbf{w} \pmod 2)) = (\mathbf{R} \cdot \mathbf{C}, \mathbf{R} \cdot \mathbf{C} \cdot (\mathbf{s} + \mathbf{u}) + \mathbf{R} \cdot \mathbf{w} \pmod 2))$$

[BLVW18] show that the resulting sample is statistically close to an (average-case) LPN sample—provided that the *smoothing distribution* $\mathcal{R}_w^{n \times m}$ is chosen appropriately.

By taking a similar approach, we develop a worst-case to average-case reduction for the LSN problem. Here, the starting point is a worst-case stabilizer decoding instance $(S, E\,|\overline{\psi}_x\rangle^S)$ for some stabilizer $S \in \mathsf{Stab}(n,k)$, Pauli error E of bounded weight, and secret $x \in \{0,1\}^k$. First, we observe that, in order to re-randomize the secret x, we need to act on the encoded data $|\overline{\psi}_x\rangle^S$ itself. Hence, it suffices to choose a random string $u \sim \{0,1\}^k$ and to apply the *logical* Pauli operator $\overline{X}^u$ associated with S to the noisy codeword itself, resulting in the desired transformation $E\,|\overline{\psi}_{x\oplus u}\rangle^S$ up to a sign.

To re-randomize the code and the error, we first observe that the shifted codeword $|\overline{\psi}_{x\oplus u}\rangle^S$ can be written as

$$|\overline{\psi}_{x\oplus u}\rangle^S = U_{\mathrm{Enc}}^S(|0^{n-k}\rangle \otimes |x \oplus u\rangle)$$

for some (not necessarily random) encoding Clifford $U_{\mathrm{Enc}}^S \in \mathsf{Cliff}_n$. Because Cliff_n forms a finite group, this suggests that one could simply sample a uniformly random $C \sim \mathsf{Cliff}_n$ and consider the state $CE\,|\overline{\psi}_{x\oplus u}\rangle^S = (CEC^\dagger)C\,|\overline{\psi}_{x\oplus u}\rangle^S$, where $C\,|\overline{\psi}_{x\oplus u}\rangle^S$ now comes from a random stabilizer code for a uniformly random encoding Clifford $C \cdot U_{\mathrm{Enc}}^S$. While this does seem to result in a re-randomized stabilizer code, the aforementioned transformation could potentially *blow up* the weight of the Pauli error E. In fact, it is well-known that random Cliffords are *Pauli-mixing* [CLLW16, ABEM17]: they take any non-identity Pauli operator and map it to a uniformly random non-identity Pauli operator via conjugation. This seems to suggest that any naive worst-case to average-case reduction for the LSN problem is doomed to fail, since the weight of the re-randomized Pauli error now follows a Binomial distribution with parameter $3/4$, which would result in an average weight of $O(n)$—thereby potentially making decoding information-theoretically impossible.

To overcome this barrier we develop a re-randomization strategy which is much more *gentle* on the error (i.e., it does not cause it to blow up), and yet still ensures that the code gets somewhat re-randomized. Our strategy is to follow the random Pauli operator with a *twirl*—another random unitary consisting of a random permutation operator, followed by a layer of random single-qubit Cliffords. Similar ensembles of unitaries been used in the *randomized compiling and benchmarking* literature in order to tailor noise with arbitrary coherence and spatial correlations into a symmetric Pauli channel [WE16, ESM+07]. To our knowledge, however, our work is the first to identify the twirl as a useful tool in the context of a worst-case to average-case reduction. Under this twirl, a worst-case error of weight w is transformed into a uniformly random error of weight w; in particular, the weight of the error remains invariant. We remark, however, that the distributions of the error and code are now *correlated*, which requires a much more refined analysis.

Complexity of Learning Stabilizers with Noise. What is the computational complexity of solving the LSN problem? Notice that the description of the learning task features quantum inputs, which means that its complexity cannot be characterized by

traditional complexity classes such as BQP or QMA which deal with classical-input decision problems. Instead, we show that the LSN problem lies in a (distributional and oracle) unitary synthesis class called $\mathsf{avgUnitaryBQP}^{\mathsf{avgUnitarySZK}_{\mathrm{HV}}}$ which was recently introduced by Bostanci et al. [BEM+23]. Our result can be seen as a quantum analog of the classical result which states that LPN is contained in $\mathsf{SearchBPP}^{\mathsf{SZK}}$ [BLVW18].

We now sketch the main idea behind our complexity upper bound. Suppose we are given as input an LSN instance of the form

$$\left(S \in \mathsf{Stab}(n,k),\ E\,|\overline{\psi_x}\rangle^S\right),$$

where $E \sim \mathcal{D}_p^{\otimes n}$ is a random Pauli error from a local depolarizing channel and where $x \sim \{0,1\}^k$ is a random string. Suppose also that the the underlying stabilizer code is non-degenerate and has distance at least $3np+1$ which, as we argued earlier, happens with overwhelming probability. Suppose that $C \in \mathsf{Cliff}_n$ is an encoding Clifford with corresponds to the stabilizer code described by S. Note that such a Clifford can always be found efficiently given S, as we show in Theorem 3.7.

We can represent the density matrix corresponding to the quantum part of the instance[8] as the result of discarding register A of the purification,

$$|Q^0\rangle_{\mathsf{AB}} = \sqrt{2^{-k}} \sum_x \sum_{E_a} \sqrt{\Pr_{E_a \sim \mathcal{D}_p^{\otimes n}}[E_a]} \left(|x\rangle \otimes |a\rangle\right)_{\mathsf{A}} \otimes \left(E_a\, C(|0^{n-k}\rangle \otimes |x\rangle) \otimes |0\rangle\right)_{\mathsf{B}}. \quad (3)$$

In addition, we also consider the following bipartite state given by

$$|Q^1\rangle_{\mathsf{AB}} = \sqrt{2^{-k}} \sum_x \sum_{E_a} \sqrt{\Pr_{E_a \sim \mathcal{D}_p^{\otimes n}}[E_a]} \left(|x\rangle \otimes |a\rangle\right)_{\mathsf{A}} \otimes \left(|0^{n-k}\rangle \otimes |x\rangle \otimes |a\rangle\right)_{\mathsf{B}}. \quad (4)$$

We now observe that a local transformation which acts solely on the B register and maps $|Q^0\rangle$ to $|Q^1\rangle$ would suffice to solve the LSN decoding task. Indeed, we show that such a transformation always exists by appealing to Uhlmann's theorem. First, we invoke the non-degeneracy of the stabilizer code described by S to argue that the fidelity between the reduced states Q^0_{A} and Q^1_{A} on register A is near maximal—provided that n is slightly larger than k, and that $p \in (0, 1/2)$ is a sufficiently small constant. Therefore, by Uhlmann's theorem (see Theorem 3.2), there exists a unitary U which acts on the B register and maps $|Q^0\rangle_{\mathsf{AB}}$ to another state which has near maximal overlap with $|Q^1\rangle_{\mathsf{AB}}$. In other words, to solve LSN, it suffices to synthesize the Uhlmann unitary on register B. Using recent results from Bostanci et al. [BEM+23], we can show that the task of synthesizing such a unitary falls within the distributional and oracle unitary synthesis class given by $\mathsf{avgUnitaryBQP}^{\mathsf{avgUnitarySZK}_{\mathrm{HV}}}$. This yields the desired complexity upper bound for solving the LSN problem.

2.2 Applications.

Learning from quantum data. Just as LPN has been fundamental to lower bounds in classical learning theory, we expect that LSN will be a useful tool for proving lower bounds in quantum learning theory. In Section 9.1 we identify one such learning setting: learning from quantum data [Car21, CL21, FQR24, CGR+24], a generalization of Probably Approximately Correct (PAC) learning to the quantum setting. Here, the goal is to learn a map $\rho : \mathcal{X} \to L(\mathcal{H}_d)$ from classical to quantum data – for example, a Hamiltonian can be construed as a map from temperatures to Gibbs states, or time-evolved states.

[8] We also append an ancilla register in the state $|0\rangle$ for convenience.

In one special case of this task known as *learning state preparation processes*, the learner is allowed to observe input-output pairs for an unknown map, but the inputs are sampled from a distribution and are not identical. Refs. [CL21, FQR24] gave sample-efficient algorithms for learning in this setting, thus showing that it is information theoretically possible. But we show that these algorithms can never be computationally efficient—assuming the hardness of our LSN assumption. While the lack of computational efficiency was implicit for the fully general learning setting due to a result of [AGS21], that result does not apply when there are physically natural restrictions on the concept class, such as learning processes that involve quantum noise. Our LSN hardness assumption fills this gap.

As a bonus, we are also able to *upper* bound the complexity of learning state preparation processes, by relating it to a unitary synthesis problem. We are to our knowledge the first to identify a complexity class that contains such learning tasks.

Constructing quantum bit commitment schemes. In Section 9.2, we give a cryptographic application of LSN and show how to construct a statistically hiding and computationally binding quantum commitment scheme [Yan22]. This is a fundamental cryptographic primitive that allows two parties (called a *sender* and *receiver*) to engage in a two-phase quantum communication protocol: in the first phase (the "commit phase"), the sender sends a commitment (i.e., a quantum register) to a bit b to the receiver; the *hiding* property of a bit commitment scheme ensures that the receiver cannot decide the value of b from the commitment alone. In the second phase (the "reveal phase"), the sender sends another quantum register to the receiver that allows the receiver to compute the value of b; the *binding* property of commitments ensures that the sender can only reveal the correct value of b, i.e. if the sender sent a reveal register that was meant to convince the receiver it had committed to a different value of b, the receiver would detect this.

2.3 Related Work

Random linear codes have been extensively studied in the field of coding theory [Gil52, Var64]. The *Learning Parity with Noise* problem was first proposed in [BFKL94]. Blum, Kalai and Wasserman [BKW03] gave an algorithm that solves LPN in time $O(2^{k/\log k})$. Berlekamp, McEliece and van Tilborg [BMT78] showed that the worst-case (decisional) syndrome decoding task is NP-complete. Brakerski, Lyubashevsky, Vaikuntanathan and Wichs [BLVW18] gave a worst-case to average-case reduction and showed that LPN is at least as hard as (a mildly hard variant of) of the nearest codeword problem (NCP). In subsequent work, Ref. [YZ21] later gave an improved worst-case to average-case reduction in the subexponentially-hard constant-noise regime. [BLVW18] also showed that LPN is contained in $\mathsf{SearchBPP}^{\mathsf{SZK}}$, and thus unlikely to be NP-hard.

Smith [Smi06] showed a quantum analog of the Gilbert-Varshamov bound using the notion of random *stabilizer codes*. The worst-case hardness of decoding quantum stabilizer codes as a classical decoding task has been extensively studied, and was found to be NP-complete [HL11, KL12] or #P-complete [IP15]—depending on the problem. The key insight in these results is that classical decoding essentially reduces to quantum decoding. Ref.s [HL11] and [KL12] use a one-to-one correspondence between stabilizer codes and classical linear codes to prove that quantum maximum-likelihood decoding is NP-complete. Ref.s [PC08, IP15] go one step further, pointing out that decoding stabilizer codes should be even harder than decoding classical codes because of *error de-*

generacy in the quantum setting, whereby multiple different errors can lead to the same syndrome. Based on this insight, [IP15] showed that quantum maximum-likelihood decoding (which accounts for error degeneracy is in fact #P-complete). The proof, once again, reduces from a classical problem: evaluating the weight-enumerator polynomial of a classical binary linear code. Ref [KK23] also show that finding the minimum distance of a quantum code is NP-hard due to a reduction from classical minimum distance decoding.

The fact that all aforementioned results about quantum decoding rely on classical complexity primitives underscores the need for a formalism that captures the inherent quantum nature of the decoding task. A recent work of Bostanci, Efron, Metger, Poremba, Qian and Yuen [BEM+23] characterized the complexity of decoding general quantum channels (which includes quantum error correction) using the language of of *unitary synthesis* problems. Crucially, this implies to uniquely quantum problems that feature quantum inputs and outputs. While their results do not explicitly analyze random stabilizer codes, we make use of their formalism in order to describe the complexity of our average-case LSN problem.

Recent work of Grewal, Iyer, Kretschmer and Liang [GIKL24a, GIKL24b] studied the sample complexity of learning general stabilizer states, and even more general states that feature few T-gates. However, their setting is, in some sense, orthogonal to ours. For example, in our (single-shot) LSN learning task, the description of the encoding Clifford is entirely public, and the apparent hardness of learning arises in the presence of noise, whereas in their setting the task is to determine the set of stabilizers (and encoding Clifford) from several identical (pristine) copies of the unknown state.

Gollakota and Liang [GL22] gave lower bounds on the sample complexity of PAC-learning noisy stabilizer states in the Statistical Query (SQ) model. While they do connect the hardness of their learning task to LPN via a reduction, it does not consider the problem of learning random stabilizers as in our setting, and therefore does not resemble our average-case learning task in any meaningful way.

2.4 Open Problems

Our work raises a number of interesting open questions; in particular:

- Can we place a special variant of LSN in MicroCrypt [MY22], whereby the hardness of the problem plausibly does not imply the existence of one-way functions?[9]
- Can we reduce the LPN problem to the standard LSN problem, where the underlying stabilizer code is uniformly random?
- What is the largest n-qubit Clifford subgroup that takes low-weight Paulis to low-weight Paulis? Note that this could potentially allow one to obtain a stronger worst-case to average-case reduction for LSN.
- Can we prove a much better worst-case to average-case reduction altogether which applies to the standard LSN problem for random stabilizer codes?
- Can we prove a search-to-decision reduction for LSN, similar to what is known for both the LPN problem [KSS10] and the LWE problem [Reg09]? This could enable a number of other cryptographic primitives, such as (succinct) quantum encryption, directly under the LSN assumption.

[9] Interestingly, the LSN assumption as currently defined is not a MicroCrypt assumption, as the stabilizer syndrome decoding problem can always be solved with the help of an NP oracle due to the non-degeneracy of random stabilizer codes. We thank Kabir Tomer and Justin Raizes for this observation.

Acknowledgements

This material is based upon work supported by the Department of Energy, Office of Science, National Quantum Information Science Research Centers, Quantum Systems Accelerator, under Grant number DOE DE-SC0012704. The authors would like to thank Alexandru Gheorghiu, Anand Natarajan, Aram Harrow, Henry Yuen, John Bostanci, Jonas Haferkamp, Jonas Helsen, Jonathan Conrad, Soonwon Choi, Thomas Vidick and Vinod Vaikuntanathan for useful discussions. The authors are supported by the National Science Foundation (NSF) under Grant No. CCF-1729369.

3 Preliminaries

Let us first introduce some basic notation and relevant background.

Notation. For $N \in \mathbb{N}$, we use $[N] = \{1, 2, \ldots, N\}$ to denote the set of integers up to N. The symmetric group on $[N]$ is denoted by $\mathfrak{S}_N$. In slight abuse of notation, we sometimes identify elements $x \in [N]$ with bit strings $x \in \{0,1\}^n$ via their binary representation whenever $N = 2^n$ and $n \in \mathbb{N}$. Similarly, we identify permutations $\pi \in \mathfrak{S}_N$ with permutations $\pi : \{0,1\}^n \to \{0,1\}^n$ over bit strings of length n.

We write $\mathsf{negl}(\cdot)$ to denote any *negligible* function, which is a function f such that, for every constant $c \in \mathbb{N}$, there exists an integer N such that for all $n > N$, $f(n) < n^{-c}$.

Probability theory. The notation $x \sim X$ describes that an element x is drawn uniformly at random from the set X, and we use $\mathrm{Unif}(X)$ to denote the uniform distribution over X. Similarly, if $\mathcal{D}$ is a general distribution, we let $x \sim \mathcal{D}$ denote sampling x according to $\mathcal{D}$. We denote the expectation value of a random variable X by $\mathbb{E}[X] = \sum_x x \Pr[X = x]$. If $\mathcal{D}$ is a distribution over a set X, then we denote by $\mathcal{D}^{\otimes n}$ the n-wise product distribution over the set $X \times \cdots \times X$. For a parameter $p \in [0,1]$, we let Ber_p denote the Bernoulli distribution with

$$\Pr[X = 1] = p \quad \text{and} \quad \Pr[X = 0] = 1 - p, \quad \text{for } X \sim \mathsf{Ber}_p.$$

We let $\mathsf{Bin}_{n,p}$ denote the Binomial distribution with $\Pr[X = k] = \binom{n}{k} p^k (1-p)^{n-k}$, for a random variable $X \sim \mathsf{Bin}_{n,p}$.

Quantum information. For a comprehensive background on quantum computation, we refer to [NC00]. We denote a finite-dimensional complex Hilbert space by $\mathcal{H}$, and we use subscripts to distinguish between different systems (or registers). For example, we let $\mathcal{H}_A$ be the Hilbert space corresponding to a system A. The tensor product of two Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$ is another Hilbert space denoted by $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$. The Euclidean norm of a vector $|\psi\rangle \in \mathcal{H}$ over the finite-dimensional complex Hilbert space $\mathcal{H}$ is denoted as $\|\psi\| = \sqrt{\langle\psi|\psi\rangle}$. Let $\mathrm{L}(\mathcal{H})$ denote the set of linear operators over $\mathcal{H}$. A quantum system over the 2-dimensional Hilbert space $\mathcal{H} = \mathbb{C}^2$ is called a *qubit*. For $n \in \mathbb{N}$, we refer to quantum registers over the Hilbert space $\mathcal{H} = \left(\mathbb{C}^2\right)^{\otimes n}$ as n-qubit states. We use the word *quantum state* to refer to both pure states (unit vectors $|\psi\rangle \in \mathcal{H}$) and density matrices $\rho \in \mathcal{D}(\mathcal{H})$, where we use the notation $\mathcal{D}(\mathcal{H})$ to refer to the space of positive semidefinite matrices of unit trace acting on $\mathcal{H}$.

A quantum channel $\Phi : \mathrm{L}(\mathcal{H}_A) \to \mathrm{L}(\mathcal{H}_B)$ is a linear map between linear operators over the Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$. Oftentimes, we use the compact notation $\Phi_{A \to B}$ to

denote a quantum channel between $\mathrm{L}(\mathcal{H}_A)$ and $\mathrm{L}(\mathcal{H}_B)$. We say that a channel Φ is *completely positive* if, for a reference system R of arbitrary size, the induced map $\mathbb{I}_R \otimes \Phi$ is positive, and we call it *trace-preserving* if $\mathrm{Tr}\{\Phi(X)\} = \mathrm{Tr}\{X\}$, for all $X \in \mathrm{L}(\mathcal{H})$. A quantum channel that is both completely positive and trace-preserving is called a quantum CPTP channel. A *unitary* $U : \mathrm{L}(\mathcal{H}_A) \to \mathrm{L}(\mathcal{H}_A)$ is a special case of a quantum channel that satisfies $U^\dagger U = UU^\dagger = \mathbb{I}_A$. We denote the n-qubit unitary group by $\mathcal{U}_n$. A *projector* Π is a Hermitian operator such that $\Pi^2 = \Pi$, and a *projective measurement* is a collection of projectors $\{\Pi_i\}_i$ such that $\sum_i \Pi_i = \mathbb{I}$. A positive-operator valued measure (POVM) is a set of Hermitian positive semidefinite operators $\{M_i\}$ acting on a Hilbert space $\mathcal{H}$ such that $\sum_i M_i = \mathbb{I}$. A linear map $U \in \mathrm{L}(\mathcal{H}_\mathsf{A}, \mathcal{H}_\mathsf{B})$ is called a partial isometry if there exists a projector $\Pi \in \mathrm{L}(\mathcal{H}_\mathsf{A})$ and an isometry $\tilde{U} \in \mathrm{L}(\mathcal{H}_\mathsf{A}, \mathcal{H}_\mathsf{B})$ such that $U = \tilde{U}\Pi$. We call the image of the projector Π the *support* of the partial isometry U. Because a partial isometry cannot be implemented in practice (it is not a trace-preserving operation), we also define a *channel completion* of a partial isometry as any quantum channel that behaves like the partial isometry on its support, and can behave arbitrarily on the orthogonal complement of the support; specifically, for any partial isometry $U \in \mathrm{L}(\mathcal{H}_\mathsf{A}, \mathcal{H}_\mathsf{B})$, a *channel completion of* U is a CPTP channel $\Phi \in \mathrm{L}(\mathcal{H}_\mathsf{A}, \mathcal{H}_\mathsf{B})$ such that

$$\Phi(\Pi\rho\Pi) = U\Pi\rho\Pi U^\dagger, \quad \text{for } \rho \in \mathcal{D}(\mathcal{H}_\mathsf{A}),$$

where $\Pi \in \mathrm{L}(\mathcal{H}_\mathsf{A})$ is the projector onto the support of U. If Φ is a unitary or isometric channel, we also call this a *unitary* or *isometric completion* of the partial isometry.

Quantum distance measures. Let $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ be two density matrices acting on the same Hilbert space $\mathcal{H}$. The (squared) *fidelity* between ρ and σ is defined as

$$\mathrm{F}(\rho, \sigma) = \|\sqrt{\rho}\sqrt{\sigma}\|_1^2,$$

where $\|\cdot\|_1$ is the trace norm. The *trace distance* of $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ is given by

$$\delta_{\mathsf{TD}}(\rho, \sigma) = \frac{1}{2}\|\rho - \sigma\|_1.$$

The two distance measures are related via the Fuchs-van de Graaf inequalities:

$$1 - \sqrt{\mathrm{F}(\rho, \sigma)} \leq \delta_{\mathsf{TD}}(\rho, \sigma) \leq \sqrt{1 - \mathrm{F}(\rho, \sigma)}.$$

We also use the following inequality.

Lemma 3.1 (Strong convexity of trace distance ([NC00], Theorem 9.3))**.** *Let* $\mathbf{p} = \{p_i\}$ *and* $\mathbf{q} = \{q_i\}$ *be probability distributions over the same index set, and let* $\{\rho_i\}$ *and* $\{\sigma_i\}$ *be density operators, also with indices from the same index set. Then,*

$$\delta_{\mathsf{TD}}\left(\sum_i p_i\rho_i, \sum_i q_i\sigma_i\right) \leq \sum_i p_i \cdot \delta_{\mathsf{TD}}\left(\rho_i, \sigma_i\right) + \delta_{\mathsf{TV}}(\mathbf{p}, \mathbf{q}).$$

Uhlmann's theorem. We frequently make use of the following theorem.

Theorem 3.2 (Uhlmann's theorem [Uhl76])**.** *Let* $|\psi\rangle_{\mathsf{AB}}$ *and* $|\phi\rangle_{\mathsf{AB}}$ *be pure states that live in a Hilbert space* $\mathcal{H}_{\mathsf{AB}}$*, and let* ρ_A *and* σ_A *denote their respective reduced states in register* A*. Then, there exists a unitary* $U \in \mathrm{L}(\mathcal{H}_\mathsf{B})$ *acting only on register* B *such that*

$$\mathrm{F}(\rho_\mathsf{A}, \sigma_\mathsf{A}) = |\langle\phi|_{\mathsf{AB}}\left(\mathbb{I}_\mathsf{A} \otimes U_\mathsf{B}\right)|\psi\rangle_{\mathsf{AB}}|^2.$$

Gentle Measurement. We also make use of the following well-known lemma, which is often called the Gentle Measurement Lemma.

Lemma 3.3 ([Wil13], Lemma 9.4.1). *Let $\rho \in \mathcal{D}(\mathcal{H})$ be an arbitrary density matrix, and let Λ be any positive semidefinite hermitian matrix. Then,*

$$\delta_{\mathsf{TD}}\left(\rho, \frac{\sqrt{\Lambda}\rho\sqrt{\Lambda}}{\mathrm{Tr}[\Lambda\rho]}\right) \leq \sqrt{1 - \mathrm{Tr}[\Lambda\rho]}.$$

Permutation operators. Let $n \in \mathbb{N}$ be an integer. Then, for a permutation $\pi \in \mathfrak{S}_n$, we define the corresponding n-qubit permutation operator $\mathcal{Q}(\pi)$ over $(\mathbb{C}^2)^{\otimes n}$ as

$$\mathcal{Q}(\pi) := \sum_{i_1,\ldots,i_n \in \{0,1\}} \left|i_{\pi^{-1}(1)}, \ldots, i_{\pi^{-1}(n)}\right\rangle \langle i_1, \cdots, i_n| .$$

In other words, $\mathcal{Q}(\pi)$ is the unitary operator that permutes all of the single-qubit qubit registers according to the permutation π. By linearity, the operator $\mathcal{Q}(\pi)$ also permutes any product of single-qubit linear operators $\mathbf{O}_1, \ldots, \mathbf{O}_n \in \mathrm{L}(\mathbb{C}^2)$ as follows:

$$\mathcal{Q}(\pi)(\mathbf{O}_1 \otimes \cdots \otimes \mathbf{O}_n)\mathcal{Q}(\pi)^\dagger = (\mathbf{O}_{\pi^{-1}(1)} \otimes \cdots \otimes \mathbf{O}_{\pi^{-1}(n)}).$$

3.1 Pauli group.

The n-qubit Pauli group $\mathcal{P}_n$ consists of n-fold tensor products of Pauli operators. In other words, this is a group of order $|\mathcal{P}_n| = 2^{2n+1}$ with

$$\mathcal{P}_n = \{\pm\mathbb{I}, \pm\mathsf{X}, \pm\mathsf{Y}, \pm\mathsf{Z}\}^{\otimes n}.$$

Once we switch to the symplectic representation, we are going to ignore signs and instead consider the quotient group given by

$$\bar{\mathcal{P}}_n = \mathcal{P}_n / \{\pm\mathbb{I}^{\otimes n}\}.$$

An important property of Pauli group is the fact that any two Pauli operators either commute or anti-commute, i.e., it holds that $PQ = \pm QP$ for any Pauli operators P, Q.

Symplectic representation. Instead of working with $\mathcal{P}_n$, we sometimes use the quotient group $\bar{\mathcal{P}}_n = \mathcal{P}_n / \{\pm\mathbb{I}^{\otimes n}\}$ to reason about Paulis. In this case, every unsigned Pauli $P \in \bar{\mathcal{P}}_n$ can be specified in terms of a pair $\mathbf{p}_x, \mathbf{p}_z \in \{0,1\}^n$ such that

$$P = \bigotimes_{i=1}^{n} \mathsf{X}^{\mathbf{p}_{x,i}} \cdot \bigotimes_{i=1}^{n} \mathsf{Z}^{\mathbf{p}_{z,i}}.$$

Therefore, we can directly identify the unsigned Paulis $\bar{\mathcal{P}}_n$ with a binary vector space of dimension $2n$ such that $\bar{\mathcal{P}}_n \cong \mathbb{Z}_2^{2n}$. Note that multplication of Paulis P, Q in $\bar{\mathcal{P}}_n$, each represented by $\mathbf{p} = (\mathbf{p}_x|\mathbf{p}_z)$ and $\mathbf{q} = (\mathbf{q}_x|\mathbf{q}_z)$, amounts to addition in $\mathbb{Z}_2^{2n}$:

$$(\mathbf{p}_x|\mathbf{p}_z) \cdot (\mathbf{q}_x|\mathbf{q}_z) = (-1)^{\mathbf{p}_x \cdot \mathbf{q}_z + \mathbf{p}_z \cdot \mathbf{q}_x}(\mathbf{p}_x \oplus \mathbf{q}_x|\mathbf{p}_z \oplus \mathbf{p}_x).$$

We call $(\mathbf{p}_x|\mathbf{p}_z) \odot (\mathbf{q}_x|\mathbf{q}_z) := \mathbf{p}_x \cdot \mathbf{q}_z + \mathbf{p}_z \cdot \mathbf{q}_x$ the symplectic inner product. Two Paulis P, Q commute if and only if their symplectic inner product of its $\mathbb{Z}_2^{2n}$ representations $(\mathbf{p}_x|\mathbf{p}_z)$ and $(\mathbf{q}_x|\mathbf{q}_z)$ vanishes[10]. In other words, if and only if $\mathbf{p} \odot \mathbf{q} = 0 \pmod 2$.

[10]The symplectic inner product of $P, Q \in \bar{\mathcal{P}}_n$ is identical to the function $c(P, Q)$ we defined earlier.

3.2 Clifford group.

The Clifford group is the set of unitaries that normalizes the Pauli group; that is

$$\mathsf{Cliff}_n = \left\{ U \in \mathcal{U}_n \mid UPU^\dagger \in \mathcal{P}_n, \forall P \in \mathcal{P}_n \right\} \tag{5}$$

The Clifford group also contains operators of the form $e^{i\theta} I$. for some $\theta \in [0, 2\pi]$. These global phase operators are often irrelevant to us because Clifford operators act by conjugation, and thus we consider two Clifford operators to be equivalent if they differ only by a global phase. Thus we will define

$$\check{\mathsf{C}}\mathsf{liff}_n = \mathsf{Cliff}_n / \{e^{i\theta}\mathcal{P}_n\}. \tag{6}$$

We also quote a result about the generators of the Clifford group:

Theorem 3.4 (Clifford generators [Got24]). *Any gate in the Clifford group* Cliff_n *can be written as a product of* $e^{i\theta} I$, H_i, $R_{\pi/4,i}$, *and* $\mathrm{CNOT}_{i,j}$, *with* $i, j = 1, \ldots, n$ *and* $\theta \in [0, 2\pi)$.

In the rest of this paper, we assume all Clifford circuits are written in terms of the above gates. In fact, we can compile the basic set of Clifford generators above into a more convenient gateset:

$$\{e^{i\theta} I, H, R_{\pi/4}, \mathrm{CNOT}, \mathrm{SWAP}, \text{C-Z}.\} \tag{7}$$

This is seen to be equivalent to the original gateset by using the identities $\text{C-Z} = (I \otimes H)\,\mathrm{CNOT}(I \otimes H)$ and $\mathrm{SWAP}_{i,j} = \mathrm{CNOT}_{i\to j} \otimes \mathrm{CNOT}_{j\to i} \otimes \mathrm{CNOT}_{i\to j}$, pictorially illustrated in Figure 1. Here we have, for the only time in this paper, used the notation $\mathrm{CNOT}_{i\to j}$ to denote a CNOT with i as the control qubit and j as the target qubit.

Permutations and Local Cliffords. Note that, for any $\pi \in \mathfrak{S}_n$, the permutation operator $\mathcal{Q}(\pi)$ can be implemented as a Clifford operation by composing transpositions. These are themselves Clifford operations consisting of three consecutive CNOTs, as illustrated in Figure 1. This motivates us to consider the n-qubit subgroup $\mathsf{PLC}_n \leq \check{\mathsf{C}}\mathsf{liff}_n$

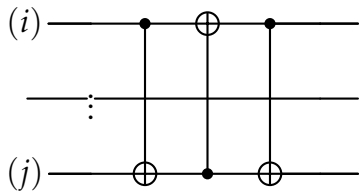


Figure 1: Circuit to transpose qubit i and j

generated by permutation operators and local Clifford gates:

$$\mathsf{PLC}_n := \left\{ C \in \check{\mathsf{C}}\mathsf{liff}_n \mid C = \bigotimes_{i=1}^{n} C_i \circ \mathcal{Q}(\pi) \;:\; C_1, \ldots, C_n \in \check{\mathsf{C}}\mathsf{liff}_1 \;\text{ and }\; \pi \in \mathfrak{S}_n \right\}.$$

It is easy to see that PLC_n forms a group, as captured by the following lemma.

Lemma 3.5 (PLC is a group). *The set of all Clifford unitaries in* PLC_n *form a subgroup of the* n*-qubit unitary group under unitary composition (i.e., matrix multiplication).*

This group is a key tool in our worst-to average-case reduction in Section 7.

3.3 Quantum Noise Channels

Single-qubit depolarizing noise. For $M \in L(\mathbb{C}^d)$, we define the single-qubit depolarizing noise channel $\mathcal{D}_p : L(\mathbb{C}^d) \to L(\mathbb{C}^d)$ as one that acts as

$$\mathcal{D}_p(M) := (1 - 4/3p)M + 4/3p\mathrm{Tr}\{M\}\frac{\mathbb{I}}{2}, \quad p \in [0, 3/4]. \tag{8}$$

This channel, when acting on a quantum state ρ, has the Kraus representation

$$\mathcal{D}_p(\rho) = \frac{p}{3}X\rho X + \frac{p}{3}Y\rho Y + \frac{p}{3}Z\rho Z + (1-p)\rho, \tag{9}$$

Thus, a single-qubit depolarizing channel can be thought of as a random Pauli channel $E(\cdot)E^\dagger$, $E \in \mathcal{P}_1$, where E can be sampled as follows: pick $wt(E) \sim \mathsf{Bernoulli}(p)$ (i.e. apply $\mathrm{id}(\cdot)$ with probability $1-p$) then uniformly sample from the $E \in \mathcal{P}_1$ with weight w.

Tensor product of single-qubit depolarizing noise channels. We will model noise acting on an n-qubit quantum state as a tensor product of n single-qubit depolarizing noise channels with parameter $p \in (0, 3/4)$, where for $\rho \in L(\mathbb{C}^{2^n})$

$$\mathcal{D}_p^{\otimes n}(\rho) := \sum_{E \in \mathcal{P}_n} \left(\frac{p}{3}\right)^{|E|} (1-p)^{n-|E|} E\rho E^\dagger =: \sum_{E \in \mathcal{P}_n} \Pr_{E \sim \mathcal{D}_p^{\otimes n}}[E] E\rho E^\dagger. \tag{10}$$

It is not hard to check that Equation (10) yields Equation (9) for $n = 1$. This channel is equivalent to acting with a random Pauli channel $E(\cdot)E^\dagger$, $E \in \mathcal{P}_n$, where E can be sampled as follows: pick $wt(E) = w \sim \mathsf{Binom}(n, p)$, then uniformly sample from the $E \in \mathcal{P}_n$ with weight w. We will often use this interpretation of noise channels as probabilistically applying Pauli errors.

Bit-flip noise channels. For $\rho \in L(\mathbb{C}^2)$, the single-qubit bit-flip noise channel $\mathcal{F}_p : L(\mathbb{C}^2) \to L(\mathbb{C}^2)$ acts as

$$\mathcal{F}_p(M) := (1-p)\rho + pX\rho X. \tag{11}$$

This may, as usual, be extended to a tensor product of n single-qubit bit-flip noise channels via:

$$\mathcal{F}_p^{\otimes n}(\rho) := \sum_{b \in \{0,1\}^n} p^{|b|}(1-p)^{n-|b|} X^b \rho X^b, \tag{12}$$

where for $i \in [n]$, the i-th qubit of X^b is $(X^b)_i := X^{\mathbb{I}(b_i=1)}$.

3.4 Stabilizer Codes.

We can use $\mathcal{P}_n$ to characterize a quantum error correcting code as follows. Let $S \leq \mathcal{P}_n$ be an abelian subgroup which has $n-k$ generators. In other words,

$$S = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \rangle.$$

Then, we define the codespace $C(S) \subseteq (\mathbb{C}^2)^{\otimes n}$ as the simultaneous $(+1)$ eigenspace of all elements of S. [11]. In other words,

$$C(S) = \left\{ |\psi\rangle \in \left(\mathbb{C}^2\right)^{\otimes n} : M|\psi\rangle = (+1)|\psi\rangle, \ \forall M \in S \right\}.$$

[11] Because $S \subseteq \mathcal{P}_n$ is an abelian subgroup its elements can all be simultaneously diagonalized.

If S has $n-k$ generators, then we know that $C(S)$ has precisely 2^k many codewords. We also use $\mathrm{Stab}(n,k)$ to denote the set of all stabilizer subgroups of $\mathcal{P}_n$ with $n-k$ generators.

The normalizer, denoted by $N(S)$, is the set of Paulis that commute with S[12]:

$$N(S) = \{P \in \mathcal{P}_n \,:\, PQ = QP,\ \forall Q \in S\}.$$

How do we detect errors? Suppose we have a class of errors $\mathcal{E} = \{E_a\}$ where each $E_a \in \mathcal{P}_n$ is a low-weight Pauli error. Let $|\psi\rangle \in C(S)$ be a codeword. Then, for every generator $\mathbf{g}_i$ in S, one of two possible events takes place:

- $\mathbf{g}_i$ commutes with E_a, in which case

$$\mathbf{g}_i E_a |\psi\rangle = E_a \mathbf{g}_i |\psi\rangle = E_a |\psi\rangle .$$

- $\mathbf{g}_i$ anti-commutes with E_a, in which case

$$\mathbf{g}_i E_a |\psi\rangle = (-1) E_a \mathbf{g}_i |\psi\rangle = (-1) E_a |\psi\rangle .$$

We can detect this phase by performing a measurement. In general, we want to do this for every generator and collect a syndrome vector $s_a \in \{0,1\}^{n-k}$ such that

$$\mathbf{g}_i E_a = (-1)^{s_{a,i}} E_a \mathbf{g}_i , \quad \forall i \in [n-k].$$

The *distance* of a quantum code C is the minimum Hamming weight d of an error that the code cannot correct. We use the following well-known fact (see e.g. [Got24]).

Theorem 3.6 (Knill-Laflamme conditions)**.** *Let $\mathcal{E}$ be a set of errors with maximum Hamming weight d. Then, $C \subseteq (\mathbb{C}^2)^{\otimes n}$ is a $[[n,k,d]]$ quantum error correcting code with distance d if and only if for every $|\psi\rangle, |\phi\rangle \in C$ and for all $E_a, E_b \in \mathcal{E}$, it holds that*

$$\langle\psi| E_a^\dagger E_b |\phi\rangle = c_{ab} \langle\psi|\phi\rangle$$

for some hermitian matrix c_{ab}. Note that c_{ab} does not depend on $|\psi\rangle$ or $|\phi\rangle$.

We say that a code is *non-degenerate* if $c_{ab} = \delta_{ab}$. The interpretation of the non-degeneracy condition is that two different errors take any two states in the same subspace to orthogonal subspaces.

Encoding circuits for stabilizers. Given a description of some stabilizer code in terms of the generating elements of the stabilizer group $S = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \rangle$, there exists a Clifford circuit $U_S \in \mathsf{Cliff}_n$ to encode any initial k-qubit state $|\psi\rangle \in (\mathbb{C}^2)^{\otimes k}$ as a state inside the subspace stabilized by S. Let

$$|\bar{\psi}\rangle^S := U_{\mathrm{Enc}}^S \left(|0^{n-k}\rangle \otimes |\psi\rangle \right) \in C(S).$$

denote the encoding of $|\psi\rangle$ in the subspace stabilized by S.

How can we find U_{Enc}^S? Note that the initial state $|0^{n-k}\rangle \otimes |\psi\rangle$ itself is also a codeword which is stabilized by the trivial stabilizer code $\langle Z_1, \ldots, Z_{n-k} \rangle$, i.e.,

$$\begin{array}{cccccccc} Z & I & I & \cdots & I & I & \cdots & I \\ I & Z & I & \cdots & I & I & \cdots & I \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ I & I & I & \cdots & Z & I & \cdots & I \end{array}$$

[12] Technically, this is the centralizer but in our case they are the same.

Any encoding unitary U_{Enc}^S also acts by conjugation to map the initial set of stabilizers Z_i to the new set of stabilizers specified by S, i.e. $\mathbf{g}_i = U_{\text{Enc}}^S Z_i U_{\text{Enc}}^{S\,\dagger}$. This is because for every $i \in [n-k]$, we have

$$\begin{aligned} U_{\text{Enc}}^S \left(|0^{n-k}\rangle \otimes |\psi\rangle \right) &= U_{\text{Enc}}^S \, Z_i \left(|0^{n-k}\rangle \otimes |\psi\rangle \right) \\ &= \underbrace{\left(U_{\text{Enc}}^S Z_i U_{\text{Enc}}^{S\,\dagger} \right)}_{=\mathbf{g}_i} \underbrace{U_{\text{Enc}}^S \left(|0^{n-k}\rangle \otimes |\psi\rangle \right)}_{=|\bar{\psi}\rangle^S}. \end{aligned}$$

Theorem 3.7 (Efficient and efficiently-findable encoding circuits for stabilizer codes)**.** *Given a description of $S \in \mathsf{Stab}(n,k)$ there is an $O(n^3)$ time classical algorithm to write down an encoding circuit for S with $O(n^2)$ Clifford gates.*

Proof. The proof is presented in Gottesman (Section 6.4.1 of [Got24]) but we write it concisely. Because of the remark before this theorem, it suffices to specify a circuit $\mathcal{C}$ that maps the final generators $\mathbf{g}_1, \ldots, \mathbf{g}_{n-k}$ to the initial generators $Z_1, \ldots Z_{n-k}$, in the sense that $\mathcal{C}\mathbf{g}_i\mathcal{C}^\dagger = Z_i$. Then we may output $U_{\text{Enc}}^S = \mathcal{C}^\dagger$. To find $\mathcal{C}$, we start by writing the generators $\mathbf{g}_1, \ldots, \mathbf{g}_{n-k}$ in symplectic notation, representing them as two matrices $A, C \in \mathbb{Z}_2^{n \times n-k}$ where the i-th column of A is $(\mathbf{g}_i)_x$ and the i-th column of C is $(\mathbf{g}_i)_z$. To find $\mathcal{C}$, we make use of the fact that conjugating a Pauli $P \in \bar{\mathcal{P}}_n$ with a Clifford $\mathcal{C}$ with symplectic representation $M_{\mathcal{C}}$ transforms the symplectic representation of P (represented as a column vector) by left multiplication, as

$$(p_x, p_z)^T \to M_{\mathcal{C}} (p_x, p_z)^T. \tag{13}$$

Working exclusively within the symplectic picture, the goal is then to find a matrix $M_{\mathcal{C}}$ representing a valid Clifford such that

$$M_{\mathcal{C}} \left(\frac{A}{C} \right) = \left(\frac{0}{\mathbb{I}_{n,n-k}} \right) \tag{14}$$

where $\mathbb{I}_{n,n-k}$ denotes a n by $n-k$ matrix with $\mathbb{I}_{n-k}$ in its first $n-k$ rows and 0s in all other locations. One can verify that the right-hand-side of the above equation represents the stabilizer $\langle Z_1, \ldots Z_{n-k} \rangle$ and hence the target.

By checking Equations 6.79-6.82 of [Got24], which give the symplectic representations of each Clifford generator, one can verify that row and column reductions on $(\frac{A}{C})$ can be implemented in symplectic space by acting with Clifford gates on the actual generators. Specifically, column reduction on A composes the operation of adding the topmost row with a leading 1 to a different row which has an undesired leading 1. This corresponds to acting with a single CNOT. Similarly, column reduction on C can be implemented by acting with a single $R_{\pi/4}$ or $C-Z$. Row reduction on A (via adding the column with a leading 1 to other columns) corresponds to multiplication of generators, which effectively does not change the stabilizer group. Since only $O(n^2)$ additions of rows are needed to reduce $(\frac{A}{C})$ to the desired form, and each addition corresponds to adding $O(1)$ gates to the circuit, the row-reduction portion of the algorithm can be carried out in $O(n^2)$ time.

There's one problem: in symplectic space phases do not exist, so the circuit $\mathcal{C}'$ resulting from the above procedure may not produce $S = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \rangle$ with the proper signs. So we need to perform some final corrections. We then need to compute which $\mathbf{g}_i$s have the wrong signs, which by the following Lemma 3.8 takes time $O(n^2)$ for each

g_i. For every $\mathbf{g}_i$ that has the wrong sign, we modify the circuit $\mathcal{C}'$ by acting with X_i at the start of the circuit. This correction operation does not change the circuit complexity asymptotically and increases the classical runtime to $O(n^3)$, though we did not optimize this. □

We also quote a well-known result on simulating Cliffords.

Lemma 3.8 (Simulating Clifford circuits)**.** *Given a description of a Clifford circuit $C \in \mathsf{Cliff}_n$ where $C = \prod_{i=1}^{m} U_i$ and each U_i is a 2-qubit gate from some generating set of $\check{\mathsf{C}}\mathsf{liff}_n$, for any Pauli $P \in \mathcal{P}_n$ we may compute $CPC^\dagger$ in time linear in $O(m+n)$.*

Proof. This is a special case of the Gottesman-Knill theorem [Got24], which says that Clifford circuits acting on an initial n-qubit stabilizer state, followed by a sequence of m Clifford group operations and Pauli measurements, can be efficiently classically simulated. The simulation algorithm is to keep track of how each operation transforms the stabilizer group of the initial state.

More concretely, in this case we have only one Pauli to keep track of, so we may compute $CPC^\dagger$ by computing the effect of each gate U_i in sequence. Since each gate acts on a constant number of qubits, the time needed for the simulation and to write down the final Pauli scales as $O(m+n)$. Lookup tables for the action of gates from popular generating gatesets can be found in, e.g. Table 6.1 of [Got24]. □

Random stabilizer codes. A random $[[n,k]]$ stabilizer code is a uniformly random choice of abelian subgroup $S \leq \mathcal{P}_n$ with $n-k$ generators. Note, here the Pauli signs are important and will determine what subspace is stabilized by S. Because the members of S must commute, choosing $n-k$ elements uniformly from $\mathcal{P}_n$ will not always give a valid stabilizer code. We claim that a random element of the Clifford group Cliff_n, acting on any initial choice of S, generates a uniformly random S and hence a uniformly random $[[n,k]]$ stabilizer code.

Theorem 3.9. *Let $n \in \mathbb{N}$ be an integer and let $S = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \rangle$ be any stabilizer with generators $\mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \in \mathcal{P}_n$. Then, the conjugated stabilizer code*

$$USU^\dagger = \langle U\mathbf{g}_1U^\dagger, \ldots, U\mathbf{g}_{n-k}U^\dagger \rangle, \quad \text{for } U \sim \mathsf{Cliff}_n,$$

yields a uniformly stabilizer in the set $\mathsf{Stab}(n,k)$.

Proof. First, we show that the Clifford group Cliff_n acts transitively on the set of stabilizers $\mathsf{Stab}(n,k)$. Let $S = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \rangle$ be an arbitrary stabilizer with $n-k$ generators. From [Got24], we know that there exists a Clifford operator $C \in \mathsf{Cliff}_n$ and a Pauli $P \in \bar{\mathcal{P}}_n$ such that the composition of the two operations maps S to the canonical stabilizer Z. In particular, we can let $V = PC$ such that

$$VSV^\dagger = \langle V\mathbf{g}_1V^\dagger, \ldots, V\mathbf{g}_{n-k}V^\dagger \rangle = \langle Z_1, \ldots, Z_{n-k} \rangle.$$

Likewise, from [Got24], we also know that once we have the canonical all-Z stabilizer $Z = \langle Z_1, \ldots, Z_{n-k} \rangle$, we can obtain any other stabilizer $S' = \langle \mathbf{g}'_1, \ldots, \mathbf{g}'_{n-k} \rangle$ via some other composition of operators $W = DQ$, where $D \in \mathsf{Cliff}_n$ and $Q \in \bar{\mathcal{P}}_n$, i.e.,

$$S' = \langle \mathbf{g}'_1, \ldots, \mathbf{g}'_{n-k} \rangle = \langle WZ_1W^\dagger, \ldots, WZ_{n-k}W^\dagger \rangle.$$

Therefore, for any pair of distinct stabilizers $S, S' \in \mathsf{Stab}(n,k)$ there exists an operator WV that maps S to S'. By using the fact that Cliffords are the normalizer of the Pauli

group, WV can be realized as a single Pauli operation followed by a single Clifford operation.

Finally, we show that the probability that a random Clifford applied to an arbitrary stabilizer $S = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \rangle$ yields any pair of distinct stabilizers S_1, S_2 with exactly the same probability. From before, there exists a Clifford $C \in \mathsf{Cliff}_n$ such that

$$\begin{aligned}\Pr_{U \sim \mathsf{Cliff}_n}[USU^\dagger = S_1] &= \Pr_{U \sim \mathsf{Cliff}_n}[(CU)S(CU)^\dagger = CS_1C^\dagger] \\ &= \Pr_{U \sim \mathsf{Cliff}_n}[(CU)S(CU)^\dagger = S_2] \\ &= \Pr_{U \sim \mathsf{Cliff}_n}[USU^\dagger = S_2].\end{aligned}$$

The last line follows from the fact that Cliff_n is a group, and hence the uniform distribution over Cliff_n is Clifford invariant. □

Quantum Gilbert-Varshamov bound. In a nutshell, the quantum Gilbert-Varshamov bound [Smi06, Got24] tells us that a random stabilizer code is both non-degenerate and has a good distance with high probability. This is captured by the following result.

Theorem 3.10 (Quantum Gilbert-Varshamov bound, [Smi06])**.** *Random stabilizer codes which are specified by a random stabilizer subgroup $S \sim \mathsf{Stab}(n,k)$ are non-degenerate and have distance d with probability at least $1 - d \cdot 2^{nH(d/n)} \cdot 3^d \cdot 2^{-n+k}$.*

The above statement is perhaps best viewed through the lens of the Knill-Laflamme error correction conditions (Theorem 3.6). Suppose that $S = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \rangle$ is a non-degenerate stabilizer code with distance $d = 2t + 1$. Define the set

$$\mathcal{E}^{(t)} = \{E_1^\dagger E_2 \in \mathcal{P}_n \,:\, |E_1|, |E_2| \leq t\}$$

which consists of weight-t products of Pauli errors. Then, for all pairs of codewords $|\overline{\psi_x}\rangle, |\overline{\psi_y}\rangle \in C(S)$ with $x \neq y$, and for all $E_a^\dagger E_b \in \mathcal{E}^{(t)}$, it holds that

$$\langle \overline{\psi_x}| E_a^\dagger E_b |\overline{\psi_y}\rangle = 0.$$

One way to see this is as follows. Suppose that $E_a^\dagger E_b \notin N(S)$, then there must exist a generator $\mathbf{g}_i$ in $S = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{n-k} \rangle$ which anti-commutes with $E_a^\dagger E_b$, and thus

$$\mathbf{g}_i E_a^\dagger E_b |\overline{\psi_x}\rangle^S = -E_a^\dagger E_b \mathbf{g}_i |\overline{\psi_x}\rangle^S = -E_a^\dagger E_b |\overline{\psi_x}\rangle^S. \tag{15}$$

Using that $\mathbf{g}_i^2 = I^{\otimes n}$ together with Equation (15), this implies that, for $x \neq y$,

$$\langle \overline{\psi_x}| E_a^\dagger E_b |\overline{\psi_y}\rangle = \langle \overline{\psi_x}| \mathbf{g}_i E_a^\dagger E_b \mathbf{g}_i |\overline{\psi_y}\rangle = -\langle \overline{\psi_x}| E_a^\dagger E_b |\overline{\psi_y}\rangle = 0. \tag{16}$$

4 The Learning Stabilizers with Noise problem

In this section, we formally define the *Learning Stabilizers with Noise* (LSN) problem as the natural quantum analog of the LPN problem. We begin with a set of definitions for the problem (and its variants), and then show that the problem is well-defined (i.e., it admits a unique solution) for appropriate choices of parameters.

4.1 Definition

We now provide a formal definition of our learning task.

Definition 4.1 (Learning Stabilizers with Noise problem)**.** *Let $k \in \mathbb{N}$ be the security parameter and let $n = \mathsf{poly}(k)$ be an integer. Let $p \in (0, 1/2)$ be a parameter. The Learning Stabilizers with Noise ($\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$) problem is to find $x \in \{0,1\}^k$ given as input a sample*

$$\left(S \in \mathsf{Stab}(n,k), E\,|\overline{\psi_x}\rangle^S\right),$$

where $S \sim \mathsf{Stab}(n,k)$ is a uniformly random stabilizer (specified in terms of a classical description of S), $E \sim \mathcal{D}_p^{\otimes n}$ is a Pauli error with $E \in \bar{\mathcal{P}}_n$, $x \sim \{0,1\}^k$ is a random string , and $|\overline{\psi_x}\rangle^S \in C(S)$ is the codeword

$$|\overline{\psi_x}\rangle^S := U^S_{\mathrm{Enc}}(|0^{n-k}\rangle \otimes |x\rangle)$$

for some canonical encoding circuit U^S_{Enc} for the stabilizer code associated with S. We say that a quantum algorithm solves the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem if it runs in time $\mathsf{poly}(k)$ and succeeds with probability at least $1/\mathsf{poly}(k)$ over the choice of S, E and x, and its internal randomness.

Let us first state a few remarks.

Remark 4.2 (Density matrix formulation)**.** *In Definition 4.1, the input to the learning algorithm is stated in the form $(S \in \mathsf{Stab}(n,k), E\,|\overline{\psi_x}\rangle^S)$, where $S \sim \mathsf{Stab}(n,k)$, $E \sim \mathcal{D}_p^{\otimes n}$ and $x \sim \{0,1\}^k$ is a random string. The pure state $E\,|\overline{\psi_x}\rangle^S$, however, should rather be understood as a density matrix of the form*

$$\rho_x^S = \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S) = \sum_{E \in \bar{\mathcal{P}}_n} \Pr_{E \sim \mathcal{D}_p^{\otimes n}}[E] \cdot E\,|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S\,E^\dagger,$$

where we think of $\mathcal{D}_p^{\otimes n}$ as a product of local depolarizing channels with parameter p.

Remark 4.3 (Learning vs. decoding)**.** *The name 'Learning Stabilizers with Noise' was chosen to parallel the name of the classical hardness assumption 'Learning Parities with Noise'. The word 'learning' here should be understood in the sense of Quantum Probably Approximately Correct (PAC) learning, where the example(s) seen by the learner/solver for LSN is a/are random quantum state(s). Notably, unlike in the setting of tomography, the quantum learner does not have multiple identical copies of the same quantum state.*

Remark 4.4 (Clifford representation)**.** *Recall that the input of the learner in Definition 4.1 consists of a random stabilizer $S \in \mathsf{Stab}(n,k)$. In Theorem 3.9, we showed that random stabilizer codes can be equivalently described by uniformly random Clifford encoding circuits. Therefore, we can alternatively think of an $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ instance as*

$$\left(C \in \mathsf{Cliff}_n,\ E\,C(|0^{n-k}\rangle \otimes |x\rangle)\right)$$

where $C \sim \mathsf{Cliff}_n$ is a random n-qubit Clifford operator and the first argument refers to a classical description of C, $E \sim \mathcal{D}_p^{\otimes n}$ is an n-qubit Pauli error, and $x \sim \{0,1\}^k$ is a random string.

Remark 4.5 (Worst-case vs. average-case)**.** *The computational task of solving $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ in Definition 4.1 is an average-case problem in the sense that the success probability of any algorithm is measured on average over the choice of stabilizer code S, error E and secret x. We also consider the worst-case variant, denoted by $\mathsf{LSN}_{n,k}$, where E, S and x are not assumed to be random and instead chosen adversarially subject to appropriate constraints. We say that an algorithm solves $\mathsf{LSN}_{n,k}$ in the worst-case with probability $\delta > 0$, if the algorithm succeeds at finding x with probability δ when challenged on any instance of the problem—even for adversarial choices of stabilizer code S, error E and secret x.*

Remark 4.6 (Polynomial vs. quasi-polynomial hardness)**.** *In Definition 4.1, we considered the "polynomial hardness" of the LSN problem, i.e., we assume that a successful solver must run in time* $\mathsf{poly}(k)$ *and succeed with probability at least* $1/\mathsf{poly}(k)$*. In Section 7, we also establish a connection to "quasi-polynomial hardness" of the problem, where we assume that a successful solver must run in quasi-polynomial-time and must succeed with inverse-quasi-polynomial probability in* k*.*

General variants. Recall that the input in Definition 4.1 consists of a sample

$$\left(S, E\,|\overline{\psi_x}\rangle^S\right) \;\sim\; \mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$$

where $S \sim \mathsf{Stab}(n,k)$ is a (classical description of) uniformly random stabilizer, where $E \sim \mathcal{D}_p^{\otimes n}$ is a Pauli error $E \in \bar{\mathcal{P}}_n$, where $x \sim \{0,1\}^k$ is a random string. Occasionally, we also consider more general variants of the problem, denoted by $\mathsf{LSN}_{n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$, that feature samples

$$\left(S, E\,|\overline{\psi_x}\rangle^S\right) \;\sim\; \mathsf{LSN}_{n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$$

which are captured by the following set of distributions (which depend on n and k):

- $\mathcal{N}$ is a *general noise distribution* with support over n-qubit Pauli errors $E \in \bar{\mathcal{P}}_n$.
- $\mathcal{S}$ is a *general distribution over Stabilizer codes*, either with support over the set of stabilizers $S \in \mathsf{Stab}(n,k)$, or over Clifford encoding circuits in $C \in \mathsf{Cliff}_n$.
- $\mathcal{I}$ is a *general distribution over input strings* of the form $x \in \{0,1\}^k$.

Depending on the choice of distributions $\mathcal{N}$, $\mathcal{S}$ and $\mathcal{I}$, the learning task may either become easier or harder than the canonical learning problem in Definition 4.1.

4.2 Existence of Unique Solutions

We now investigate for which parameter regime the learning problem $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ in Definition 4.1 is well-defined and admits unique solutions. In anticipation of Section 8 and Section 9.2, we show the existence of a unique solution by phrasing the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem as an *Uhlmann transformation problem*. An Uhlmann transformation problem, which will be formally defined in Section 8, can be understood as the problem of implementing the unitary in Uhlmann's theorem Theorem 3.2.

Uhlmann transformations and the LSN problem. To argue that the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem has a unique solution, we appeal to Uhlmann's theorem. Suppose we are given as input an instance with repsect to the Clifford representation,

$$\left(C \in \mathsf{Cliff}_n, E\,C(|0^{n-k}\rangle \otimes |x\rangle)\right)$$

where $C \sim \mathsf{Cliff}_n$ is a random n-qubit Clifford, $E \sim \mathcal{D}_p^{\otimes n}$ is an n-qubit Pauli error from a local depolarizing channel, and $x \sim \{0,1\}^k$ is a random k-bit secret. We can represent the density matrix corresponding to the quantum part of the instance[13] as the result of discarding register A of the purification,

$$|Q^0\rangle_{\mathsf{AB}} = \sqrt{2^{-k}} \sum_x \sum_{E_a} \sqrt{\Pr_{E_a \sim \mathcal{D}_p^{\otimes n}}[E_a]} \left(\,|x\rangle \otimes |a\rangle\,\right)_{\mathsf{A}} \otimes \left(E_a\, C(|0^{n-k}\rangle \otimes |x\rangle) \otimes |0\rangle\right)_{\mathsf{B}}. \quad (17)$$

[13] We also append an ancilla register in the state $|0\rangle$ for convenience.

For the sake of the proof, we also consider the following bipartite state given by

$$|Q^1\rangle_{\mathsf{AB}} = \sqrt{2^{-k}} \sum_x \sum_{E_a} \sqrt{\Pr_{E_a \sim \mathcal{D}_p^{\otimes n}}[E_a]} \left(|x\rangle \otimes |a\rangle \right)_{\mathsf{A}} \otimes \left(|0^{n-k}\rangle \otimes |x\rangle \otimes |a\rangle \right)_{\mathsf{B}}. \tag{18}$$

Using the quantum Gilbert-Varshamov bound, we can argue that the fidelity between the reduced states Q^0_{A} and Q^1_{A} on register A is near maximal—provided that n is slightly larger than k, and that $p \in (0, 1/2)$ is a sufficiently small constant. Therefore, by Uhlmann's theorem, there exists a unitary $U \in \mathrm{L}(\mathcal{H}_{\mathsf{B}})$ which acts on the B register and maps $|Q^0\rangle_{\mathsf{AB}}$ to another state which has near maximal overlap with $|Q^1\rangle_{\mathsf{AB}}$. In other words, the Uhlmann unitary U allows us to solve $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ and to recover the x with overwhelming probability—thereby proving uniqueness.

To make this explicit, we first show the following technical lemma.

Lemma 4.7. *Let $n, k \in \mathbb{N}$ and let $|Q^0\rangle_{\mathsf{AB}}$ and $|Q^1\rangle_{\mathsf{AB}}$ be the bipartite states in Eq. (17) and Eq. (18), respectively, for some $p \in (0, 1/2)$ and Clifford $C \in \mathsf{Cliff}_n$. Then, with probability at least $1 - 3np \cdot 2^{nH(3p)} \cdot 3^{3np} \cdot 2^{-n+k}$ over the choice of the random Clifford C, it holds that*

$$\mathrm{F}(Q^0_{\mathsf{A}}, Q^1_{\mathsf{A}}) \geq 1 - 4 \cdot e^{-\frac{np}{24}}.$$

Proof. Before we bound the fidelity, we first make a couple of observations. Define the projector onto system A given by

$$\mathbf{\Pi}_{n,p} = \mathbb{I}_k \otimes \sum_{a \,:\, |E_a| \leq \frac{3}{2}np} |a\rangle\langle a|. \tag{19}$$

Note that $\mathbf{\Pi}_{n,p}$ projects onto the support of all n-qubit Pauli operators of weight at most $\frac{3}{2}np$. For convenience, we also define the reduced density matrices

$$\hat{Q}^b_{\mathsf{A}} := \frac{\mathrm{Tr}_{\mathsf{B}}\left[(\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}}) Q^b_{\mathsf{AB}} (\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}})\right]}{\mathrm{Tr}[(\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}}) Q^b_{\mathsf{AB}} (\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}})]}, \qquad \text{for } b \in \{0, 1\}.$$

First, we observe the following about the reduced state $Q^0_{\mathsf{A}} = \mathrm{Tr}_{\mathsf{B}}\left[Q^0_{\mathsf{AB}}\right]$:

$$\begin{aligned}
&\delta_{\mathsf{TD}}\left(\mathrm{Tr}_{\mathsf{B}}\left[Q^0_{\mathsf{AB}}\right], \hat{Q}^0_{\mathsf{A}}\right) \\
&\leq \delta_{\mathsf{TD}}\left(Q^0_{\mathsf{AB}}, \frac{(\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}}) Q^0_{\mathsf{AB}} (\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}})}{\mathrm{Tr}[(\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}}) Q^0_{\mathsf{AB}} (\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}})]}\right) && \text{(Monotinicity of } \delta_{\mathsf{TD}}) \\
&\leq \sqrt{1 - \mathrm{Tr}[(\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}}) Q^0_{\mathsf{AB}}]} && \text{(Gentle measurement)} \\
&= \sqrt{\Pr_{E \sim \mathcal{D}_p^{\otimes n}}\left[|E| > \frac{3}{2}np\right]} \\
&\leq \exp\left(-\frac{np}{24}\right). && \text{(Chernoff bound)}
\end{aligned}$$

Next, we make the following observation. By the quantum Gilbert-Varshamov, it follows that a random stabilizer code is non-degenerate and has distance at least $d = 3np + 1$ with probability at least $1 - 3np \cdot 2^{nH(3p)} \cdot 3^{3np} \cdot 2^{-n+k}$ over the choice of encoding Clifford C, in which case for any pair of codewords with $x, y \in \{0, 1\}^k$:

$$\langle\overline{\psi_x}| E_a^\dagger E_b |\overline{\psi_y}\rangle = 0 \tag{20}$$

by the Knill-Laflamme conditions—provided the errors E_a, E_b have weight at most $|E_a|, |E_b| \leq \frac{3}{2}np$. Thus, the aforementioned reduced states must be identical:

$$\hat{Q}^0_{\mathsf{A}} = \hat{Q}^1_{\mathsf{A}}. \tag{21}$$

To complete the proof, we now make another observation about the reduced state $Q^1_{\mathsf{A}} = \mathrm{Tr}_{\mathsf{B}}\left[Q^1_{\mathsf{AB}}\right]$. Using a similar approach as before, we obtain the bound

$$\begin{aligned}
\delta_{\mathsf{TD}}\left(\hat{Q}^0_{\mathsf{A}}, Q^1_{\mathsf{A}}\right) &\leq \delta_{\mathsf{TD}}\left(\hat{Q}^0_{\mathsf{A}}, \hat{Q}^1_{\mathsf{A}}\right) + \delta_{\mathsf{TD}}\left(\hat{Q}^1_{\mathsf{A}}, Q^1_{\mathsf{A}}\right) && \text{(Triangle ineq.)} \\
&\leq 0 + \delta_{\mathsf{TD}}\left(\hat{Q}^1_{\mathsf{A}}, Q^1_{\mathsf{A}}\right) && \text{(By Equation (21))} \\
&\leq \sqrt{1 - \mathrm{Tr}[(\mathbf{\Pi}_{n,p} \otimes \mathbb{I}_{\mathsf{B}})Q^0_{\mathsf{AB}}]} && \text{(Gentle measurement)} \\
&= \sqrt{\Pr_{E\sim\mathcal{D}_p^{\otimes n}}\left[|E| > \frac{3}{2}np\right]} \\
&\leq \exp\left(-\frac{np}{24}\right). && \text{(Chernoff bound)}
\end{aligned}$$

Putting everything together, we can now apply the Fuchs-van de Graaf inequality, followed by the triangle inequality, followed by Bernoulli's inequality, to lower bound the fidelity between the reduced states as follows:

$$\begin{aligned}
\mathrm{F}(Q^0_{\mathsf{A}}, Q^1_{\mathsf{A}}) &\geq \left(1 - \delta_{\mathsf{TD}}\left(Q^0_{\mathsf{A}}, Q^1_{\mathsf{A}}\right)\right)^2 \\
&\geq \left(1 - \delta_{\mathsf{TD}}\left(\mathrm{Tr}_{\mathsf{B}}\left[Q^0_{\mathsf{AB}}\right], \hat{Q}^0_{\mathsf{A}}\right) - \delta_{\mathsf{TD}}\left(\hat{Q}^0_{\mathsf{A}}, \mathrm{Tr}_{\mathsf{B}}\left[Q^1_{\mathsf{AB}}\right]\right)\right)^2 \\
&\geq \left(1 - 2\exp\left(-\frac{np}{24}\right)\right)^2 \;\geq\; 1 - 4\exp\left(-\frac{np}{24}\right).
\end{aligned}$$

□

Using Lemma 4.7, we can now argue that $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ admits unique solutions with overwhelming probability—provided that n is slightly larger than k, say $n \geq 8k$, and that $p \in (0, 1/2)$ is a sufficiently small constant, for example $p = 0.05$.

Lemma 4.8 (Existence of unique solutions)**.** *Let $n, k \in \mathbb{N}$ be integers and let $p \in (0, 1/2)$ be a parameter. Then, the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem admits a unique solution and can be solved information-theoretically with probability at least $1 - \delta$, where*

$$\delta \;\leq\; 3np \cdot 2^{nH(3p)} \cdot 3^{3np} \cdot 2^{-n+k} + 2 \cdot e^{-\frac{np}{48}}.$$

Proof. Suppose we are given as input an instance in the Clifford representation,

$$\left(C \in \mathsf{Cliff}_n, E_a\, C(|0^{n-k}\rangle \otimes |x\rangle)\right) \tag{22}$$

where $C \sim \mathsf{Cliff}_n$ is a random n-qubit Clifford, $E_a \sim \mathcal{D}_p^{\otimes n}$ is an n-qubit Pauli error from a local depolarizing channel, and $x \sim \{0,1\}^k$ is a random k-bit secret.

Let $|Q^0\rangle_{\mathsf{AB}}$ and $|Q^1\rangle_{\mathsf{AB}}$ be the bipartite states in Eq. (17) and Eq. (18), respectively, for some $p \in (0, 1/2)$. Note that $|Q^0\rangle_{\mathsf{AB}}$ depends on the Clifford $C \in \mathsf{Cliff}_n$. Then, Lemma 4.7 shows that, with probability at least $1 - 3np \cdot 2^{nH(3p)} \cdot 3^{3np} \cdot 2^{-n+k}$ over the choice of the random Clifford C, it holds that $\mathrm{F}(Q^0_{\mathsf{A}}, Q^1_{\mathsf{A}}) \geq 1 - 4 \cdot e^{-\frac{np}{24}}$. Therefore, by Uhlmann's

theorem, there exists a unitary $U \in \mathrm{L}(\mathcal{H}_{\mathsf{B}})$ which acts on the B register and maps $|Q^0\rangle_{\mathsf{AB}}$ to another state which achieves an overlap of at least $4 \cdot e^{-\frac{np}{24}}$ with $|Q^1\rangle_{\mathsf{AB}}$. Applying the Uhlmann unitary U on the quantum part of the input from Equation (22) together with an additional ancilla register $|0\rangle$, i.e.,

$$E_a C(|0^{n-k}\rangle \otimes |x\rangle) \otimes |0\rangle \quad \mapsto \quad U\left(E_a\, C(|0^{n-k}\rangle \otimes |x\rangle) \otimes |0\rangle\right)$$

we obtain a state which—via Fuchs-van De Graaf—is within trace distance at most $2e^{-\frac{np}{48}}$ of the state $|0^{n-k}\rangle \otimes |x\rangle \otimes |a\rangle$. In other words, the Uhlmann unitary U allows us to solve $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ and to recover the x with the desired success probability. □

4.3 Multi-Shot Variant

Recall that the quantum learning problem $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ in Definition 4.1 only involves a single quantum sample of the form

$$\left(S, E\,|\overline{\psi_x}\rangle^S\right) \;\sim\; \mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}.$$

One may reasonably ask: does the learning problem become easier if the learner instead receives many independently chosen samples $\{S_i, E_i\,|\overline{\psi_x}\rangle^{S_i}\}_{i\in[m]}$, where the secret x remains the same throughout each sample? This motivates us to consider a multi-shot variant of the LSN problem, which we study in this section. Note that the multi-sample learning task also shows up naturally in the related LPN problem (see Section 5). We define the multi-shot variant of the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ as follows.

Definition 4.9 (Multi-Shot Learning Stabilizers with Noise)**.** *Let $k \in \mathbb{N}$ be the security parameter and let $n = \mathsf{poly}(k)$ be an integer. Let $p \in (0, 1/2)$ be a parameter. The multi-shot Learning Stabilizers with Noise ($\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$) problem is to find $x \in \{0,1\}^k$ given*

$$\left\{S_i \in \mathsf{Stab}(n,k), E_i\,|\overline{\psi_x}\rangle^{S_i}\right\}_{i=1}^m,$$

where, for each index $i \in [m]$, $S_i \sim \mathsf{Stab}(n,k)$ is a uniformly random stabilizer and $E_i \sim \mathcal{D}_p^{\otimes n}$ is a Pauli error $E \in \bar{\mathcal{P}}_n$, and where $x \sim \{0,1\}^k$ is a random string.

Depending on the application, it may be more natural to think of m, the sample complexity, as a parameter that one wishes to minimize, rather than as a fixed parameter. As previously mentioned in Remark 4.4, we can equivalently use the Clifford representation and think of an $\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$ instance a a tuple

$$\left\{C_i \in \mathsf{Cliff}_n,\; E_i\, C_i(|0^{n-k}\rangle \otimes |x\rangle)\right\}_{i=1}^m,$$

where $C_i \sim \mathsf{Cliff}_n$ and $E_i \sim \mathcal{D}_p^{\otimes n}$, for each index $i \in [m]$, and where $x \sim \{0,1\}^k$.

It is not hard to see that $\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$ reduces to (a special variant of) the single-shot problem in Definition 4.1. Indeed, we can show the following.

Lemma 4.10 ($\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$ reduces to a special variant of LSN)**.** *Let $k \in \mathbb{N}$, let $n = \mathsf{poly}(k)$ be an integer and $p \in (0, 1/2)$ be a parameter. Then, there exist distributions $\mathcal{S}$ and $\mathcal{I}$ which depend on m, n, k and p and have support over Cliff_{mn} and $\{0,1\}^{mk}$, respectively, such that $\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$ reduces to $\mathsf{LSN}_{mn,mk,\mathcal{D}_p^{\otimes mn},\mathcal{S},\mathcal{I}}$.*

Proof. Suppose we are given as input an $\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$ instance of the form

$$\left\{C_i \in \mathsf{Cliff}_n,\ E_i\, C_i(|0^{n-k}\rangle \otimes |x\rangle)\right\}_{i=1}^{m},$$

where $C_i \sim \mathsf{Cliff}_n$ and $E_i \sim \mathcal{D}_p^{\otimes n}$, for each index $i \in [m]$, and where $x \sim \{0,1\}^k$.

Consider the reduction which proceeds as follows:

1. Let $\pi \in \mathfrak{S}_{mn}$ be the permutation on mn elements with permutation operator

$$Q(\pi)\left((|0^{n-k}\rangle \otimes |x\rangle) \otimes \cdots \otimes (|0^{n-k}\rangle \otimes |x\rangle)\right) = |0^{m(n-k)}\rangle \otimes |x^m\rangle$$

   where $x^m = x \cdots x$ such that x is repeated m times.

2. Run the solver for the $\mathsf{LSN}_{mn,mk,\mathcal{N},\mathcal{S},\mathcal{I}}$ problem on input

$$\left(Q(\pi)\Big(\bigotimes_{i\in[m]} C_i\Big)Q(\pi)^\dagger,\ Q(\pi)\bigotimes_{i\in[m]} E_i\, C_i(|0^{n-k}\rangle \otimes |x\rangle)\right).$$

Here, we define the corresponding distributions $\mathcal{N}$, $\mathcal{S}$ and $\mathcal{I}$ as follows:

- $\mathcal{N}$ is the product distribution $\mathcal{D}_p^{\otimes mn}$ over local depolarizing channels.
- $\mathcal{S}$ is the distribution over Stabilizer codes which outputs Cliffords of the form

$$Q(\pi)\Big(\bigotimes_{i\in[m]} C_i\Big)Q(\pi)^\dagger, \qquad \text{for } C_i \sim \mathsf{Cliff}_n,\ \forall i \in [m].$$

- $\mathcal{I}$ is the distribution over $\{0,1\}^{mk}$ which first samples $x \sim \{0,1\}^k$, and then outputs $x^m = x \cdots x$ where x is repeated m times.

We now argue that our reduction allows us to solve $\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$—provided that we have a solver for $\mathsf{LSN}_{mn,mk,\mathcal{N},\mathcal{S},\mathcal{I}}$. To this end, we observe that

$$\begin{aligned}
&Q(\pi)\bigotimes_{i\in[m]} E_i\, C_i(|0^{n-k}\rangle \otimes |x\rangle) \\
&= Q(\pi)\left(\bigotimes_{i\in[m]} E_i\, C_i\right) Q(\pi)^\dagger Q(\pi)\left((|0^{n-k}\rangle \otimes |x\rangle) \otimes \cdots \otimes (|0^{n-k}\rangle \otimes |x\rangle)\right) \\
&= Q(\pi)\left(\bigotimes_{i\in[m]} E_i\, C_i\right) Q(\pi)^\dagger\left(|0^{m(n-k)}\rangle \otimes |x^m\rangle\right) \\
&= Q(\pi)\left(\bigotimes_{i\in[m]} E_i\right) Q(\pi)^\dagger Q(\pi)\left(\bigotimes_{i\in[m]} C_i\right) Q(\pi)^\dagger\left(|0^{m(n-k)}\rangle \otimes |x^m\rangle\right).
\end{aligned}$$

Since the distribution $\mathcal{D}_p^{\otimes mn}$ is invariant under permutations via $Q(\pi)$, our reduction produces an instance of $\mathsf{LSN}_{mn,mk,\mathcal{N},\mathcal{S},\mathcal{I}}$, as desired. □

5 Reduction from Learning Parity with Noise

In this section, we show that LSN is a rich assumption which captures the classical LPN problem as a special case. Despite of more than a quarter of a century of study, the fastest known (classical or quantum) algorithms for LPN still run in exponential time [BKW03]. Because the LSN problem (in some sense) subsumes LPN, this can be seen as additional evidence for the average-case hardness of our learning problem.

In fact, our also results suggest that the LSN problem—precisely because it is an inherently quantum assumption—may in fact be even harder to break than its classical counterpart, which makes it particularly appealing as a basis of hardness in quantum cryptography.

5.1 Learning Parity with Noise

Recall that the LPN assumption [BFKL94] says that it is computationally difficult to decode a random linear code under Bernoulli noise.

Definition 5.1 (Learning Parity with Noise)**.** *Let $n, k \in \mathbb{N}$ and let $p \in (0, 1/2)$ be a parameter. The Learning Parity with Noise ($\mathsf{LPN}_{n,k,\mathsf{Ber}_p^{\otimes n}}$) problem is to find $\mathbf{x}$ given as input*

$$(\mathbf{A} \sim \mathbb{Z}_2^{n\times k}, \mathbf{A}\cdot\mathbf{x} + \mathbf{e} \pmod 2)$$

where $\mathbf{x} \sim \mathbb{Z}_2^k$ and $\mathbf{e} \sim \mathsf{Ber}_p^{\otimes n}$ is a random Bernoulli error term.

The LPN decoding problem is believed to be hard against both classical and quantum algorithms running in time $\mathsf{poly}(k)$ in the *constant-noise* regime, where $p \in (0, 1/2)$ is a constant and $n = \mathsf{poly}(k)$. In this regime, the celebrated BKW algorithm [BKW03] solves LPN with time/sample complexity given by $O(2^{k/\log k})$.

Next, we study the relationship between LPN and LSN.

5.2 Quantum Reduction to LSN

The goal of this section is make the connection between LSN and LPN more explicit; in particular, we show that LPN reduces to a special case of LSN. Specifically, we show that $\mathsf{LPN}_{n,k,\mathsf{Ber}_p^{\otimes n}}$ reduces to $\mathsf{LSN}_{n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$ with respect to the following set of distributions:

- $\mathcal{N}$ is the product $\mathcal{F}_p^{\otimes n}$ of bit-flip noise channels $\mathcal{F}_p$ with parameter $p > 0$.
- $\mathcal{S}$ is the distribution over Cliff_n which samples $C \sim \mathcal{S}$ as follows: first, sample a random matrix $\mathbf{A} \in \mathbb{Z}_2^{n\times k}$ of full column-rank, and let C be the corresponding matrix-multiplication Clifford operator $C : |0^{n-k}\rangle \otimes |\mathbf{x}\rangle \to |\mathbf{A}\cdot\mathbf{x} \pmod 2\rangle$.
- $\mathcal{I}$ is the uniform distribution over $x \in \{0,1\}^k$.

Finally, we ask: can we also provide hardness results in the multi-shot setting? We give a quantum reduction from LPN which applies in both settings.

Single-shot variant. First, we focus on the single-shot setting and show the following theorem relating LSN and LPN.

Theorem 5.2 (LPN reduces to LSN)**.** *Let $n, k \in \mathbb{N}$ be integers and let $p \in (0, 1/2)$ be a parameter. Suppose there exists an algorithm $\mathcal{A}$ that runs in time T and solves $\mathsf{LSN}_{n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$ with probability ϵ. Then, there exists an algorithm $\mathcal{B}$ which runs in time $\mathsf{poly}(k, T)$ and solves $\mathsf{LPN}_{n,k,\mathsf{Ber}_p^{\otimes n}}$ with probability at least $\epsilon \cdot \left(1 - k \cdot 2^{k-n-1}\right)$.*

Proof. Suppose we are given as input an instance

$$(\mathbf{A} \in \mathbb{Z}_2^{n\times k}, \mathbf{A}\cdot\mathbf{s}+\mathbf{e} \pmod 2) \quad \sim \quad \mathsf{LPN}_{n,k,\mathsf{Ber}_p^{\otimes n}}$$

where $\mathbf{A} \sim \mathbb{Z}_2^{n\times k}$, $\mathbf{s} \sim \mathbb{Z}_2^k$ is a secret vector and $\mathbf{e} \sim \mathsf{Ber}_p^{\otimes n}$ is a random Bernoulli error term. Consider the reduction $\mathcal{B}$ which proceeds as follows:

- If col-rank$(\mathbf{A}) < k$, $\mathcal{B}$ aborts.
- Else, if col-rank$(\mathbf{A}) = k$, $\mathcal{B}$ runs $\mathcal{A}$ on input

  $$(U_{\mathbf{A}}, |\mathbf{A}\cdot\mathbf{s}+\mathbf{e} \pmod 2\rangle)$$

  where $U_{\mathbf{A}}$ is the Clifford encoding circuit[14] encoding operation

  $$U_{\mathbf{A}} : |0^{n-k}\rangle \otimes |\mathbf{x}\rangle \to |\mathbf{A}\cdot\mathbf{x} \pmod 2\rangle$$

  which is an injective matrix multiplication for any vector $\mathbf{x} \in \mathbb{Z}_2^k$. In other words, $U_{\mathbf{A}}$ gives rise to the stabilizer subgroup $S_{\mathbf{A}} = \langle U_{\mathbf{A}} Z_1 U_{\mathbf{A}}^\dagger, \ldots, U_{\mathbf{A}} Z_{n-k} U_{\mathbf{A}}^\dagger\rangle$.

Let is now analyze the probability that $\mathcal{B}$ succeeds. First, we observe that

$$\Pr_{\mathbf{A}\sim\mathbb{Z}_2^{n\times k}}[\text{col-rank}(\mathbf{A}) = k] = \prod_{i=1}^{k}\left(1-2^{i-n-1}\right) \geq \left(1-2^{k-n-1}\right)^k \geq 1-k\cdot 2^{k-n-1}.$$

Here, the last inequality follows from Bernoulli's inequality. In other words, a uniformly random matrix $\mathbf{A} \sim \mathbb{Z}_2^{n\times k}$ has full column-rank with overwhelming probability, provided that n is only slightly larger than k. We can interpret the noisy sample $\mathbf{A}\cdot\mathbf{s}+\mathbf{e} \pmod 2$ as an ensemble of n-qubit pure states

$$\begin{aligned} |\mathbf{A}\cdot\mathbf{s}+\mathbf{e} \pmod 2\rangle &= X^{\mathbf{e}} |\mathbf{A}\cdot\mathbf{s} \pmod 2\rangle \\ &= X^{\mathbf{e}} U_{\mathbf{A}} \left(|0^{n-k}\rangle \otimes |\mathbf{s}\rangle\right), \end{aligned}$$

where $X^{\mathbf{e}} = X^{e_1} \otimes \cdots \otimes X^{e_n}$ is a product of Pauli-X operators. Note that, since the error $\mathbf{e} \sim \mathsf{Ber}_p^{\otimes n}$ comes from a Bernoulli distribution, it follows that $X^{\mathbf{e}} \sim \mathcal{F}_p^{\otimes n}$ corresponds to an n-qubit bit-flip error with parameter p.

Therefore, we conclude that $\mathcal{B}$ runs in time $\mathsf{poly}(k, T)$ and solves $\mathsf{LPN}_{n,k,\mathsf{Ber}_p^{\otimes n}}$ with probability at least $\epsilon \cdot \prod_{i=1}^{k}\left(1-2^{i-n-1}\right)$. □

Multi-shot variant. Finally, we consider the multi-shot setting and show the following theorem which relates LPN and MSLSN.

Theorem 5.3 (LPN reduces to MSLSN)**.** *Let $m, n, k \in \mathbb{N}$ be integers and $p \in (0, 1/2)$. Suppose there exists an algorithm $\mathcal{A}$ that runs in time T and solves $\mathsf{MSLSN}_{m,n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$ with probability ϵ. Then, there exists an algorithm $\mathcal{B}$ which runs in time $\mathsf{poly}(k, T)$ and solves $\mathsf{LPN}_{nm,k,\mathsf{Ber}_p^{\otimes nm}}$ with probability at least $\epsilon \cdot \left(1 - m\cdot k\cdot 2^{k-n-1}\right)$.*

Proof. Suppose we are given as input an instance

$$(\mathbf{A} \sim \mathbb{Z}_2^{nm\times k}, \mathbf{A}\cdot\mathbf{s}+\mathbf{e} \pmod 2) \quad \sim \quad \mathsf{LPN}_{nm,k,\mathsf{Ber}_p^{\otimes nm}}$$

where $\mathbf{A} = [\mathbf{A}_1|\cdots|\mathbf{A}_m]^\intercal \in \mathbb{Z}_2^{nm\times k}$ consists of $\mathbf{A}_i \sim \mathbb{Z}_2^{n\times k}$, $\mathbf{s} \sim \mathbb{Z}_2^k$ is a secret vector and $\mathbf{e} = [\mathbf{e}_1|\cdots|\mathbf{e}_m]^\intercal \sim \mathsf{Ber}_p^{\otimes nm}$ consists of random Bernoulli error terms $\mathbf{e}_i \sim \mathsf{Ber}_p^{\otimes n}$. Consider the reduction $\mathcal{B}$ which proceeds as follows:

[14] Note that $U_{\mathbf{A}}$ can be described in terms of CNOT gates [PMH08], which are themselves Clifford gates.

- If there exists an index $i \in [m]$ such that col-rank$(\mathbf{A}_i) < k$, $\mathcal{B}$ aborts.
- Else, if col-rank$(\mathbf{A}_i) = k$ for all $i \in [m]$, $\mathcal{B}$ runs $\mathcal{A}$ on input

$$\left\{(U_{\mathbf{A}_i}, |\mathbf{A}_i \cdot \mathbf{s} + \mathbf{e}_i \pmod 2\rangle)\right\}_{i\in[m]}$$

  where $U_{\mathbf{A}_i}$ is the Clifford encoding circuit

$$U_{\mathbf{A}_i} : |0^{n-k}\rangle \otimes |\mathbf{x}\rangle \to |\mathbf{A}_i \cdot \mathbf{x} \pmod 2\rangle$$

  which is an injective matrix multiplication for any vector $\mathbf{x} \in \mathbb{Z}_2^k$.

Let is now analyze the probability that $\mathcal{B}$ succeeds. Again, we observe that

$$\begin{aligned}&\Pr_{\mathbf{A}=[\mathbf{A}_1|\cdots|\mathbf{A}_m]^\intercal\sim\mathbb{Z}_2^{nm\times k}}\left[\forall i \in [m] \,:\, \text{col-rank}(\mathbf{A}_i) = k\right]\\ &= \left(\prod_{i=1}^{k}\left(1-2^{i-n-1}\right)\right)^m \geq \left(1-2^{k-n-1}\right)^{mk} \geq 1 - m\cdot k\cdot 2^{k-n-1}.\end{aligned}$$

Therefore, we conclude that $\mathcal{B}$ runs in time $\mathsf{poly}(k, m, T)$ and solves $\mathsf{LPN}_{nm,k,\mathsf{Ber}_p^{\otimes nm}}$ with probability at least $\epsilon \cdot \left(1 - m \cdot k \cdot 2^{k-n-1}\right)$. □

6 Quantum Algorithms for Learning Stabilizers with Noise

In this section, we give both polynomial-time and exponential-time quantum algorithms for solving LSN in various depolarizing noise regimes. Concretely, we study the following quantum algorithms:

- **An *extremely low-noise* algorithm** for the regime with parameter $p \leq \frac{1}{n} - \frac{1}{n^{1+c}}$, for $c > 0$. In this case, we show that a simple projection onto the stabilizer codespace running in time $O(n^3)$ suffices to solve the problem with an inverse-polynomial success probability $1/n^c$.
- **A *low-noise* algorithm.** Surprisingly, for low enough noise rates, only a single sample is needed to solve the canonical LSN problem. In fact, we show that the so-called pretty good measurement (PGM) suffices. The proof leverages the quantum Gilbert-Varshamov bound, which says that a random stabilizer code has good distance with high probability.
- **A *higher noise* algorithm, up to a threshold.** We can extend our algorithm to work with higher constant noise rates, by taking more samples. We also show that there is a sharp threshold noise rate at which no more samples will help.

6.1 Single-Shot Decoding for Extremely Low Noise Rates

In this section, we first consider an extremely low noise regime in which the LSN problem becomes computationally tractable. Specifically, we consider the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem with parameter $p = \frac{1}{n} - \frac{1}{n^{1+c}}$, for any $c > 0$ and $n = \mathsf{poly}(k)$. We show that a simple projection onto the stabilizer code space suffices to solve $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ in time $O(n^3)$ with inverse-polynomial success probability.

Theorem 6.1 (Single-Shot Decoding for Extremely Low Noise Rates)**.** *Let $n, k \in \mathbb{N}$ be integers with $n \geq k$ and $n = \mathsf{poly}(k)$. Let $p \leq \frac{1}{n} - \frac{1}{n^{1+c}}$ be a noise parameter, for any $c > 0$. Then, Algorithm 1 runs in time $O(n^3)$ and solves the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem with inverse-polynomial probability of at least $\delta \geq 1/n^c$.*

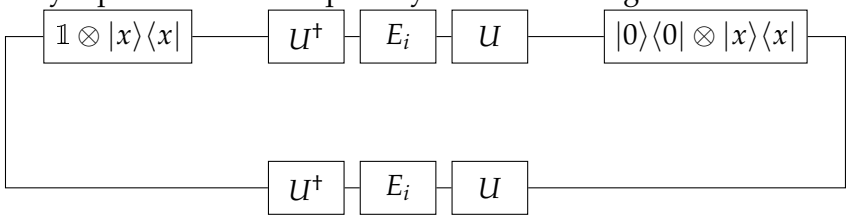
Proof. Suppose we are given as input an instance of the $\mathsf{LSN}_{n,k,\mathcal{D}}$ problem, i.e.,

$$\left(S \in \mathsf{Stab}(n,k), \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S)\right) \sim \mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$$

where $S \sim \mathsf{Stab}(n,k)$ describes a random stabilizer code and $x \sim \{0,1\}^k$ is a random element. The success probability of Algorithm 1 is then

$$\mathrm{Tr}\left(\mathbb{1}_{n-k} \otimes |x\rangle\langle x| (U^\dagger D_p^{\otimes n}(U|\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|U^\dagger)U\right) \text{ where } |\bar{0}\rangle := |0\rangle^{\otimes n-k}. \tag{23}$$

We may represent the trace quantity as the following tensor network diagram:



We will be interested in the average success probability over code, error and secret. In fact, we'll tackle the average over U and E_i first, fixing x:

$$\mathbb{E}_{x\sim\{0,1\}^n}\mathbb{E}_{U\sim\mathrm{Cliff}}\mathbb{E}_{E\sim\mathcal{D}_p^{\otimes n}}\left[\mathrm{Tr}\left(\mathbb{1}_{n-k} \otimes |x\rangle\langle x|(U^\dagger E(U|\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|)E^\dagger U^\dagger)U\right)\right] \tag{24}$$

$$= \sum_{E_i\in P_n} \Pr_{E_i\sim\mathcal{D}_p^{\otimes n}}[E_i]\langle\Gamma|\mathbb{1}_{n-k} \otimes |x\rangle\langle x| \otimes \mathbb{1}_n \mathbb{E}_{U\sim\mathrm{Cliff}}(U^{\dagger\otimes 2}E_i^{\otimes 2}U^{\otimes 2})|\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x| \otimes \mathbb{1}_n|\Gamma\rangle \tag{25}$$

$$= \Pr_{E_i\sim\mathcal{D}_p^{\otimes n}}[\mathbb{1}]\mathrm{Tr}(\mathbb{1}_{n-k} \otimes |x\rangle\langle x| \cdot |\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|) \tag{26}$$

$$+ \sum_{E_i\in P_n} \Pr_{E_i\sim\mathcal{D}_p^{\otimes n}}[E_i]\langle\Gamma|\mathbb{1}_{n-k} \otimes |x\rangle\langle x| \otimes \mathbb{1}_n \left(\frac{1}{4^n-1}\sum_{P\in P_n\setminus\mathbb{1}} P\otimes P\right)|\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x| \otimes \mathbb{1}_n|\Gamma\rangle \tag{27}$$

$$= (1-p)^n + \frac{1-(1-p)^n}{4^n-1}\sum_{P\in P_n\setminus\mathbb{1}} \mathrm{Tr}(\mathbb{1}_{n-k} \otimes |x\rangle\langle x|P|\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|P) \tag{28}$$

Here, we let $|\Gamma\rangle = \sum_{i\in[2^n]} |i\rangle\,|i\rangle$. In the second last equality, we have used the fact that Cliffords form a 2-design and thus it suffices to integrate over the Haar measure. In the last equality, we have used $\Pr_{E_i\sim\mathcal{D}_p^{\otimes n}}[\mathbb{1}] = (1-p)^n$. As a sanity check, we find that $1-p$ is the probability that noise applies no channel on every qubit, so p small implies the

Algorithm 1: Projection onto the Stabilizer Codespace

Input: Instance $(S \in \mathsf{Stab}(n,k), \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S)) \sim \mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$.

Output: A string $x' \in \{0,1\}^k$.

1 Let $\rho_x^S \leftarrow \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S)$ denote the ensemble instance.

2 Use the algorithm in Theorem 3.7 to find an encoding Clifford U_{Enc}^S associated with the stabilizer subgroup $S \in \mathsf{Stab}(n,k)$.

3 Compute $(U_{\mathrm{Enc}}^S)^\dagger \rho_x^S U_{\mathrm{Enc}}^S$ and measure in the computational basis.

4 Output the string x' corresponding to the last k bits of the measurement outcome.

noise rate is low. Continuing,

$$= (1-p)^n + \frac{1-(1-p)^n}{4^n-1} \cdot \mathrm{Tr}\left[\mathbb{1}_{n-k} \otimes |x\rangle\langle x| \left(4^n \mathbb{E}_{P\in P_n}(P|\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|P) - |\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|\right)\right] \tag{29}$$

$$= (1-p)^n + \frac{1-(1-p)^n}{4^n-1} \mathrm{Tr}\left[\mathbb{1}_{n-k} \otimes |x\rangle\langle x| \left(4^n \frac{\mathrm{Tr}(|\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|)}{2^n} \mathbb{1} - |\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|\right)\right] \tag{30}$$

$$= (1-p)^n + \frac{1-(1-p)^n}{4^n-1} \mathrm{Tr}\left(\mathbb{1}_{n-k} \otimes |x\rangle\langle x| \cdot 2^n - |\bar{0}\rangle\langle\bar{0}| \otimes |x\rangle\langle x|\right) \tag{31}$$

$$= (1-p)^n + (1-(1-p)^n)/2^k \tag{32}$$

$$\geq 1 - np + O((np)^2) \tag{33}$$

$$\geq \frac{1}{n^c} \tag{34}$$

In the second equality, we used the fact that Paulis are a 1-design and integrated over the Haar measure. The first inequality follows from Bernoulli's inequality and the last inequality follows from our assumption that $p \leq \frac{1}{n} - \frac{1}{n^{1+c}}$, for $c > 0$. □

6.2 Single-Shot Decoding for Low Constant Noise Rates

Recall that LSN is the following decoding problem: given as input an instance

$$\left(S \in \mathsf{Stab}(n,k), E\,|\overline{\psi_x}\rangle^S\right),$$

where $S \sim \mathsf{Stab}(n,k)$ is a random stabilizer code and $E \sim \mathcal{D}_p^{\otimes n}$ is a random n-qubit Pauli error $E \in \bar{\mathcal{P}}_n$, and where $x \sim [2^k]$ is a random element. In other words, we have a noisy state discrimination task $\{\rho_x\}_{x\in[2^k]}$ where ρ_x is the mixed state

$$\rho_x = \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|) = \sum_{E\in\bar{\mathcal{P}}_n} \Pr_{E\sim\mathcal{D}_p^{\otimes n}}[E] \cdot E\,|\overline{\psi_x}\rangle\langle\overline{\psi_x}|\,E^\dagger. \tag{35}$$

Our algorithm for solving LSN is to implement a *Pretty Good Measurement* [BK00, Mon07], but with a twist that enables us to bound its success probability: we will implement an approximation of the PGM that works for *truncated depolarizing noise*, noise from which we have culled the highest weight (and thus most destructive) errors. The reason this works is that the PGM is actually the optimal measurement for orthogonal states and

discriminates them perfectly. Using the Gilbert-Varshamov bound, we are able to harness the fact that under truncated depolarizing noise, encoded orthogonal states remain approximately orthogonal, and thus the PGM still works well for the task. This can be understood in another way: the purpose of a good error-correcting code is to encode quantum information in subspaces that do not contract much under quantum channels (i.e. two initial orthogonal states remain approximately orthogonal under the action of quantum channels)—a random stabilizer code is an example of such a code.

Pretty Good Measurement. We recall the following result by Montanaro:

Lemma 6.2 ([Mon07]). *Let $\{\rho_x\}$ be an ensemble of quantum states and let $\mathbf{\Lambda}_x = \mathbf{\Sigma}^{-\frac{1}{2}}\rho_x\mathbf{\Sigma}^{-\frac{1}{2}}$ with $\mathbf{\Sigma} = \sum_x \rho_x$, where inverses of $\mathbf{\Sigma}$ are taken with respect to its support. Then, the worst-case error of the pretty good measurement ensemble $\{\mathbf{\Lambda}_x\}$ is at most*

$$p_{err} = \max_x \left(1 - \mathrm{Tr}\left\{\mathbf{\Lambda}_x\rho_x\right\}\right) \leq \sum_{x\neq y} \sqrt{\mathsf{F}(\rho_x,\rho_y)}$$

where $\mathsf{F}(\rho_x,\rho_y)$ denotes the fidelity between ρ_x and ρ_y. Moreover, the PGM is optimal if the states in $\{\rho_x\}$ are pair-wise orthogonal, as then $\mathsf{F}(\rho_x,\rho_y) = 0$.

We're going to use the following block-encoding based algorithm in [GLM+22] for implementing the PGM. Let κ_ρ denote the reciprocal of the smallest eigenvalue of a density matrix ρ. We use the following theorem.

Theorem 6.3 ([GLM+22]). *The PGM measurement channel for $\{\rho_x\}_{x\in[2^k]}$ can be implemented with error ϵ (in terms of diamond distance) in time*

$$\widetilde{\mathcal{O}}\left(\sqrt{2^k\kappa_{\bar\rho}}N_\rho(\kappa_{\bar\rho} + \min(\kappa_\rho, 2^k\cdot\kappa_{\bar\rho}/\epsilon^2)\right),$$

where $\bar\rho = 2^{-k}\sum_{x\in[2^k]}\rho_x$ and where N_ρ denotes the size of the quantum circuit needed to implement a purification of ρ_x.

In our case, since a purification of ρ_x is

$$\sum_{E\in\mathcal{P}_n}\sqrt{\Pr_{E_a\sim\mathcal{D}_p^{\otimes n}}[E_a]}\,|a\rangle\otimes E_a U^S_{\mathrm{Enc}}(|0^{n-k}\rangle\otimes|x\rangle), \tag{36}$$

we may prepare this purification simply by applying U^S_{Enc} on a coherent superposition. The number of gates required to implement an n-qubit Clifford is $N_\rho = O(n^2)$. We show the following theorem.

Theorem 6.4 (Single-Shot Decoding for Low Constant Noise Rates). *Let $n,k\in\mathbb{N}$ and $\epsilon\in(0,1)$. Let $\mathcal{D}_p^{\otimes n}$ be the n-qubit depolarizing channel, for some $p\in(0,1/2)$ such that*

$$H(3p) + 3\log_2(3)p + k/n < .99 - \frac{\log(1/\epsilon)}{n}. \tag{37}$$

Then, there exists a quantum algorithm which runs in time

$$\widetilde{\mathcal{O}}\left(n^2\sqrt{2^k\kappa_{\bar\rho}}(\kappa_{\bar\rho} + \min(\kappa_\rho, 2^k\cdot\kappa_{\bar\rho}/\epsilon^2)\right).$$

and solves the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem with probability at least $1 - O(\epsilon)$.

Proof. Suppose we are given as input an instance of the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem, i.e.,

$$\left(S \in \mathsf{Stab}(n,k), \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S)\right) \sim \mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$$

where $S \sim \mathsf{Stab}(n,k)$ describes a random stabilizer code and $x \sim \{0,1\}^k$ is a random element. We now show that running Algorithm 2 on input $(S, \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S))$ and parameter $\epsilon \in (0,1)$ yields x with the desired success probability.

Algorithm 2, in fact, implements the PGM with respect to a slightly different ensemble as compared to the true ensemble of problem instances. That is to say, it will suffice to implement the pretty good measurement with respect to the ensemble $\{\tilde{\rho}_x^S\}_{x\in\{0,1\}^k}$ where

$$\tilde{\rho}_x^S := \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S) = \sum_{E\in\bar{\mathcal{P}}_n} \tilde{\mathbf{p}}^{(\frac{3n}{2})}(E)\cdot E\,|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S\,E^\dagger, \tag{38}$$

where $\tilde{\mathcal{D}}^{(\frac{3np}{2})}$ is the truncated depolarizing noise channel to be defined shortly; while we remind readers that the true ensemble of problem instances is

$$\rho_x^S := \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S) = \sum_{E\in\bar{\mathcal{P}}_n} \tilde{\mathbf{p}}^{(n)}(E)\cdot E\,|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S\,E^\dagger. \tag{39}$$

We now define the *truncated depolarizing noise channel* as the channel which acts on an input state ρ as

$$\tilde{\mathcal{D}}^{(w)}(\rho) := \sum_{E\in\mathcal{P}_n:|E|\leq w} \tilde{\mathbf{p}}^{(w)}(E) E\rho E^\dagger \tag{40}$$

where $p \in (0, 3/4)$ denotes the depolarizing noise parameter (c.f. Equation (10)), and the truncation lies in the fact that we restrict the support to Paulis with bounded weight. We define the truncated probability distribution via

$$\tilde{\mathbf{p}}^{(w)}(E) := \frac{1}{N}\left(\frac{p}{3}\right)^{|E|}(1-p)^{n-|E|} \tag{41}$$

and N is a normalization factor, i.e. $N = \sum_{E\in\bar{\mathcal{P}}_n:|E|\leq w} \tilde{\mathbf{p}}^{(w)}(E)$, that ensures that $\tilde{\mathbf{p}}^{(w)}$ is a probability distribution over $\bar{\mathcal{P}}_n$. The distribution corresponding to the n-qubit local depolarizing noise channel corresponds to $w = n$, i.e. $\tilde{\mathbf{p}}^{(n)}(E) = \Pr_{E\sim\mathcal{D}_p^{\otimes n}}[E]$ and the truncated channel corresponds to acting only with the Paulis with weight at most w, with the same relative probabilities as in n-qubit local depolarizing noise. Because the weights of the Pauli channels in the decomposition of $\mathcal{D}_p^{\otimes n}$ are distributed as a binomial $w \sim \mathsf{Binom}(n,p)$, one can show that for $w = 3/2np$, the total variation distance between the probability distribution over the n-qubit Pauli channels induced by $\mathcal{D}_p^{\otimes n}$ and $\tilde{\mathcal{D}}^{(w)}$ is

$$\delta_{\mathsf{TV}}\left(\tilde{\mathbf{p}}^{(3/2np)}, \tilde{\mathbf{p}}^{(n)}\right) \leq \Pr_{E\sim\mathcal{D}_p^{\otimes n}}\left[|E| \geq \frac{3}{2}np\right] \leq \exp\left(-\frac{np}{12}\right), \tag{42}$$

using a Chernoff bound.

Algorithm 2 implements the PGM measurement channel with (diamond distance) error $\epsilon \in (0,1)$ with the stated time complexity. Let us first analyze the error probability

Algorithm 2: Pretty Good Measurement for LSN

Input: Instance $(S \in \mathsf{Stab}(n,k), \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S)) \sim \mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ and $\epsilon \in (0,1)$.

Output: A string $x' \in \{0,1\}^k$.

1 Let $\rho_x^S \leftarrow \mathcal{D}_p^{\otimes n}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S)$ denote the ensemble instance.

2 Use the algorithm in Theorem 6.3 with precision ϵ to measure ρ_x^S via the POVM

$$\{\tilde{\mathbf{\Lambda}}_x^S\}_{x\in\{0,1\}^k} \quad \text{with} \quad \tilde{\mathbf{\Lambda}}_x^S = \mathbf{\Sigma}^{-\frac{1}{2}}\tilde{\rho}_x^S\,\mathbf{\Sigma}^{-\frac{1}{2}}$$

where $\mathbf{\Sigma} = \sum_x \tilde{\rho}_x^S$ and where inverses of $\mathbf{\Sigma}$ are taken with respect to its support, and where the state $\tilde{\rho}_x^S$ is defined in Equation (38).

3 Output the measurement outcome $x' \in \{0,1\}^k$.

of the (ideal) pretty good measurement $\{\tilde{\mathbf{\Lambda}}_x^S\}_x$:

$$\begin{aligned}
\max_x \left(1 - \mathrm{tr}(\widetilde{\mathbf{\Lambda}}_x^S \rho_x^S)\right) &= \max_x \left(1 - \mathrm{tr}\left(\widetilde{\mathbf{\Lambda}}_x \left(\rho_x^S - \tilde{\rho}_x^S\right)\right) - \mathrm{tr}\left(\widetilde{\mathbf{\Lambda}}_x \tilde{\rho}_x^S\right)\right) && (43)\\
&\leq \max_x \left(1 + \delta_{\mathsf{TD}}\left(\rho_x^S, \tilde{\rho}_x^S\right) - \mathrm{tr}\left(\widetilde{\mathbf{\Lambda}}_x \tilde{\rho}_x^S\right)\right) && (44)\\
&\leq \delta_{\mathsf{TV}}\left(\tilde{\mathbf{p}}^{(n)}, \tilde{\mathbf{p}}^{(w)}\right) + \max_x \left(1 - \mathrm{tr}\left(\widetilde{\mathbf{\Lambda}}_x \tilde{\rho}_x^S\right)\right) && (45)\\
&= e^{-\frac{np}{12}} + \max_x \left(1 - \mathrm{tr}\left(\widetilde{\mathbf{\Lambda}}_x \tilde{\rho}_x^S\right)\right) && (46)\\
&\leq e^{-\frac{np}{12}} + \sum_{x\neq y} \sqrt{\mathsf{F}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^S))}. && (47)
\end{aligned}$$

The second inequality comes from the strong convexity of trace distance and Equations (38) and (39). The second last equality comes from Eq. 42. The last inequality comes from Lemma 6.2.

To finish the proof, it suffices to bound the pair-wise fidelities in Equation (47). Here, we exploit the special structure of the encoded states. We appeal to the Gilbert-Varshamov bound (Theorem 3.10), which states that random stabilizer codes are non-degenerate (i.e. have good distance) with high probability. This means that errors of weight at most $\frac{3}{2}np$ acting on orthogonal states keep them orthogonal. Concretely, this implies that

$$\begin{aligned}
&\Pr_{S\sim\mathsf{Stab}(n,k)}\left[E_a^\dagger E_b \notin N(S), \forall\, |E_a|, |E_b| \leq \frac{3}{2}np\right] && (48)\\
&= \Pr_{S\sim\mathsf{Stab}(n,k)}\left[\langle\overline{\psi_x}|^S E_a^\dagger E_b\, |\overline{\psi_y}\rangle^S = 0\, \forall x \neq y, \forall |E_a|, |E_b| \leq \frac{3}{2}np\right] && (49)\\
&\geq 1 - 3/2np \cdot 2^{nH(3/2p)} \cdot 3^{3/2np} \cdot 2^{-n+k}. && (50)
\end{aligned}$$

The first equality follows from the observations mentioned above (Theorem 3.10). Moreover, it follows from Uhlmann's theorem that

$$\begin{aligned}
&\mathsf{F}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^S)) && (51)\\
&= \max_U |\langle\phi^{\rho_x}|\,(U\otimes I)\,|\phi^{\rho_y}\rangle|^2 && (52)\\
&= \max_U \left|\sum_{E_a,E_b\in\bar{\mathcal{P}}_n} \sqrt{\tilde{\mathbf{p}}^{(\frac{3np}{2})}(E_a)\tilde{\mathbf{p}}^{(\frac{3np}{2})}(E_b)}\,\langle a|\,U\,|b\rangle \cdot \langle\overline{\psi_x}|\,E_a^\dagger E_b\,|\overline{\psi_y}\rangle\right|^2 && (53)
\end{aligned}$$

where, for $x \in \{0,1\}^k$, we defined the purification

$$|\phi^{\rho_x}\rangle = \sum_{E_a \in \bar{\mathcal{P}}_n} \sqrt{\tilde{\mathbf{p}}^{(\frac{3np}{2})}(E_a)}\ |a\rangle \otimes E_a\, |\overline{\psi_x}\rangle^S .$$

Note that the superposition ranges over Pauli errors of weight at most $\frac{3}{2}np$ as we are using the truncated distribution over Paulis. Therefore, conditioned on the event that the stabilizer code S is non-degenerate and has distance at least $d = 3np + 1$, the Knill-Laflamme error correction conditions imply that $\langle\overline{\psi_x}|\, E_a^\dagger E_b\, |\overline{\psi_y}\rangle = 0$, for any pair of codewords with $x, y \in \{0,1\}^k$, and thus

$$\mathsf{F}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^S), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^S)) = 0. \tag{54}$$

Further conditioning on the event that the implementation of the PGM succeeded, the probability of error due to the PGM mis-identifying the state (Equation (66)) is $e^{-np/12}$.

We can now put everything together to compute the final success probability of Algorithm 2, union bounding over all the three error sources:

1. S is degenerate.
2. Diamond-distance approximation error of implementing the PGM channel.
3. PGM measurement mis-identifies x.

We get that Algorithm 2 successfully outputs the correct x with probability at least $1 - \delta$, where

$$\delta \leq 3np \cdot 2^{nH(3p)} \cdot 3^{3np} \cdot 2^{-n+k} + \epsilon + e^{-np/12}. \tag{55}$$

This algorithm thus succeeds with constant probability for any noise rate p such that the term

$$3np \cdot 2^{nH(3p)} \cdot 3^{3np} \cdot 2^{-n+k} = \exp[\log(3np) + n(H(3p) + 3\log(3)p + k/n - 1)] \tag{56}$$

does not blow up. Noting that $k/n = O(1)$ in the definition of LSN, we can then check that as long as p is a constant that satisfies

$$H(3p) + 3\log_2(3)p + k/n < .99 - \frac{\log(1/\epsilon)}{n}, \tag{57}$$

this term vanishes exponentially in n and the total probability of error is $O(\epsilon)$. □

6.3 Multi-Shot Decoding Up to a Threshold

By taking more samples, we can slightly increase the noise rate at which decoding is still possible, but this only works up to a certain noise threshold that we also compute.

The algorithm is to run a modified PGM on a larger state space defined by the tensor product of the state spaces of all the samples. Note that we cannot appeal to Lemma 4.10 to argue that the algorithm and proof in the previous section carries over with the trivial replacement $n \leftarrow mn$, because the LSN problem on the larger state space corresponding to the MSLSN problem does not have the same distribution over stabilizers as in the canonical LSN.

Let the support of every sample define a *block*. For $P \in \mathcal{P}_{mn}$, let the *block support* of P (denoted $\mathsf{BSupp}(P)$) be the number of blocks on which P has at least one non-identity Pauli. We will call an error $E \in \mathcal{P}_{mn}$ *typical* if it has full block support (i.e. $\mathsf{BSupp}(E) = m$) and on each block its weight is between $[1, 3/2np]$. Let us call the set of all typical errors $\mathcal{P}_{typical}$.

Theorem 6.5 (Multi-shot decoding at higher noise)**.** *Let $n, k \in \mathbb{N}$ and $\epsilon \in (0,1)$. Let $\mathcal{D}_p^{\otimes n}$ be the n-qubit depolarizing channel, for some $p \in (0, 1/2)$ such that*

$$H(3p) + 3\log_2(3)p + k/n < .99 - \frac{\log(1/\epsilon)}{mn}. \tag{58}$$

Then, there exists a quantum algorithm which runs in time

$$\widetilde{\mathcal{O}}\left(n^2 m^2 \sqrt{2^{mk}\kappa_{\bar{\rho}}}(\kappa_{\bar{\rho}} + \min(\kappa_\rho, 2^{mk} \cdot \kappa_{\bar{\rho}}/\epsilon^2)\right)$$

and solves the $\mathsf{MSLSN}_{n,k,m,\mathcal{D}_p^{\otimes n}}$ problem (equivalently the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem with m samples) with probability at least $1 - O(\epsilon)$.

Proof. Suppose we are given as input an instance of the $\mathsf{MSLSN}_{n,k,\mathcal{D}_p^{\otimes n},m}$ problem, i.e.,

$$\left(\otimes_{i=1}^m S_i \in \mathsf{Stab}(n,k)^{\otimes m}, \mathcal{D}_p^{\otimes mn}(\otimes_{i=1}^m |\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i})\right)$$

where each $S_i \sim \mathsf{Stab}(n,k)$ describes a random stabilizer code and $x \sim [2^k]$ is a random element.

For a given MSLSN instance, define $T = \otimes_{i=1}^m S_i \in \mathsf{Stab}(mn, mk)$. Let us denote the true ensemble instance by

$$\rho_x^T \leftarrow \mathcal{D}_p^{\otimes mn}(\otimes_{i=1}^m |\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}). \tag{59}$$

We will run the PGM $\{(\tilde{\mathbf{\Lambda}}_x^T)\}_{x\in\{0,1\}^k}$ which is the PGM relative to the modified ensemble $\{(\tilde{\rho}_x^T)\}_{x\in\{0,1\}^k}$ where

$$\tilde{\rho}_x^T := \mathcal{D}^{typical}(\otimes_{i=1}^m |\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}) \tag{60}$$

That is, we approximate the true noise channel $\mathcal{D}_p^{\otimes mn}$ by a channel $\mathcal{D}^{typical}$ that still has tensor product structure across the blocks:

$$\mathcal{D}^{typical}(\rho) := \sum_{P\in\mathcal{P}_{typical}} p^{typical}(P) P\rho P^\dagger = \tilde{\mathcal{D}}^{(3/2np)\otimes m}(\rho). \tag{61}$$

This is precisely the channel on the Hilbert space of nm qubits where the truncated noise channel for the single-shot problem, $\tilde{\mathcal{D}}^{(3/2np)}$, is applied on each n-qubit block. As before, this is a good approximation of the distribution over the corresponding Pauli channels:

$$\left\| p^{typical} - \tilde{p}^{(mn)} \right\|_1 \leq m \left\| \tilde{p}^{(3/2np)} - \tilde{p}^{(n)} \right\|_1 \leq m \exp\left(-\frac{np}{12}\right), \tag{62}$$

where the first inequality follows from the fact that both $\tilde{p}^{(mn)}$ and $p^{typical}$ are product distributions, so we may use subadditivity of total variation distance over product distributions.

Next, we explicitly analyze the error probability of the (ideal) pretty good measurement given by $\{(\tilde{\mathbf{\Lambda}}_x^T)\}_{x\in\{0,1\}^k}$. Specifically, we find that

$$\max_{x\in\{0,1\}^k} 1 - \mathrm{tr}(\tilde{\mathbf{\Lambda}}_x^T \rho_x^T) \tag{63}$$

$$\leq m e^{-\frac{np}{12}} + \max_x \left(1 - \mathrm{tr}\left(\widetilde{\mathbf{\Lambda}}_x^T \tilde{\rho}_x^T\right)\right) \tag{64}$$

$$\leq m e^{-\frac{np}{12}} + \sum_{x\neq y} \sqrt{\mathsf{F}(\mathcal{D}^{typical}(\otimes_{i=1}^m |\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}), \mathcal{D}^{typical}(\otimes_{i=1}^m |\overline{\psi_y}\rangle\langle\overline{\psi_y}|^{S_i})} \tag{65}$$

$$\leq m e^{-\frac{np}{12}} + \sum_{x\neq y} \sqrt{\prod_{i=1}^m \mathsf{F}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^{S_i})} \tag{66}$$

where we have used the fact that fidelity is multiplicative across tensor product:

$$F\left(\rho_1 \otimes \rho_2, \sigma_1 \otimes \sigma_2\right) = F\left(\rho_1, \sigma_1\right) F\left(\rho_2, \sigma_2\right).$$

Call S_i "good" if has the property that

$$\mathsf{F}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^{S_i}) = 0, \quad \forall x, y. \tag{67}$$

Note that as long as at least one S_i is "good", the entire term

$$\sum_{x \neq y} \sqrt{\prod_{i=1}^{m} \mathsf{F}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^{S_i})} \tag{68}$$

vanishes, and

$$\max_{x \in \{0,1\}^k} 1 - \mathrm{tr}(\tilde{\mathbf{\Lambda}}_x^T \rho_x^T) \leq m e^{-\frac{np}{12}}. \tag{69}$$

This fortuitous event happens with probability

$$1 - P(\text{all } S_i \text{ are bad}) = 1 - (3/2np \cdot 2^{nH(3/2p)} \cdot 3^{3/2np} \cdot 2^{-n+k})^m. \tag{70}$$

Finally, accounting for the same three error sources as in the proof of Theorem 6.3, the total probability of failure of the whole algorithm is

$$\delta \leq \left(\frac{3np \cdot 2^{nH(3p)} \cdot 3^{3np}}{2^{n-k}}\right)^m + \epsilon + m e^{-np/12}. \tag{71}$$

So as long as

$$H(3p) + 3\log_2(3)p + k/n < .99 - \frac{\log(1/\epsilon)}{mn}, \tag{72}$$

and m is polynomial in n, the total probability of error is $O(\epsilon)$. □

While the above analysis shows that asking for more samples in MSLSN will increase the noise rate p at which the PGM still recovers the secret, there is a threshold level of noise at which taking more samples can never help:

Remark 6.6 (Noise threshold for PGM)**.** *Decoding, or error recovery, becomes even information-theoretically intractable at some threshold noise rate. Our PGM-based algorithms fail at noise rate p whenever*

$$H(3p) + 3\log_2(3)p + k/n > 1. \tag{73}$$

7 Worst-Case to Average-Case Reductions

In this section, we provide further evidence for the hardness of LSN by showing a worst-to-average-case reduction for a variant of the problem. Let us intuitively explain the importance of this. By definition, the worst-case instance within a problem class is harder to solve than all other instances. However, a worst-to-average-case reduction states that an algorithm that succeeds with high probability over a uniformly chosen instance within that class (i.e., succeeds in the average case) would also suffice to solve the worst-case instance within that class. The upshot is that "most" instances within that class are hard.

In complexity theory, such reductions are regarded as critical pieces of evidence that a problem is indeed as hard as conjectured. While a worst-to-average-case reduction for

LWE was already identified in [Reg05], LPN resisted similar attempts until the work of Brakerski, Lyubashevsky, Vaikuntanathan and Wichs [BLVW18] in 2018.

Our quantum reduction for LSN proceeds by entirely different means and uses a unitary "twirl" to randomize the secret, the code and the error all at once. Unfortunately, our reduction only applies to a variant of LSN where the average-case instance has some mild dependence on the worst-case instance. We leave it as an open question whether it is possible to reduce to a class of average-case problems that is completely independent of the worst-case instance.

7.1 Overview of the Reduction

Suppose we are presented with a worst-case LSN instance of the form

$$(S, E\,|\overline{\psi_x}\rangle^S) \tag{74}$$

where $S \in \mathsf{Stab}(n,k)$ is a stabilizer subgroup, $E \in \bar{P}_n$ is a Pauli error and $x \in \{0,1\}^k$ is a hidden secret—each potentially chosen adversarially. The goal of this section is to turn such a worst-case instance into an average-case instance of the LSN problem. Specifically, we will show how to draw a *re-randomizing* Clifford unitary $R \in \mathsf{Cliff}_n$, that simultaneously re-randomizes

- the underlying secret $x \in \{0,1\}^k$ of the instance, as well as
- the error E and the underlying stabilizer subgroup S of the instance.

In other words, our reduction applies R to the quantum part of the input, thereby obtaining a new (and ideally re-randomized) state of the form

$$RE\,|\overline{\psi_x}\rangle^S = E'\,|\overline{\psi_{x'}}\rangle^{S'}. \tag{75}$$

In the next sections, we show how to perform these steps separately. First, we show in Lemma 7.1 how to re-randomize the LSN secret. Next, in Lemma 7.2, we show how to re-randomize both the error and the underlying code. Finally, in Theorem 7.3 we put everything together and obtain the desired worst-case to average-case reduction.

7.2 Re-Randomization of the Secret

We now show how to sample a Pauli operator $P \in \mathcal{P}_n$ that allows us to re-randomize the secret which underlies the LSN sample.

For any stabilizer $S \in \mathsf{Stab}(n,k)$, we let $\mathcal{L}_X(S) = \{\overline{X^u}\}_{u\in\{0,1\}^k}$ denote the set of the logical Pauli X operators associated with S. While the choice of logical Paulis associated with a given stabilizer code S is not unique, we will use the prescription

$$\overline{P} = U^S_{\mathrm{Enc}} P (U^S_{\mathrm{Enc}})^\dagger, \quad \text{for } P \in \mathcal{P}_n. \tag{76}$$

We show that it suffices to apply a random Pauli in $\mathcal{L}_X(S)$ to re-randomize the secret.

Lemma 7.1 (Re-randomization of secret)**.** *Suppose that $(S, E\,|\overline{\psi_x}\rangle^S)$ is a fixed instance, for some stabilizer $S \in \mathsf{Stab}(n,k)$, error $E \in \bar{P}_n$ and secret $x \in \{0,1\}^k$. Let $u \sim \{0,1\}^k$ be a random string and let $\overline{X^u} \in \mathcal{L}_X(S)$ denote the logical Pauli X^u with respect to S. Then,*

$$\overline{X^u} E\,|\overline{\psi_x}\rangle^S = E\,|\overline{\psi_{x\oplus u}}\rangle^S. \tag{77}$$

Moreover, the distribution of $x \oplus u$ is now uniform over $\{0,1\}^k$.

Proof. First, we observe that $\overline{X^u}E\left|\overline{\psi_x}\right\rangle^S = \pm E\overline{X^u}\left|\overline{\psi_x}\right\rangle^S$ because Paulis either commute or anticommute. We adopt the convention of ignoring global phases, so we will evaluate $E\overline{X^u}\left|\overline{\psi_x}\right\rangle^S$ from now on. Calculating the action of $\overline{X^u}$ on the state, we find

$$EP\left|\overline{\psi_x}\right\rangle^S = E\overline{X^u}U^S_{\mathrm{Enc}}(|0^{n-k}\rangle \otimes |x\rangle) \tag{78}$$

$$= EU^S_{\mathrm{Enc}}\left((\mathbb{I}_{n-k} \otimes X^u)\,|0^{n-k}\rangle \otimes |x\rangle\right) \tag{79}$$

$$= EU^S_{\mathrm{Enc}}(|0^{n-k}\rangle \otimes |x \oplus u\rangle). \tag{80}$$

Because $u \sim \{0,1\}^k$ is random, it follows that $x \oplus u$ is also uniformly distributed for any fixed $x \in \{0,1\}^k$. This proves the claim. □

7.3 Re-Randomization of the Code and the Error

We now show how to re-randomize both the stabilizer S and the error E of a particular LSN instance. Note, however, there is an important subtlety; namely, the two cannot be randomized independently of each other. Acting with some unitary U on a given noisy codeword $E\left|\overline{\psi_x}\right\rangle^S$ re-randomizes the code S and the error E simultaneously. Moreover, as we will show in this section, the two are inherently correlated:

$$UE\left|\overline{\psi_x}\right\rangle^S = (UEU^\dagger)\left|\overline{\psi_x}\right\rangle^{USU^\dagger}.$$

Nevertheless, we are able to perform the desired re-randomization *approximately*. In the following, we use the notation $\mathcal{U}_w$ to denote the uniform distribution over the set of n-qubit weight-w Pauli operators. We now prove the following lemma.

Lemma 7.2 (Re-randomization of the code and error)**.** *Suppose that $(S, E\left|\overline{\psi_x}\right\rangle^S)$ is a fixed instance, for some stabilizer subgroup $S \in \mathsf{Stab}(n,k)$, error $E \in \bar{\mathcal{P}}_n$ and secret $x \in \{0,1\}^k$. Then, for any n-qubit unitary $U \in \mathcal{U}_n$, it holds that*

$$UE\left|\overline{\psi_x}\right\rangle^S = (UEU^\dagger)\left|\overline{\psi_x}\right\rangle^{USU^\dagger}.$$

Moreover, suppose that E has bounded weight $w = O(\log n)$. Then, the joint distribution of $(UEU^\dagger, USU^\dagger)$, for $U \sim \mathsf{PLC}_n$, is within total variation distance $1 - O(1/n^{\log n})$ of the product distribution $\mathcal{U}_w \times \mathrm{Unif}(\{USU^\dagger\}_{U\in\mathsf{PLC}_n})$ with support $\mathcal{P}_n \times \mathsf{Stab}(n,k)$.

Proof. To show the first claim, we observe that, for any $U \in \mathcal{U}_n$, we have

$$UE\left|\overline{\psi_x}\right\rangle^S = (UEU^\dagger)U\left|\overline{\psi_x}\right\rangle^S = (UEU^\dagger)\left|\overline{\psi_x}\right\rangle^{USU^\dagger}.$$

Next, we analyze the joint distribution of $(UEU^\dagger, USU^\dagger)$, for $U \sim \mathsf{PLC}_n$. We may regard $U : \mathcal{P}_n \times \mathsf{Stab}(n,k) \to \mathcal{P}_n \times \mathsf{Stab}(n,k)$ as a map that acts on the pair of initial error and stabilizer subgroup $(E,S) \in \mathcal{P}_n \times \mathsf{Stab}(n,k)$ via the group action

$$U * (E,S) := (UEU^\dagger, USU^\dagger). \tag{81}$$

We claim that for $U \sim \mathsf{PLC}_n$, the joint distribution of $(UEU^\dagger, USU^\dagger)$ is uniform over the PLC_n orbit of (E,S), namely the set

$$\mathsf{PLC}_n * (E,S) := \{(P,T) \in \mathcal{P}_n \times \mathsf{Stab}(n,k) \,:\, \exists\, U \in \mathsf{PLC}_n \text{ s.t. } P = UEU^\dagger, T = USU^\dagger\}. \tag{82}$$

To see this, note that for any $(P_1, T_1), (P_2, T_2) \in \mathsf{PLC}_n * (E, S)$, there must exist $V \in \mathsf{PLC}_n$ such that $VP_1V^\dagger = P_2$ and $VT_1V^\dagger = T_2$. This is because PLC_n is a group (Lemma 3.5), and by the definition of $\mathsf{PLC}_n * (E, S)$, there exist $V_1, V_2 \in \mathsf{PLC}_n$ such that

$$(P_1, T_1) = (V_1 E V_1^\dagger, V_1 S V_1^\dagger), (P_2, T_2) = (V_2 E V_2^\dagger, V_2 S V_2^\dagger).$$

Thus, V is exactly $V_2 V_1^\dagger$. As a result, we conclude that

$$\Pr_{U \sim \mathsf{PLC}_n}[UEU^\dagger = P_1 \cap USU^\dagger = T_1] \tag{83}$$

$$= \Pr_{U \sim \mathsf{PLC}_n}[VUEU^\dagger V^\dagger = VP_1V^\dagger \cap VUSU^\dagger V^\dagger = VT_1V^\dagger] \tag{84}$$

$$= \Pr_{U \sim \mathsf{PLC}_n}[VUEU^\dagger V^\dagger = P_2 \cap VUSU^\dagger V^\dagger = T_2] \tag{85}$$

$$= \Pr_{U \sim \mathsf{PLC}_n}[UEU^\dagger = P_2 \cap USU^\dagger = T_2]. \tag{86}$$

We will now argue that the uniform distribution $\mathsf{Unif}(\mathsf{PLC}_n * (E, S))$ is a good approximation to the distribution $\mathcal{U}_w \times \mathrm{Unif}(\{USU^\dagger\}_{U \in \mathsf{PLC}_n})$ by bounding the total variation distance between the two. First, note that the total variation distance between the uniform distribution on finite sets $\mathcal{X}$ and $\mathcal{Y} \subseteq \mathcal{X}$ takes the following simple form:

$$\delta_{\mathsf{TV}}(\mathrm{Unif}(\mathcal{X}), \mathrm{Unif}(\mathcal{Y})) = \frac{1}{|\mathcal{X}|}(|\mathcal{X}| - |\mathcal{Y}|) = 1 - \frac{|\mathcal{Y}|}{|\mathcal{X}|}. \tag{87}$$

Let $\mathcal{P}_w$ be the set of Paulis with Hamming weight w. Letting set $\mathcal{X}$ be the set of all tuples $\{(P, USU^\dagger)\}_{P \in \mathcal{P}_w, U \in \mathsf{PLC}_n}$ and set $\mathcal{Y} = \mathsf{PLC}_n * (E, S)$, we have that

$$\delta_{\mathsf{TV}}\left(\mathcal{U}_w \times \mathrm{Unif}(\{USU^\dagger\}_{U \in \mathsf{PLC}_n}), \mathsf{Unif}(\mathsf{PLC}_n * (E, S))\right) \tag{88}$$

$$= 1 - \frac{|\mathsf{PLC}_n * (E, S)|}{|\mathcal{P}_w| \cdot |\{USU^\dagger\}_{U \in \mathsf{PLC}_n}|} \tag{89}$$

$$\leq 1 - \frac{1}{|\mathcal{P}_w|} = 1 - \frac{1}{2\binom{n}{w}3^w}. \tag{90}$$

where we have used the fact that $|\mathsf{PLC}_n * (E, S)| \geq |\{USU^\dagger\}_{U \in \mathsf{PLC}_n}|$, because for every distinct $T \in \{USU^\dagger\}_{U \in \mathsf{PLC}_n}$, $\exists V \in \mathsf{PLC}_n$ such that $T = VSV^\dagger$, and so there is at least one element with $(VEV^\dagger, T) \in \mathsf{PLC}_n * (E, S)$. Plugging in the assumption that $w = O(\log n)$, we conclude that $2\binom{n}{w}3^w \leq 2\left(\frac{3ne}{w}\right)^w = O(n^{\log n})$, which proves the claim. □

7.4 Worst-Case to Average-Case Reduction

In this section, we formally state our worst-case to average-case reduction for a variant of the LSN problem. Specifically, we show how an appropriate average-case solver allows us to solve worst-case LSN instances $(S, E|\overline{\psi_x}\rangle^S)$, for a stabilizer subgroup $S \in \mathsf{Stab}(n, k)$, error $E \in \bar{\mathcal{P}}_n$ with $w := |E| = O(\log n)$ and secret $x \in \{0,1\}^k$. To this end, we assume that S is a fixed (and worst-case choice of stabilizer) and we consider the average-case problem $\mathsf{LSN}_{n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$ with respect to the following set of distributions:

- $\mathcal{N}$ is the uniform distribution $\mathcal{U}_w$ over n-qubit Pauli errors of weight precisely w.
- $\mathcal{S}$ is the uniform distribution $\mathrm{Unif}(\{USU^\dagger\}_{U \in \mathsf{PLC}_n}$ over stabilizers $\{USU^\dagger\}_{U \in \mathsf{PLC}_n}$.
- $\mathcal{I}$ is the uniform distribution over bit strings $x \in \{0,1\}^k$.

While our worst-case to average-case reduction only applies to a highly specialized variant (in particular, not to the standard variant) of LSN in the quasi-polynomial hardness regime[15], it nevertheless results in a meaningful reduction. Concretely, it allows us to solve a worst-case problem with inverse-quasi-polynomial success probability whenever we have a sufficiently good solver for an average-case version of the problem.

Theorem 7.3 (Worst-case to average-case reduction)**.** *Let $n, k \in \mathbb{N}$ with $n = \mathsf{poly}(k)$ and let $(S, E\,|\overline{\psi_x}\rangle^S)$ be any worst-case instance, for some stabilizer subgroup $S \in \mathsf{Stab}(n,k)$, error $E \in \bar{\mathcal{P}}_n$ of weight $w = O(\log n)$, and secret $x \in \{0,1\}^k$. Suppose there exists an algorithm $\mathcal{A}$ that runs in time T and solves the average-case problem $\mathsf{LSN}_{n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$ (implicitly depending on S) with probability $1-\epsilon$. Then, there exists an algorithm $\mathcal{B}$ which runs in time $\mathsf{poly}(k,T)$ and solves the worst-case instance $(S, E\,|\overline{\psi_x}\rangle^S)$ with probability at least $O(1/n^{\log n}) - \epsilon$.*

Proof. By assumption, the instance to the worst-case problem is of the form $(S, E\,|\overline{\psi_x}\rangle^S)$, for some stabilizer subgroup $S \in \mathsf{Stab}(n,k)$, error $E \in \bar{\mathcal{P}}_n$ of weight $w = O(\log n)$, and secret $x \in \{0,1\}^k$. We will now give a reduction $\mathcal{B}$ which transforms the given sample $(S, E\,|\overline{\psi_x}\rangle^S)$ into a new sample which approximates the average-case instance for the $\mathsf{LSN}_{n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$ problem. Our reduction $\mathcal{B}$ uses the solver $\mathcal{A}$ and proceeds as follows:

1. $\mathcal{B}$ samples a random logical operator $\overline{X^u} \sim \mathcal{L}_X(S)$, for $u \in \{0,1\}^k$.
2. $\mathcal{B}$ samples a random unitary $U \sim \mathsf{PLC}_n$ from the PLC_n ensemble.
3. $\mathcal{B}$ runs the solver $\mathcal{A}$ for the $\mathsf{LSN}_{n,k,\mathcal{N},\mathcal{S},\mathcal{I}}$ problem on input

$$(USU^\dagger, U\overline{X^u}E\,|\overline{\psi_x}\rangle^S)$$

   to obtain a string $x' \in \{0,1\}^k$. Then, the reduction $\mathcal{B}$ outputs $x' \oplus u$.

In other words, the reduction $\mathcal{B}$ applies the n-qubit re-randomizing unitary consisting of $R = U\overline{X^u}$ to the initial noisy codeword $E\,|\overline{\psi_x}\rangle^S$.

We now analyze the probability that $\mathcal{B}$ succeeds at successfully recovering the secret $x \in \{0,1\}^k$. First, we use our insights from Lemma 7.1 and Lemma 7.2 to argue that

$$(USU^\dagger, U\overline{X^u}E\,|\overline{\psi_x}\rangle^S) = (USU^\dagger, (UEU^\dagger)\,|\overline{\psi_{x\oplus u}}\rangle^{USU^\dagger}).$$

In addition, we know from Lemma 7.1 that the distribution of the secret is precisely $\mathcal{I}$, and we know from Lemma 7.2 that the distribution of the stabilizer subgroup and error of the resulting state is within total variation distance at most $1 - O(1/n^{\log n})$ of the product distribution $\mathcal{S} \times \mathcal{N}$. Therefore, by the strong convexity of the trace distance (Lemma 3.1), we know that $\mathcal{B}$ succeeds with probability at least $O(1/n^{\log n}) - \epsilon$.

We now argue that it takes $\mathsf{poly}(k,T)$ time to perform the reduction. First, we can invoke Lemma 3.8 to argue that a random logical Pauli can be computed in time $O(n^2)$. Then, from Lemma 3.8, it follows that the reduction $\mathcal{B}$ can compute a classical description of the stabilizer subgroup $USU^\dagger$ in time polynomial in k and n, since $U \sim \mathsf{PLC}_n$ is an efficient Clifford operator. Because $\mathcal{A}$ runs in time T, this completes the proof. □

While we do not explicitly carry out the proof, we remark that a similar worst-case to average-case reduction also applies to the multi-shot variant of LSN: the reduction can simply apply a fresh re-randomizing unitary for every block of n qubits.

[15] See Remark 4.6 for a definition of quasi-polynomial hardness for the LSN problem.

8 Complexity of Learning Stabilizers with Noise

In previous sections, we have postulated a *lower bound* for the time-complexity of LSN (Definition 4.1); that is, there do not exist efficient quantum algorithms that solve the problem. In this section, we upper-bound the complexity of LSN, placing it within the constellation of unitary synthesis problems proposed in [RY21, BEM+23]. Specifically, we show that the LSN problem, for a worst-case choice of non-degenerate code, is contained in $\mathsf{avgUnitaryBQP}^{\mathsf{avgUnitarySZK}_{\mathrm{HV}}}$, a (distributional and oracle) unitary complexity class which was recently defined by Bostanci et al. [BEM+23]. Classically, a worst-case variant of LPN was shown to be contained in $\mathsf{SearchBPP}^{\mathsf{SZK}}$ [BLVW18]; this can be seen as a quantum analog of that result for reasons we explain later.

Our complexity upper bound is based heavily on the recent work of Bostanci et al. [BEM+23] who studied the complexity of general decodable channels and its connection to the Uhlmann transformation problem. In this section, we give a self-contained proof which is tailored to towards our learning task.

8.1 A Review of Unitary Complexity

This subsection reprises some problems and complexity classes introduced in [RY21, BEM+23] as a way of giving background for our complexity upper bound.

Unitary synthesis problems. Many quantum problems whose output is a quantum state or unitary fall outside the purview of traditional complexity theory. Some examples include implementing Hamiltonian time evolution and state preparation tasks. All of these tasks have the flavor of *preparing a target unitary* upon input of some classical description of the target. This led to the formalization of *unitary synthesis problems*:

Definition 8.1 (Unitary synthesis problems)**.** *A unitary synthesis problem is given by a sequence* $\mathscr{U} = (U_x)_{x\in\{0,1\}^*}$ *of partial isometries.*[16]

We may understand $x \in \{0,1\}^*$ as the way that the particular target partial isometry is specified to the algorithm that solves the problem.

In the definition above, we call x the instance of the problem and U_x the transformation of $\mathscr{U}$ corresponding to x. The goal of an algorithm handed an instance x of a unitary synthesis problem is then to implement a quantum channel C_x which approximates a channel completion of the target unitary U_x in diamond norm. In fact the algorithm must accomplish this for all problem instances, $x \in \{0,1\}^*$. One could consider various metrics for how well the algorithm's output C_x approximates the target; a "worst-case" measure is to require the existence of a channel completion Φ_x of U_x such that

$$\|C_x - \Phi_x\|_\diamond \leq \delta(|x|) \quad \forall x \in \{0,1\}^*. \tag{91}$$

The strict requirement of diamond-norm approximation makes this a "worst-case" measure of closeness: it says there must exist a channel completion of the target unitary such that for any choice of registers to trace out, tracing out those registers of the channel completion still gives a channel that is well-approximated by the channel C_x after tracing out the same registers.

[16] See Section 3 for a formal definition of partial isometries.

Average-case/distributional unitary synthesis problems: It is not necessary for us to use this strict notion of approximation; we will instead be using an *average-case* notion of approximation captured by a *distributional (or average-case) unitary synthesis problem*. Here, in addition to a specifying a target partial isometry, we also specify an input state $|\psi_x\rangle$ and a register of this state on which the partial isometry is going to act, and we only care about closeness with respect to this register. We call the register that U_x (or its channel completion) acts on the *quantum input* to the unitary synthesis problem.

Definition 8.2 (Distributional unitary synthesis problem)**.** *A* unitary synthesis problem *is given by a sequence $\mathscr{U} = (U_x)_{x\in\{0,1\}^*}$ of partial isometries. We say that a pair $(\mathscr{U}, \Psi)$ is a* distributional unitary synthesis problem *if $\mathscr{U} = (U_x)_{x\in\{0,1\}^*}$ is a unitary synthesis problem with $U_x \in \mathrm{L}(\mathcal{H}_{\mathsf{A}_x}, \mathcal{H}_{\mathsf{B}_x})$ for some registers $\mathsf{A}_x\mathsf{B}_x$, and $\Psi = (|\psi_x\rangle)_{x\in\{0,1\}^*}$ is a family of bipartite pure states on registers $\mathsf{A}_x\mathsf{R}_x$. We call $|\psi_x\rangle$ the* distribution state *with* target register A_x *and* ancilla register R_x.

Definition 8.3 (Average-case implementation of distributional unitary synthesis)**.** *Let $(\mathscr{U}, \Psi)$ denote a distributional unitary synthesis problem, where $\mathscr{U} = (U_x)_{x\in\{0,1\}^*}$ and $\Psi = (|\psi_x\rangle)_{x\in\{0,1\}^*}$, and let $\epsilon : \mathbb{N} \to \mathbb{R}$ be a function. Let $C = (C_x)_{x\in\{0,1\}^*}$ denote a family of quantum circuits, where C_x implements a channel whose input and output registers are the same as those of U_x. We say that C* implements *$(\mathscr{U}, \Psi)$* with **average-case error** *ϵ if, for all sufficiently long $x \in \{0,1\}^*$, there exists a channel completion Φ_x of U_x such that*

$$\delta_{\mathsf{TD}}\Big((C_x \otimes \mathbb{I})(\psi_x),\ (\Phi_x \otimes \mathbb{I})(\psi_x)\Big) \leq \epsilon(|x|)\,,$$

where the identity channel acts on the ancilla register of $|\psi_x\rangle$.

Uhlmann Transformation as a canonical unitary synthesis problem. In this section, we introduce the Uhlmann transformation problem [BEM+23] as a canonical unitary synthesis problem which we then connect to the LSN problem. Let us first explain in words what this problem is. Given two circuits C, D acting on $2n$ qubits, the Uhlmann transformation problem is the problem "Synthesize a n-qubit unitary that approximately transforms the last n qubits of $|C\rangle$ into $|D\rangle$." In order to formally state this problem, however, we have to define what "approximately" means, and establish when there exists a unitary that achieves the desired approximation accuracy.

Definition 8.4 (Valid Uhlmann instances)**.** *We say that a string $x \in \{0,1\}^*$ is a* valid Uhlmann instance *if it encodes a tuple $(1^n, C, D)$ where C, D are explicit descriptions of* unitary *circuits that each act on $2n$ qubits. We further say that a valid Uhlmann instance x is a* fidelity-κ instance *if the reduced states ρ, σ of the states $|C\rangle = C|0^{2n}\rangle$, $|D\rangle = D|0^{2n}\rangle$ on the first n qubits satisfy $\mathrm{F}(\rho, \sigma) \geq \kappa$.*

The reason this defines a valid Uhlmann instance is given by Uhlmann's theorem (Theorem 3.2). Whenever x is a fidelity-κ instance, then the problem of synthesizing a unitary mapping $|C\rangle$ into a κ-approximation of $|D\rangle$ has a solution: there *must* exist a unitary U_B acting on the last n qubits of $|C\rangle_{AB}$ such that

$$\kappa = |\,\langle D|_{\mathsf{AB}}\,(\mathbb{I}_{\mathsf{A}} \otimes U_{\mathsf{B}})\,|C\rangle_{\mathsf{AB}}\,|^2. \tag{92}$$

In fact,

$$U = \mathrm{sgn}_\eta\,(\mathrm{Tr}_{\mathsf{A}}(|D\rangle\langle C|)) \tag{93}$$

is such a unitary. Here for $\eta \in \mathbb{R}$, $\mathrm{sgn}_\eta : \mathbb{C} \to \mathbb{C}$ is the threshold function that doesn't change inputs $|x| > \eta$ but maps inputs $|x| \leq \eta$ to 0; we extend it to a function on matrices in the usual way. We call U the "Canonical Uhlmann isometry".

Definition 8.5 (Uhlmann Transformation Problem)**.** *Let $\kappa, \eta : \mathbb{N} \to [0,1]$ be functions. The* (κ,η)-Uhlmann Transformation Problem *is the unitary synthesis problem given by* $\mathsf{Uhlmann}_{\kappa,\eta} = (U_x)_{x\in\{0,1\}^*}$ *where whenever x is a fidelity-$\kappa(n)$ Uhlmann instance specifying a pair (C,D) of unitary circuits that each act on $2n$ qubits for some n, then U_x is the Canonical Uhlmann isometry for the states $|C\rangle = C|0^{2n}\rangle$ and $|D\rangle = D|0^{2n}\rangle$, with U_x acting on the last n qubits.*

Otherwise if x is not a valid Uhlmann instance, then we define $U_x = 0$ (i.e., a partial isometry with zero-dimensional support).

Average-case/distributional Uhlmann transformation problem: Recall that to define average-case, or distributional, versions of worst-case unitary transformation problems we introduce a new element called the *distribution state*, a bipartite state which specifies a input state to the target unitary as well as the register of this state on which the synthesized unitary will be evaluated for accuracy. In this vein, to define the average case version of the Uhlmann Transformation Problem we specify a distribution state $|\psi_x\rangle_{AR}$ for every valid Uhlmann instance x. Recall that x encodes a tuple $(1^n, C, D)$ where C, D are explicit descriptions of *unitary* circuits that each act on $2n$ qubits; in that case, the distribution state $|\psi_x\rangle$ is also on $2n$ qubits.

Definition 8.6 (Distributional Uhlmann Transformation Problems)**.** *We define a state sequence $\Psi_{\mathsf{Uhlmann}} = (|\psi_x\rangle)_{x\in\{0,1\}^*}$ as follows: for all $x \in \{0,1\}^*$,*

$$|\psi_x\rangle = \begin{cases} |C\rangle & \text{if } x = (1^n, C, D) \text{ is a valid Uhlmann instance,} \\ 0 & \text{otherwise.} \end{cases}$$

Then, the distributional (κ,η)-Uhlmann Transformation Problem *is the distributional unitary synthesis problem* $\mathsf{DistUhlmann}_{\kappa,\eta} = (\mathsf{Uhlmann}_{\kappa,\eta}, \Psi_{\mathsf{Uhlmann}})$.

See Definition 8.5 for the definition of $\mathsf{Uhlmann}_{\kappa,\eta}$.

Unitary/state complexity classes. Unitary complexity problems can be grouped into complexity classes. These complexity classes are organized based on the amount of computational resources needed to perform state transformations.

As important background, let us informally introduce the unitary complexity class unitaryBQP. Analogous to BQP which is the set of all decision problems that can be solved by a polynomial-time quantum computer with at most $1/3$ probability of error, unitaryBQP is the set of all partial isometries that can be approximately applied in polynomial time in their description length. That is, it is the set of all sequences of unitary operators $(U_x)_{x\in\{0,1\}^*}$ where there is a polynomial-time quantum algorithm A that, given an instance $x \in \{0,1\}^*$ and a quantum system B as input, (approximately) applies U_x to system B. Here the input system B could contain any state, even part of a state on a larger system.

Average-case/distributional unitary complexity classes: We will eventually be interested in the average-case version of UnitaryBQP, which is avgUnitaryBQP. Recall that to go from worst-case to distributional, or average-case unitary synthesis problems, we introduce *distribution states* – input states whose specific registers we will be implementing the desired unitary transformation on. In order to properly define avgUnitaryBQP, therefore, we must introduce the state complexity class stateBQP which was introduced in [RY21]. Intuitively, this class contains sequences of quantum states that require polynomial time to be synthesized.

Definition 8.7 (stateBQP). *Let $\delta : \mathbb{N} \to [0,1]$ be a function. Then,* $\mathsf{stateBQP}_\delta$ *is the class of all sequences of density matrices $(\rho_x)_{x\in\{0,1\}^*}$ such that each ρ_x is a state on* $\mathsf{poly}(|x|)$ *qubits, and there exists a time-uniform family of general quantum circuits $(C_x)_{x\in\{0,1\}^*}$ such that, for all sufficiently long $x \in \{0,1\}^*$, the circuit C_x takes no inputs and C_x outputs a density matrix σ_x such that*

$$\delta_{\mathsf{TD}}(\sigma_x, \rho_x) \leq \delta(|x|)\,.$$

We define

$$\mathsf{stateBQP} = \bigcap_{c\in\mathbb{N}} \mathsf{stateBQP}_{n^{-c}}.$$

With this definition in hand, we can define avgUnitaryBQP as the set of polynomial-time solvable distributional unitary synthesis problems, with the restriction that their input state is in stateBQP (i.e. is polynomial-time preparable).

Definition 8.8 (avgUnitaryBQP). *Let $\epsilon : \mathbb{N} \to \mathbb{R}$ be a function. Define the unitary complexity class* $\mathsf{avgUnitaryBQP}_\epsilon$ *to be the set of distributional unitary synthesis problems $\left(\mathscr{U} = (U_x)_{x\in\{0,1\}^*}, \Psi = (|\psi\rangle_x)_{x\in\{0,1\}^*}\right)$ where $\Psi \in$ stateBQP and there exists a uniform polynomial-time quantum algorithm C that implements $(\mathscr{U}, \Psi)$ with* ***average-case error*** *ϵ.*

We define

$$\mathsf{avgUnitaryBQP} = \bigcap_{c\in\mathbb{N}} \mathsf{avgUnitaryBQP}_{n^{-c}}.$$

Average-case oracle unitary complexity classes. We also consider reductions between (average-case) variants of unitary complexity classes. Hence, we also make use of oracular variants of (average-case) unitary complexity classes, which behave analogously as their classical counterparts. We refer to Section 3.3 of [BEM+23] for more details. We will primarily be interested in the oracular complexity class $\mathsf{avgUnitarySZK}_{\mathsf{HV}}$, which is a unitary synthesis analog of the complexity class $\mathsf{QSZK}_{\mathsf{HV}}$ [Wat06].

Definition 8.9 ($\mathsf{avgUnitarySZK}_{\mathsf{HV}}$). *Let $c, s, \delta : \mathbb{N} \to [0,1]$ be functions. The interactive unitary synthesis class* $\mathsf{avgUnitarySZK}_{\mathsf{HV},c,s,\delta}$ *is the set of distributional unitary synthesis problems $(\mathscr{U}, \Psi)$ with $\mathscr{U} = (U_x)_{x\in\{0,1\}^*}$ and $\Psi = (|\psi_x\rangle)_{x\in\{0,1\}^*} \in$ stateBQP for which there exists a polynomial-time quantum verifier $V^* = (V_x^*)_{x\in\{0,1\}^*}$ (called the* honest verifier*), an unbounded prover P^* (called the* honest prover*), and a polynomial-time quantum algorithm* Sim *(called the* simulator*) such that for all sufficiently long $x \in \{0,1\}^*$:*

1. *(**Completeness**) P^* is accepted on instance x with probability at least $c(|x|)$, i.e.,*

$$\Pr[V_x^*(|\psi_x\rangle) \leftrightarrows P^* \text{ accepts}] \geq c(|x|)$$

   where V_x^* acts the identity on the ancilla register of $|\psi_x\rangle$.

2. *(**Soundness**) For all (possibly malicious) quantum provers P, there exists a channel completion Φ_x of U_x such that the following holds:*

$$\text{if} \quad \Pr[V_x^*(|\psi_x\rangle) \leftrightarrows P \text{ accepts}] \geq s(|x|) \qquad \text{then} \qquad \delta_{\mathsf{TD}}(\sigma, (\Phi_x \otimes \mathbb{I})(\psi_x)) \leq \delta(|x|)\,.$$

   where σ denotes the output of $V_x^*(|\psi_x\rangle) \leftrightarrows P$ conditioned on accepting and V_x^* acts the identity on the ancilla register of $|\psi_x\rangle$.

3. *(**Zero Knowledge**) There exists a negligible function $\epsilon : \mathbb{N} \to \mathbb{R}$ such that the simulator* Sim*, on input (x, j) (for $j \in \mathbb{N}$), outputs a state ρ satisfying*

$$\delta_{\mathsf{TD}}(\rho, \sigma_{x,j}) \leq \epsilon(|x|)$$

where $\sigma_{x,j}$ is the joint density matrix of both the verifier V_x^*'s private register and *the ancilla register of the input $|\psi_x\rangle$, immediately after the j'th round of interaction with the honest prover P^*.*

Finally, define

$$\mathsf{avgUnitarySZK}_{\mathsf{HV},\delta} = \bigcup_{\epsilon(n)\ \mathsf{negl}} \mathsf{avgUnitarySZK}_{\mathsf{HV},1-\epsilon,\frac{1}{2},\delta}\,,$$

where the union ranges over all negligible functions $\epsilon(n)$, and

$$\mathsf{avgUnitarySZK}_{\mathsf{HV}} = \bigcap_{c\in\mathbb{N}} \mathsf{avgUnitarySZK}_{\mathsf{HV},n^{-c}}\,.$$

We use the following theorem on the complexity of the Uhlmann transformation problem, which was formally shown in [BEM+23].

Theorem 8.10 ([BEM+23], Proposition 6.1)**.** $\mathsf{DistUhlmann}_{1-\mu}$ *is contained in the average-case interactive unitary synthesis class* $\mathsf{avgUnitarySZK}_{\mathsf{HV}}$ *for all negligible functions $\mu(n)$.*

8.2 Complexity Upper Bound

We conclude this section by showing that the average-case problem $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ from Definition 4.1—for a worst-case choice of non-degenerate Stabilizer code—is contained in the complexity class $\mathsf{avgUnitaryBQP}^{\mathsf{avgUnitarySZK}_{\mathrm{HV}}}$. Note that this requirement that the stabilizer code is non-degenerate is essentially without loss of generality: the quantum Gilbert-Varshamov bound guarantees that a random stabilizer code is non-degenerate and has large distance with overwhelming probability.

Our result can be seen as a quantum analog of the classical result which states that (a worst-case variant of) LPN is contained in $\mathsf{SearchBPP}^{\mathsf{SZK}}$ [BLVW18]. We remark that the non-degeneracy requirement in our theorem is similar to [BLVW18], where the authors assumed that the underlying code of the LPN instance is *balanced*.

In the following theorem and proof, the stabilizer code associated with Clifford $C \in \mathsf{Cliff}_n$ is assumed to be fixed and non-degenerate. When we refer to the "average-case" problem, we mean average-case over choice of error and secret. This is sound in the sense that the stabilizer code (and its accompanying encoding circuit C) is known to the LSN algorithm, whereas the error and secret are not.

Theorem 8.11 (Complexity of LSN)**.** *Let $k \in \mathbb{N}$ and $n \geq 8k$. Let $\mathcal{D}_p^{\otimes n}$ be the n-qubit depolarizing channel, for some $p \in (0, 0.05)$. Then, the average-case problem $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ with a worst-case choice of non-degenerate Stabilizer code with distance $d > 3np$ is contained in the complexity class* $\mathsf{avgUnitaryBQP}^{\mathsf{avgUnitarySZK}_{\mathrm{HV}}}$.

Proof. Let us fix $C \in \mathsf{Cliff}_n$ as the encoding Clifford for a worst-case non-degenerate Stabilizer code with distance $d > 3np$. As we have previously observed in Section 4.2, we can interpret the random (over E and s) instance

$$\left(C,\ E\,C(|0^{n-k}\rangle \otimes |s\rangle)\right) \quad \sim \quad \mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}} \tag{94}$$

as an instance of the Distributional Uhlmann transformation problem.

Specifically, the instance to the problem consists of $x = (1^k, Q^0, Q^1)$ and an input $|\Psi_x\rangle_{\mathsf{AB}}$ which denotes the 'starting state' of the Uhlmann transformation. (Q^0, Q^1) are a pair of quantum circuits that, upon input $|0^{2n}\rangle$ prepare the states

$$|Q^0\rangle_{\mathsf{AB}} = \sqrt{2^{-k}} \sum_x \sum_{E_a} \sqrt{\Pr_{E_a \sim \mathcal{D}_p^{\otimes n}}[E_a]} \left(|x\rangle \otimes |a\rangle \right)_{\mathsf{A}} \otimes \left(E_a \, C(|0^{n-k}\rangle \otimes |x\rangle) \otimes |0\rangle \right)_{\mathsf{B}}$$
$$|Q^1\rangle_{\mathsf{AB}} = \sqrt{2^{-k}} \sum_x \sum_{E_a} \sqrt{\Pr_{E_a \sim \mathcal{D}_p^{\otimes n}}[E_a]} \left(|x\rangle \otimes |a\rangle \right)_{\mathsf{A}} \otimes \left(|0^{n-k}\rangle \otimes |x\rangle \otimes |a\rangle \right)_{\mathsf{B}}.$$

while $|\Psi_x\rangle_{\mathsf{AB}} = |Q^0\rangle_{\mathsf{AB}}$. We note that $|Q^0\rangle_{\mathsf{AB}}$ is simply a purification of the register containing the quantum part of the input to LSN, i.e. Equation (94). Another way of saying this is that the ensemble of states in Equation (94) is the effective state if we trace out the A register of $|Q^0\rangle$.

Next, we argue that our choice of parameters ensures that the fidelity between the reduced states Q^0_{A} and Q^1_{A} on register A is near maximal. Indeed, Lemma 4.7 shows that for our specific choice of Clifford encoding circuit C, we have the fidelity guarantee that $\mathrm{F}(Q^0_{\mathsf{A}}, Q^1_{\mathsf{A}}) \geq 1 - 4 \cdot e^{-\frac{np}{24}}$. In other words, $x = (1^k, Q^0, Q^1)$ yields a valid fidelity-$\kappa(n)$ Uhlmann instance for $\kappa(n) = 1 - 4 \cdot e^{-\frac{np}{24}}$. Therefore, by Uhlmann's theorem, there exists a unitary $U \in \mathrm{L}(\mathcal{H}_{\mathsf{B}})$ which acts on the B register and maps $|Q^0\rangle_{\mathsf{AB}}$ to another state which achieves an overlap of at least $4 \cdot e^{-\frac{np}{24}}$ with $|Q^1\rangle_{\mathsf{AB}}$. In other words, the task of solving our given $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ instance amounts to synthesizing the Uhlmann unitary U on the density matrix representing the random input $E\, C(|0^{n-k}\rangle \otimes |s\rangle)$ in Equation (94). Because $x = (1^k, Q^0, Q^1)$ is a valid fidelity-$\kappa(n)$ DistUhlmann instance for $\kappa(n) = 1 - \mu(n)$, for $\mu(n) = 4 \cdot e^{-\frac{np}{24}}$, this immediately implies that any particular instance of the problem can be solved in $\mathsf{avgUnitaryBQP}^{\mathsf{DistUhlmann}_{1-\mu}}$.

Finally, we invoke Theorem 8.10 which states that $\mathsf{DistUhlmann}_{1-\mu}$ is contained in the average-case interactive unitary synthesis class $\mathsf{avgUnitarySZK}_{\mathsf{HV}}$ for all negligible functions $\mu(n)$. This yields the desired result.

□

9 Applications

We now discuss two applications which rest on the hardness of the LSN problem.

9.1 Learning From Quantum Data

The hardness of LPN implies conditional computational lower bounds on many classical learning tasks. Here we show that the hardness of LSN would also imply (quantum) computational lower bounds on the task of *learning from quantum data* [CGR+24], by studying a special case known as *learning state preparation processes*. As a bonus, we also provide an *upper* bound on the complexity of this task; specifically, we analyze it using the newly developed framework of unitary synthesis problems. We hope that this contribution paves the way for future work on the complexity of quantum learning tasks.

What is the quantum generalization of the hugely fruitful classical framework of Probably Approximately Correct (PAC) learning [Val84]? This question has received much attention recently [Car21, FQR24, CL21], culminating in the formulation of a powerful general framework known as 'learning from quantum data' by Ref. [CGR+24],

which encompasses the settings of PAC learning quantum states [Aar07], PAC learning from quantum examples [BJ95, AW17], variational quantum machine learning [BPP21] and so on.

Lower bounding the complexity of learning from quantum data. We lower bound the complexity of PAC-learning state-preparation processes, a special case of learning from quantum data. This implies a bound on the more general task. This setting is intended to model the scenario in which an experimentalist lacks significant control over the inputs to a process occurring in nature that she nevertheless wishes to understand. Here the process takes classical inputs (e.g. time, temperature, magnetic fields) to quantum states (e.g. electromagnetic radiation collected from astronomical events). The learner observes random input, output pairs $(x, \rho(x))$, and in particular cannot query the process at identical input points.

The learner's task is to output an estimate of ρ. To measure how far the learner's output is from the true ρ, we introduce the notion of *risk* (relative to ρ) for any process $h : \mathcal{X} \to L(\mathcal{H}_d)$:

$$R_\rho(h) := \mathbb{E}_{x\sim\mathcal{D}}[L(\rho(x), h(x))], \tag{95}$$

where $L : L(\mathcal{H}_d) \times L(\mathcal{H}_d) \to \mathbb{R}$ is the trace distance.

Definition 9.1 (Learning State Preparation processes)**.** *Let unknown process $\rho : \mathcal{X} \to L(\mathcal{H}_d)$ be a map that assigns to points in a classical input space $\mathcal{X}$ a corresponding quantum state, $\mathcal{C} = \{h : \mathcal{X} \to L(H)\}$ be a set of hypotheses for what ρ could be, and $\mathcal{D} : \mathcal{X} \to [0, 1]$ be a known distribution over the inputs.*

The Learning State Preparation processes problem is to output some hypothesis $h \in \mathcal{C}$ satisfying $R_\rho(h) \leq \epsilon$, given as input copies of a classical-quantum state σ,

$$\sigma = \mathop{\mathbb{E}}_{x\sim\mathcal{D}}[|x\rangle\langle x| \otimes \rho(x)]. \tag{96}$$

We say that the learner solves the problem of Learning State Preparation processes with sample complexity m if, given input $\sigma^{\otimes m}$, it succeeds with probability at least $1 - \delta$ to output a h satisfying Equation (95) where δ is constant.

Due to concentration, to output h minimizing the risk it suffices to minimize the empirical risk i.e., the average loss computed on the examples $(x_i, \rho(x_i))_{i=1}^n$:

$$\hat{R}_\rho(h) := \frac{1}{m}\sum_{i=1}^{m} L(\rho(x_i), h(x_i)).$$

The sample complexity of empirical risk minimization for Learning State Preparation processes was resolved by Ref. [FQR24]. They defined a quantum version of empirical risk minimization and showed that it constitutes a learning algorithm:

Theorem 9.2 (Quantum Empirical Risk Minimization (Theorem 3 of [FQR24]))**.** *There exists a learner for state preparation processes, which, for any ρ not necessarily within $\mathcal{C}$, with probability $1 - \delta$ outputs $\sigma^* \in \mathcal{C}$ that approximately minimizes the empirical risk:*

$$\hat{R}_\rho(\sigma^*) \leq 3 \min \hat{R}_\rho(\sigma_i) + 4\epsilon \tag{97}$$

with sample complexity

$$m = \tilde{O}\left(\frac{\log d \log \frac{1}{\delta} \max\left(\log \frac{|\mathcal{C}|}{\delta}, \log^2(e|\mathcal{C}|)\right)}{\epsilon^5}\right). \tag{98}$$

Quantum empirical risk minimization Theorem 9.2 is sample-efficient whenever $d = O(\exp(n))$ and $|\mathcal{C}| = O(\exp(n))$. However, the time complexity of learning was not addressed in Ref. [FQR24]. We now do so. In a nutshell, the following theorem says that an algorithm that could learn state-preparation processes can also decode exactly in the presence of noise.

Theorem 9.3 (Learning state preparation processes can be sample- but not time-efficient)**.** *Let $p \in (0, 1/2)$ such that*

$$H(3p) + 3\log_2(3)p + k/n < .99 - \frac{\log(3/2)}{n}. \tag{99}$$

Conditioned on the hardness of $\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$, there is no time-efficient algorithm for learning state preparation processes, even when only $1/poly(n)$ success probability is required and the unknown process ρ is guaranteed to be in the concept class $\mathcal{C}$ (the "proper" learning setting).

Proof. We observe that any algorithm for proper learning state preparation processes could also solve $\mathsf{MSLSN}_{m,n,k,\mathcal{D}_p^{\otimes n}}$, by choosing

$$\mathcal{X} := \{S\}_{S\in\mathsf{Stab}(n,k)} \qquad \text{(Classical input domain)} \tag{100}$$

$$\mathcal{D} := \mathsf{Unif}(\{S\}_{S\in\mathsf{Stab}(n,k)}) \qquad \text{(Distribution over inputs)} \tag{101}$$

$$\rho_z(S) := \mathcal{D}_p^{\otimes n}(U_{\mathrm{Enc}}^S(|z\rangle\langle z|)) \qquad \text{(Unknown map to be learned)} \tag{102}$$

$$\mathcal{C} := \{\rho_z\}_{z\in\{0,1\}^k} \qquad \text{(Concept class);} \tag{103}$$

Moreover ϵ in Definition 9.1 must be chosen so that the algorithm outputs the exact concept instead of a mere approximation; this is the meaning of decoding (solving MSLSN). ϵ must be such that no other concept lies in the ϵ-ball of the correct solution:

$$\epsilon \leq \min_{z_1,z_2\in\{0,1\}^k} \frac{1}{m}\sum_{i=1}^m \delta_{\mathrm{tr}}(\rho_{z_1}(S_i), \rho_{z_2}(S_i)), \tag{104}$$

and we proceed to show that a constant-ϵ learner for state preparation processes suffices. Using the convexity of trace distance (with an argument similar to that in the proof of Theorem 6.5), it suffices to tackle the case when the noise on each copy is given by the truncated depolarizing channel $\tilde{\mathcal{D}}^{(3/2np)}$ at the cost of an exponentially small increase in failure probability $me^{-\frac{np}{12}}$. Therefore it suffices to take

$$\epsilon \leq \min_{x,y\in\{0,1\}^k} \frac{1}{m}\sum_{i=1}^m \delta_{\mathrm{tr}}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^{S_i})) \tag{105}$$

$$\leq \min_{x,y\in\{0,1\}^k} \frac{1}{m}\sum_{i=1}^m \sqrt{1 - \mathsf{F}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^{S_i}))^2} \tag{106}$$

where the second inequality follows from Fuchs-van de Graf. We further recall from Theorem 3.10 that with probability at least $1 - \delta_1, \delta_1 = 3np \cdot 2^{nH(3p)} \cdot 3^{3np} \cdot 2^{-n+k}$ over choice of $S_i \in \mathsf{Stab}(n,k)$,

$$1 - \mathsf{F}(\tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_x}\rangle\langle\overline{\psi_x}|^{S_i}), \tilde{\mathcal{D}}^{(\frac{3np}{2})}(|\overline{\psi_y}\rangle\langle\overline{\psi_y}|^{S_i}))^2 = 1; \tag{107}$$

call S_i "good" if this is true. (The condition Equation (99) in the Theorem statement is necessary to make $\delta_1 \leq 1$.)

By Höffding's inequality, with probability $1 - \exp(-\delta_1^2 n/2)$, the number of samples featuring S_i that are "good" is at least $(1 - 3\delta_1/2)m$. Therefore any learner of state preparation processes up to any constant ϵ,

$$\epsilon \leq \frac{1}{m}((1 - 3\delta_1/2)m) = 1 - 3\delta_1/2, \tag{108}$$

will solve the MSLSN problem with a total failure probability of at most

$$\delta + \exp(-\delta_1^2 n/2) + \exp(-\frac{np}{12}) = \delta + O(\exp(-n)). \tag{109}$$

□

Remark 9.4. *Theorem 9.3 should be compared to the hardness result of Ref [AGS21]. There, the concept class to be learned is the set of quantum circuits that compute classical Boolean functions c output by* AC^0 *and* TC^0 *circuits, and in one of their learning models, the learner is given access to quantum examples of the form*

$$\sum_x \sqrt{\mathcal{D}(x)}\,|x, c(x)\rangle\,. \tag{110}$$

The goal is to output a hypothesis h such that $\Pr_{x\sim\mathcal{D}}[h(x) \neq c(x)] \leq \varepsilon$. *Ref [AGS21] showed that conditioned on the quantum hardness of* RingLWE *and* LWE, *no such polynomial-time learner can exist. This also implies the non-existence of polynomial-time quantum learners for general state-preparation processes, corresponding to the case when the unknown process is classical – maps to binary labels* $|0\rangle\langle 0|, |1\rangle\langle 1|$ *and is computable by a classical* TC^0 *or* AC^0 *circuit. To complete the reduction, note that measuring the first* $n - 1$ *qubits of a quantum example (Equation (110)) yields the learner's input (Equation (96)).*

Given that the setting of [AGS21] constrains quantum learners of classical functions, it does not say much about learning from 'natively quantum' data. For example, when the concept class consists of noise channels, it is unclear how to write their purifications as Boolean functions output by TC^0 *or* AC^0 *circuits. We expect* LSN *to be more useful to constrain learning in such situations.*

Learning state preparation processes: upper bound We also show an upper bound for the average-case complexity of proper learning state preparation processes, where here the average is over an unknown process ("concept") uniformly chosen from $\mathcal{C}$.

Specifically, we consider the task where the input to the learner is given by

$$\rho = \mathbb{E}_{\rho_c\sim\mathcal{C}}\mathbb{E}_{\mathbf{x}\sim\mathcal{D}^m}[|\mathbf{x}\rangle\langle\mathbf{x}| \otimes \rho_c(\mathbf{x})] \qquad \text{for} \qquad \rho_c(\mathbf{x}) = \otimes_{i=1}^m \rho_c(x_i). \tag{111}$$

where ρ_c is the unknown concept.

Our complexity upper bound holds as long as as particular condition on the class $\mathcal{C}$ is met. To make this more formal, we consider the following bipartite states

$$|Q^0\rangle = \frac{1}{\sqrt{|\mathcal{C}|}} \sum_{c\in\mathcal{C}} \sum_{\mathbf{x}\in\mathcal{X}^m} \sum_{i_\mathbf{x}} \sqrt{p(\mathbf{x}, i_\mathbf{x})}(|c\rangle \otimes |i_\mathbf{x}\rangle)_\mathsf{A} \otimes (|\mathbf{x}\rangle \otimes |\psi^\mathbf{x}_{c,i}\rangle)_\mathsf{B} \tag{112}$$

$$|Q^1\rangle = \frac{1}{\sqrt{|\mathcal{C}|}} \sum_{c\in\mathcal{C}} \sum_{\mathbf{x}\in\mathcal{X}^m} \sum_{i_\mathbf{x}} \sqrt{p(\mathbf{x}, i_\mathbf{x})}(|c\rangle \otimes |i_\mathbf{x}\rangle)_\mathsf{A} \otimes (|\mathbf{x}\rangle \otimes |c\rangle \otimes |i_\mathbf{x}\rangle \otimes |\bar{0}\rangle)_\mathsf{B}. \tag{113}$$

Here, the state $|Q^0\rangle$ can be thought of as a purification of the random learning instance in register B which is handed to the learner, and $|Q^1\rangle$ can be thought of as the purification of the exact target state (in a strong sense) which is to be output by the learner.

Definition 9.5 (Condition for learning state preparation processes)**.** *We require the following condition on our learning task; namely, the natural purifications which we defined in Equation* (112) *and Equation* (113) *must satisfy the condition that*

$$\mathrm{F}(\mathrm{Tr}_{\mathsf{B}}(|Q^0\rangle\langle Q^0|), \mathrm{Tr}_{\mathsf{B}}(|Q^1\rangle\langle Q^1|) \geq 1 - \kappa(n), \tag{114}$$

where $\kappa(n) = \mathsf{negl}(n)$ *is a negligible function.*

We believe it should be possible to similarly upper bound the complexity of other tasks of learning from quantum data in [CGR+24], but we leave this to future work.

Theorem 9.6 (Upper bound on the average-case complexity of proper learning classical-quantum processes)**.** *The problem of proper learning state preparation processes for a uniformly random concept from* $\mathcal{C}$ *is contained in the complexity class* $\mathsf{avgUnitaryBQP}^{\mathsf{avgUnitarySZK}_{\mathrm{HV}}}$.

Proof. Recall that the input to the learner is

$$\rho = \mathbb{E}_{\rho_c \sim \mathcal{C}} \mathbb{E}_{\mathbf{x} \sim \mathcal{D}^m} [|\mathbf{x}\rangle\langle\mathbf{x}| \otimes \rho_c(\mathbf{x})] \tag{115}$$

and

$$\rho_c(\mathbf{x}) = \otimes_{i=1}^m \rho_c(x_i), \tag{116}$$

where ρ_c is the unknown concept. We see now that $|Q^0\rangle_{AB}$ is a purification of the input to the learner Equation (115) which is held in register B. By Uhlmann's theorem, there exists a unitary $U \in \mathrm{L}(\mathcal{H}_{\mathsf{B}})$ which acts on the B register and maps $|Q^0\rangle_{\mathsf{AB}}$ to a κ-approximation of $|Q^1\rangle_{\mathsf{AB}}$ as long as the condition holds.

If the learner could synthesize a unitary that maps $|Q^0\rangle_{AB}$ to $|Q^1\rangle_{AB}$ *exactly*, the learner could apply that unitary on his input and measure the second subregister of register B, returning the *exact* concept c with probability 1. Given that the guarantee Equation (114) only implies approximate synthesis of $|Q^1\rangle_{AB}$, by the data-processing inequality for fidelity, applying the unitary whose existence is guaranteed by Uhlmann's theorem, and then measuring, outputs the *exact* concept c with probability at least $1 - \kappa$. Concluding similarly to the proof of Theorem 8.11, the problem of learning state preparation processes is contained in the complexity class $\mathsf{avgUnitaryBQP}^{\mathsf{avgUnitarySZK}_{\mathrm{HV}}}$. □

Remark 9.7. *This complexity upper bound could potentially be tightened by using the fact that for PAC learning, it is not necessary to output the* exact *unknown concept; it often suffices to output any concept that attains a risk (Equation* (95)*) of* ϵ*. Our upper bound pertains to the case when* $\epsilon = 0$*; it remains an open question if it is possible to get even tighter upper bounds if we do not require this.*

9.2 Quantum Bit Commitments

LPN is fundamental to classical cryptography. In this section, we show that our LSN assumption also has applications in quantum cryptography; we use it to construct a (statistically hiding and computationally binding) quantum bit commitment scheme.

Bit commitment is a fundamental primitive in cryptography with a multitude of applications, ranging from secure coin flipping, to zero-knowledge proofs, and secure computation. In classical cryptography, commitment schemes can be constructed from any one-way function [Nao03]. In quantum cryptography, potentially even weaker and inherently quantum assumptions suffice; for example, the existence of so-called pseudorandom states [Kre21, AQY22, MY22].

A (canonical) quantum bit commitment scheme [Yan22] is a pair of efficient quantum circuits (Q^0, Q^1) which output two registers: a "commitment register" C and a "reveal register" R. To commit to a bit $b \in \{0,1\}$, the sender prepares the state $|Q^b\rangle_{\mathsf{CR}} = Q^b |0\rangle_{\mathsf{CR}}$ and sends the register C to the receiver. In the "reveal phase", the sender simply reveals the bit b together with register R. The receiver accepts if the inverse unitary $Q^{b,\dagger}$ applied to registers CR yields $|0\rangle$ when measured in the computational basis. In terms of security, there are to important properties that we associate with a commitment scheme. First, the *statistical hiding* property ensures that the commitment register (information-theoretically) *hides* the committed bit b, i.e. after the commit phase, the receiver cannot guess what bit the sender committed to. Second, the *computational binding* property ensures that, after the commitment phase, it is computationally intractable for the sender to change the bit b they committed to.

We now give a formal definition.

Definition 9.8 (Quantum bit commitment)**.** *Let $\lambda \in \mathbb{N}$ denote the security parameter. A quantum bit commitment scheme is a uniform family of unitary quantum circuits $\{Q^b_\lambda\}_{\lambda\in\mathbb{N},b\in\{0,1\}}$ where for each λ, the circuits Q^0_λ, Q^1_λ act on $n = \mathsf{poly}(\lambda)$ qubits and output two registers C,R. The scheme consists of two separate phases:*

1. *(**Commit phase:**) to commit to a bit $b \in \{0,1\}$, the sender prepares the state*

$$|Q^b_\lambda\rangle_{\mathsf{RC}} = Q^b_\lambda |0^n\rangle$$

   and then sends the "commitment register" C to the receiver.

2. *(**Reveal phase:**) the sender announces the bit b and sends the "reveal register" R to the receiver. The receiver then accepts if performing the inverse unitary $Q^{b,\dagger}_\lambda$ on registers C,R and measuring in the computational basis yields the state $|0^n\rangle$.*

For security, we require that the following two properties hold:

- *(**Stat. Hiding:**) For every quantum algorithm $\mathcal{A}_\lambda$ with single-bit output, it holds that*

$$\left|\Pr\left[\mathcal{A}_\lambda(\rho^0_\lambda) = 1\right] - \Pr\left[\mathcal{A}_\lambda(\rho^1_\lambda) = 1\right]\right| \leq \mathsf{negl}(\lambda)\,,$$

  where $\rho_{\lambda,b}$ denotes the reduced density matrix of $|Q^b_\lambda\rangle$ on register C.

- *(**Comp. Binding:**) For every efficient quantum algorithm $\mathcal{A}_\lambda$ acting on R, it holds that*

$$\mathrm{F}\left(\left(\mathcal{A}_\lambda \otimes \mathbb{I}_{\mathsf{C}}\right)(|Q^0_\lambda\rangle\langle Q^0_\lambda|), |Q^1_\lambda\rangle\langle Q^1_\lambda|\right) \leq \mathsf{negl}(\lambda).$$

We now show the following theorem.

Theorem 9.9. *Let $\lambda \in \mathbb{N}$ denote the security parameter. Let $k \in \mathbb{N}$ and $n \geq 8k$ be integers which are polynomial in λ. Let $\mathcal{D}_p^{\otimes n}$ denote the n-qubit depolarizing channel, for $p = O(1)$. Let $U \sim \mathsf{Cliff}_n$ be a random Clifford. Consider the pair of quantum circuits (Q^0, Q^1) given by*

$$|Q^0\rangle_{\mathsf{CR}} = \sqrt{2^{-k}} \sum_x \sum_{E_a} \sqrt{\Pr_{E_a\sim\mathcal{D}_p^{\otimes n}}[E_a]} \left(|x\rangle \otimes |a\rangle\right)_{\mathsf{C}} \otimes \left(E_a\, U(|0^{n-k}\rangle \otimes |x\rangle) \otimes |0\rangle\right)_{\mathsf{R}}$$

$$|Q^1\rangle_{\mathsf{CR}} = \sqrt{2^{-k}} \sum_x \sum_{E_a} \sqrt{\Pr_{E_a\sim\mathcal{D}_p^{\otimes n}}[E_a]} \left(|x\rangle \otimes |a\rangle\right)_{\mathsf{C}} \otimes \left(|0^{n-k}\rangle \otimes |x\rangle \otimes |a\rangle\right)_{\mathsf{R}}.$$

Then, assuming the hardness of the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem, the pair (Q^0, Q^1) is a statistically hiding and computationally binding quantum bit commitment scheme.

Proof. From Lemma 4.7, we know that for $n \geq 8k$ and $p = O(1)$, e.g., $p = 0.04$, with overwhelming probability over the choice of the random Clifford U, it holds that

$$\delta_{\mathsf{TD}}(Q_{\mathsf{C}}^0, Q_{\mathsf{C}}^1) \leq \sqrt{1 - \mathrm{F}(Q_{\mathsf{C}}^0, Q_{\mathsf{C}}^1)} \leq 2 \cdot e^{-\frac{np}{48}} \leq \mathsf{negl}(\lambda).$$

This implies that (Q^0, Q^1) is a statistically hiding. The computational binding property follows immediately from the hardness of the $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ problem. This is because a successful adversary against the computational binding property would allow us to solve $\mathsf{LSN}_{n,k,\mathcal{D}_p^{\otimes n}}$ in polynomial-time with success probability at least $1/\mathsf{poly}(n)$.

□

References

[Aar07] Scott Aaronson. "The learnability of quantum states". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 463.2088 (2007), pp. 3089–3114. DOI: 10.1098/rspa.2007.0113 (cit. on p. 48).

[ABEM17] Dorit Aharonov, Michael Ben-Or, Elad Eban, and Urmila Mahadev. *Interactive Proofs for Quantum Computations*. 2017. arXiv: 1704.04487 [quant-ph]. URL: https://arxiv.org/abs/1704.04487 (cit. on pp. 4, 8).

[ABN23] Anurag Anshu, Nikolas P. Breuckmann, and Chinmay Nirkhe. "NLTS Hamiltonians from Good Quantum Codes". In: *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*. STOC 2023. Orlando, FL, USA: Association for Computing Machinery, 2023, pp. 1090–1096. ISBN: 9781450399135. DOI: 10.1145/3564246.3585114. URL: https://doi.org/10.1145/3564246.3585114 (cit. on p. 3).

[ACPS09] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. "Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems". In: *Proceedings of the 29th Annual International Cryptology Conference on Advances in Cryptology*. CRYPTO '09. Santa Barbara, CA: Springer-Verlag, 2009, pp. 595–618. ISBN: 9783642033551. DOI: 10.1007/978-3-642-03356-8_35. URL: https://doi.org/10.1007/978-3-642-03356-8_35 (cit. on p. 3).

[AF24] Victor V. Albert and Philippe Faist, eds. *The Error Correction Zoo*. 2024. URL: https://errorcorrectionzoo.org/ (cit. on p. 4).

[AGS21] Srinivasan Arunachalam, Alex Bredariol Grilo, and Aarthi Sundaram. "Quantum Hardness of Learning Shallow Classical Circuits". In: *SIAM J. Comput.* 50.3 (2021), pp. 972–1013. DOI: 10.1137/20M1344202. URL: https://doi.org/10.1137/20M1344202 (cit. on pp. 10, 50).

[AHI+17] Benny Applebaum, Naama Haramaty-Krasne, Yuval Ishai, Eyal Kushilevitz, and Vinod Vaikuntanathan. "Low-Complexity Cryptographic Hash Functions". In: *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*. Ed. by Christos H. Papadimitriou. Vol. 67. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017, 7:1–7:31. ISBN: 978-3-95977-029-3. DOI: 10.4230/LIPIcs.ITCS.2017.7. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITCS.2017.7 (cit. on p. 3).

[Ale03] Michael Alekhnovich. "More on Average Case vs Approximation Complexity". In: *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*. FOCS '03. USA: IEEE Computer Society, 2003, p. 298. ISBN: 0769520405 (cit. on p. 3).

[AQY22] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. "Cryptography from Pseudorandom Quantum States". In: *Advances in Cryptology – CRYPTO 2022*. Ed. by Yevgeniy Dodis and Thomas Shrimpton. Cham: Springer Nature Switzerland, 2022, pp. 208–236. ISBN: 978-3-031-15802-5. DOI: 10.1007/978-3-031-15802-5_8 (cit. on p. 51).

[AW17] Srinivasan Arunachalam and Ronald de Wolf. *A Survey of Quantum Learning Theory*. 2017. arXiv: 1701.06806 [quant-ph]. URL: https://arxiv.org/abs/1701.06806 (cit. on p. 48).

[BEM+23] John Bostanci, Yuval Efron, Tony Metger, Alexander Poremba, Luowen Qian, and Henry Yuen. *Unitary Complexity and the Uhlmann Transformation Problem*. 2023. arXiv: 2306.13073 [quant-ph]. URL: https://arxiv.org/abs/2306.13073 (cit. on pp. 4, 9, 11, 42, 43, 45, 46).

[BFKL94] Avrim Blum, Merrick Furst, Michael Kearns, and Richard J. Lipton. "Cryptographic Primitives Based on Hard Learning Problems". In: *Advances in Cryptology — CRYPTO' 93*. Ed. by Douglas R. Stinson. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 278–291. ISBN: 978-3-540-48329-8 (cit. on pp. 3, 10, 27).

[BHHP24] John Bostanci, Jonas Haferkamp, Dominik Hangleiter, and Alexander Poremba. *Efficient Quantum Pseudorandomness from Hamiltonian Phase States*. Cryptology ePrint Archive, Paper 2024/1639. 2024. URL: https://eprint.iacr.org/2024/1639 (cit. on p. 4).

[BJ95] Nader H. Bshouty and Jeffrey C. Jackson. "Learning DNF over the Uniform Distribution using a Quantum Example Oracle". In: *Proceedings of the Eigth Annual Conference on Computational Learning Theory, COLT 1995, Santa Cruz, California, USA, July 5-8, 1995*. Ed. by Wolfgang Maass. ACM, 1995, pp. 118–127. DOI: 10.1145/225298.225312. URL: https://doi.org/10.1145/225298.225312 (cit. on p. 48).

[BK00] H. Barnum and E. Knill. *Reversing quantum dynamics with near-optimal quantum and classical fidelity*. 2000. arXiv: quant-ph/0004088 [quant-ph]. URL: https://arxiv.org/abs/quant-ph/0004088 (cit. on pp. 7, 31).

[BKW00] Avrim Blum, Adam Kalai, and Hal Wasserman. *Noise-Tolerant Learning, the Parity Problem, and the Statistical Query Model*. 2000. arXiv: cs/0010022 [cs.LG]. URL: https://arxiv.org/abs/cs/0010022 (cit. on p. 5).

[BKW03] Avrim Blum, Adam Kalai, and Hal Wasserman. "Noise-tolerant learning, the parity problem, and the statistical query model". In: *J. ACM* 50.4 (July 2003), pp. 506–519. ISSN: 0004-5411. DOI: 10.1145/792538.792543. URL: https://doi.org/10.1145/792538.792543 (cit. on pp. 3, 10, 27).

[BLVW18] Zvika Brakerski, Vadim Lyubashevsky, Vinod Vaikuntanathan, and Daniel Wichs. *Worst-Case Hardness for LPN and Cryptographic Hashing via Code Smoothing*. Cryptology ePrint Archive, Paper 2018/279. https://eprint.iacr.org/2018/279. 2018. URL: https://eprint.iacr.org/2018/279 (cit. on pp. 3, 5, 7–10, 38, 42, 46).

[BMT78] E. Berlekamp, R. McEliece, and H. van Tilborg. "On the inherent intractability of certain coding problems (Corresp.)" In: *IEEE Transactions on Information Theory* 24.3 (1978), pp. 384–386. DOI: 10.1109/TIT.1978.1055873 (cit. on pp. 5, 10).

[BPP21] Leonardo Banchi, Jason Pereira, and Stefano Pirandola. "Generalization in Quantum Machine Learning: A Quantum Information Standpoint". In: *PRX Quantum* 2 (4 Nov. 2021), p. 040321. DOI: 10.1103/PRXQuantum.2.040321. URL: https://link.aps.org/doi/10.1103/PRXQuantum.2.040321 (cit. on p. 48).

[BV11] Zvika Brakerski and Vinod Vaikuntanathan. "Efficient Fully Homomorphic Encryption from (Standard) LWE". In: *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*. 2011, pp. 97–106. DOI: 10.1109/FOCS.2011.12 (cit. on p. 3).

[Car21] Matthias C. Caro. "Binary classification with classical instances and quantum labels". In: *Quantum Machine Intelligence* 3.1 (May 2021). ISSN: 2524-4914. DOI: 10.1007/s42484-021-00043-z. URL: http://dx.doi.org/10.1007/s42484-021-00043-z (cit. on pp. 9, 47).

[CGR+24] Matthias C. Caro, Tom Gur, Cambyse Rouzé, Daniel Stilck França, and Sathyawageeswar Subramanian. "Information-theoretic generalization bounds for learning from quantum data". In: *Proceedings of Thirty Seventh Conference on Learning Theory*. Ed. by Shipra Agrawal and Aaron Roth. Vol. 247. Proceedings of Machine Learning Research. PMLR, 30 Jun–03 Jul 2024, pp. 775–839. URL: https://proceedings.mlr.press/v247/caro24a.html (cit. on pp. 9, 47, 51).

[CL21] Kai-Min Chung and Han-Hsuan Lin. *Sample Efficient Algorithms for Learning Quantum Channels in PAC Model and the Approximate State Discrimination Problem*. 2021. arXiv: 1810.10938 [quant-ph]. URL: https://arxiv.org/abs/1810.10938 (cit. on pp. 9, 10, 47).

[CLLW16] Richard Cleve, Debbie Leung, Li Liu, and Chunhao Wang. "Near-linear constructions of exact unitary 2-designs". In: *Quantum Info. Comput.* 16.9–10 (July 2016), pp. 721–756. ISSN: 1533-7146 (cit. on p. 8).

[DDN14] Bernardo David, Rafael Dowsley, and Anderson C. A. Nascimento. "Universally Composable Oblivious Transfer Based on a Variant of LPN". In: *Cryptology and Network Security - 13th International Conference, CANS 2014, Heraklion, Crete, Greece, October 22-24, 2014. Proceedings*. Ed. by Dimitris Gritzalis, Aggelos Kiayias, and Ioannis G. Askoxylakis. Vol. 8813. Lecture Notes in Computer Science. Springer, 2014, pp. 143–158. DOI: 10.1007/978-3-319-12280-9_10. URL: https://doi.org/10.1007/978-3-319-12280-9%5C_10 (cit. on p. 3).

[Din06] Irit Dinur. "The PCP theorem by gap amplification". In: *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*. STOC '06. Seattle, WA, USA: Association for Computing Machinery, 2006, pp. 241–250. ISBN: 1595931341. DOI: 10.1145/1132516.1132553. URL: https://doi.org/10.1145/1132516.1132553 (cit. on p. 3).

[DS18] Yfke Dulek and Florian Speelman. *Quantum ciphertext authentication and key recycling with the trap code*. 2018. arXiv: 1804.02237 [quant-ph]. URL: https://arxiv.org/abs/1804.02237 (cit. on p. 4).

[ESM+07] Joseph Emerson, Marcus Silva, Osama Moussa, Colm Ryan, Martin Laforest, Jonathan Baugh, David G. Cory, and Raymond Laflamme. "Symmetrized Characterization of Noisy Quantum Processes". In: *Science* 317.5846 (Sept. 2007), pp. 1893–1896. ISSN: 1095-9203. DOI: 10.1126/science.1145699. URL: http://dx.doi.org/10.1126/science.1145699 (cit. on p. 8).

[FGKP06] Vitaly Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. "New Results for Learning Noisy Parities and Halfspaces". In: *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*. FOCS '06. USA: IEEE Computer Society, 2006, pp. 563–574. ISBN: 0769527205. DOI: 10.1109/FOCS.2006.51. URL: https://doi.org/10.1109/FOCS.2006.51 (cit. on p. 3).

[FQR24] Marco Fanizza, Yihui Quek, and Matteo Rosati. "Learning Quantum Processes Without Input Control". In: *PRX Quantum* 5 (2 June 2024), p. 020367. DOI: 10.1103/PRXQuantum.5.020367. URL: https://link.aps.org/doi/10.1103/PRXQuantum.5.020367 (cit. on pp. 9, 10, 47–49).

[FS96] Jean-Bernard Fischer and Jacques Stern. "An Efficient Pseudo-Random Generator Provably as Secure as Syndrome Decoding". In: *Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, Spain, May 12-16, 1996, Proceeding*. Ed. by Ueli M. Maurer. Vol. 1070. Lecture Notes in Computer Science. Springer, 1996, pp. 245–255. DOI: 10.1007/3-540-68339-9_22. URL: https://doi.org/10.1007/3-540-68339-9%5C_22 (cit. on p. 3).

[Gen09] Craig Gentry. "A fully homomorphic encryption scheme". AAI3382729. PhD thesis. Stanford, CA, USA, 2009. ISBN: 9781109444506 (cit. on p. 3).

[GIKL24a] Sabee Grewal, Vishnu Iyer, William Kretschmer, and Daniel Liang. *Efficient Learning of Quantum States Prepared With Few Non-Clifford Gates*. 2024. arXiv: 2305.13409 [quant-ph]. URL: https://arxiv.org/abs/2305.13409 (cit. on p. 11).

[GIKL24b] Sabee Grewal, Vishnu Iyer, William Kretschmer, and Daniel Liang. "Improved Stabilizer Estimation via Bell Difference Sampling". In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*. STOC 2024. Vancouver, BC, Canada: Association for Computing Machinery, 2024, pp. 1352–1363. ISBN: 9798400703836. DOI: 10.1145/3618260.3649738. URL: https://doi.org/10.1145/3618260.3649738 (cit. on p. 11).

[Gil52] Edgar N. Gilbert. "A comparison of signalling alphabets". In: *Bell System Technical Journal* 31 (1952), pp. 504–522. URL: https://api.semanticscholar.org/CorpusID:120421020 (cit. on pp. 3, 10).

[GL22] Aravind Gollakota and Daniel Liang. "On the Hardness of PAC-learning Stabilizer States with Noise". In: *Quantum* 6 (Feb. 2022), p. 640. ISSN: 2521-327X. DOI: 10.22331/q-2022-02-02-640. URL: http://dx.doi.org/10.22331/q-2022-02-02-640 (cit. on p. 11).

[GLM+22] András Gilyén, Seth Lloyd, Iman Marvian, Yihui Quek, and Mark M. Wilde. "Quantum Algorithm for Petz Recovery Channels and Pretty Good Measurements". In: *Phys. Rev. Lett.* 128 (22 June 2022), p. 220502. DOI: 10.1103/PhysRevLett.128.220502. URL: https://link.aps.org/doi/10.1103/PhysRevLett.128.220502 (cit. on p. 32).

[Got24] Daniel Gottesman. *Surviving as a Quantum Computer in a Classical World*. en. 2024-03 2024. URL: https://www.cs.umd.edu/class/spring2024/cmsc858G/QECCbook-2024-ch1-15.pdf (cit. on pp. 4, 15, 17–20).

[Got97] Daniel Gottesman. *Stabilizer Codes and Quantum Error Correction*. 1997. arXiv: quant-ph/9705052 [quant-ph]. URL: https://arxiv.org/abs/quant-ph/9705052 (cit. on p. 4).

[HB01] Nicholas J. Hopper and Manuel Blum. "Secure Human Identification Protocols". In: *Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*. ASIACRYPT '01. Berlin, Heidelberg: Springer-Verlag, 2001, pp. 52–66. ISBN: 3540429875 (cit. on p. 3).

[HH13] Daniel Harlow and Patrick Hayden. "Quantum computation vs. firewalls". In: *Journal of High Energy Physics* 2013.6 (June 2013). ISSN: 1029-8479. DOI: 10.1007/jhep06(2013)085. URL: http://dx.doi.org/10.1007/JHEP06(2013)085 (cit. on p. 4).

[HL11] Min-Hsiu Hsieh and Fran çois Le Gall. "NP-hardness of decoding quantum error-correction codes". In: *Phys. Rev. A* 83 (5 May 2011), p. 052331. DOI: 10.1103/PhysRevA.83.052331. URL: https://link.aps.org/doi/10.1103/PhysRevA.83.052331 (cit. on pp. 4, 5, 10).

[HP07] Patrick Hayden and John Preskill. "Black holes as mirrors: quantum information in random subsystems". In: *Journal of High Energy Physics* 2007.09 (Sept. 2007), p. 120. DOI: 10.1088/1126-6708/2007/09/120. URL: https://dx.doi.org/10.1088/1126-6708/2007/09/120 (cit. on p. 4).

[HRF23] Christoph Hirche, Cambyse Rouzé, and Daniel Stilck França. *Quantum Differential Privacy: An Information Theory Perspective*. 2023. arXiv: 2202.10717 [quant-ph]. URL: https://arxiv.org/abs/2202.10717 (cit. on p. 7).

[IP15] Pavithran Iyer and David Poulin. "Hardness of Decoding Quantum Stabilizer Codes". In: *IEEE Trans. Inf. Theor.* 61.9 (Sept. 2015), pp. 5209–5223. ISSN: 0018-9448. DOI: 10.1109/TIT.2015.2422294. URL: https://doi.org/10.1109/TIT.2015.2422294 (cit. on pp. 4, 5, 10, 11).

[JKPT12] Abhishek Jain, Stephan Krenn, Krzysztof Pietrzak, and Aris Tentes. "Commitments and Efficient Zero-Knowledge Proofs from Learning Parity with Noise". In: *Advances in Cryptology – ASIACRYPT 2012*. Ed. by Xiaoyun Wang and Kazue Sako. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 663–680. ISBN: 978-3-642-34961-4 (cit. on p. 3).

[JW05] Ari Juels and Stephen A. Weis. "Authenticating pervasive devices with human protocols". In: *Proceedings of the 25th Annual International Conference on Advances in Cryptology*. CRYPTO'05. Santa Barbara, California: Springer-Verlag, 2005, pp. 293–308. ISBN: 3540281142. DOI: 10.1007/11535218_18. URL: https://doi.org/10.1007/11535218_18 (cit. on p. 3).

[KK23] Upendra Kapshikar and Srijita Kundu. "On the Hardness of the Minimum Distance Problem of Quantum Codes". In: *IEEE Trans. Inf. Theor.* 69.10 (Oct. 2023), pp. 6293–6302. ISSN: 0018-9448. DOI: 10.1109/TIT.2023.3286870. URL: https://doi.org/10.1109/TIT.2023.3286870 (cit. on p. 11).

[KL12] Kao-Yueh Kuo and Chung-Chin Lu. "On the hardness of decoding quantum stabilizer codes under the depolarizing channel". In: *2012 International Symposium on Information Theory and its Applications*. 2012, pp. 208–211 (cit. on pp. 5, 10).

[Kre21] William Kretschmer. "Quantum Pseudorandomness and Classical Complexity". In: *16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021)*. Ed. by Min-Hsiu Hsieh. Vol. 197. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. ISBN: 978-3-95977-198-6. DOI: 10.4230/LIPIcs.TQC.2021.2 (cit. on pp. 4, 51).

[KSS10] Jonathan Katz, Ji Sun Shin, and Adam Smith. "Parallel and Concurrent Security of the HB and HB+ Protocols". In: *J. Cryptol.* 23.3 (July 2010), pp. 402–421. ISSN: 0933-2790. DOI: 10.1007/s00145-010-9061-2. URL: https://doi.org/10.1007/s00145-010-9061-2 (cit. on p. 11).

[Lyu05] Vadim Lyubashevsky. "The parity problem in the presence of noise, decoding random linear codes, and the subset sum problem". In: *Proceedings of the 8th International Workshop on Approximation, Randomization and Combinatorial Optimization Problems, and Proceedings of the 9th International Conference on Randamization and Computation: Algorithms and Techniques*. APPROX'05/RANDOM'05. Berkeley, CA: Springer-Verlag, 2005, pp. 378–389. ISBN: 3540282394. DOI: 10.1007/11538462_32. URL: https://doi.org/10.1007/11538462_32 (cit. on p. 7).

[Mah22] Urmila Mahadev. "Classical Verification of Quantum Computations". In: *SIAM J. Comput.* 51.4 (Jan. 2022), pp. 1172–1229. ISSN: 0097-5397. DOI: 10.1137/20M1371828. URL: https://doi.org/10.1137/20M1371828 (cit. on p. 3).

[McE78] Robert J. McEliece. *A Public-Key Cryptosystem Based on Algebraic Coding Theory*. DSN Progress Report 42-44. Pasadena, CA: Jet Propulsion Laboratory, 1978, pp. 114–116 (cit. on p. 3).

[Mon07] Ashley Montanaro. "On the Distinguishability of Random Quantum States". In: *Communications in Mathematical Physics* 273.3 (Mar. 2007), pp. 619–636. ISSN: 1432-0916. DOI: 10.1007/s00220-007-0221-7. URL: http://dx.doi.org/10.1007/s00220-007-0221-7 (cit. on pp. 7, 31, 32).

[MPSY24] Tony Metger, Alexander Poremba, Makrand Sinha, and Henry Yuen. *Simple constructions of linear-depth t-designs and pseudorandom unitaries*. 2024. arXiv: 2404.12647 [quant-ph]. URL: https://arxiv.org/abs/2404.12647 (cit. on p. 4).

[MY22] Tomoyuki Morimae and Takashi Yamakawa. "Quantum Commitments and Signatures Without One-Way Functions". In: *Advances in Cryptology – CRYPTO 2022*. Ed. by Yevgeniy Dodis and Thomas Shrimpton. Cham: Springer Nature Switzerland, 2022, pp. 269–295. ISBN: 978-3-031-15802-5. DOI: 10.1007/978-3-031-15802-5_10 (cit. on pp. 4, 11, 51).

[Nao03] Moni Naor. "On Cryptographic Assumptions and Challenges". In: *Advances in Cryptology – CRYPTO 2003*. Ed. by Dan Boneh. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 96–109. ISBN: 978-3-540-45146-4. DOI: 10.1007/978-3-540-45146-4_6 (cit. on p. 51).

[NC00] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000 (cit. on pp. 12, 13).

[PC08] David Poulin and Yeojin Chung. "On the iterative decoding of sparse quantum codes". In: *Quantum Info. Comput.* 8.10 (Nov. 2008), pp. 987–1000. ISSN: 1533-7146 (cit. on p. 10).

[PMH08] Ketan N. Patel, Igor L. Markov, and John P. Hayes. "Optimal synthesis of linear reversible circuits". In: *Quantum Info. Comput.* 8.3 (Mar. 2008), pp. 282–294. ISSN: 1533-7146 (cit. on pp. 5, 28).

[Reg05] Oded Regev. "On lattices, learning with errors, random linear codes, and cryptography". In: *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*. Ed. by Harold N. Gabow and Ronald Fagin. ACM, 2005, pp. 84–93. DOI: 10.1145/1060590.1060603. URL: https://doi.org/10.1145/1060590.1060603 (cit. on p. 38).

[Reg09] Oded Regev. "On lattices, learning with errors, random linear codes, and cryptography". In: *Journal of the ACM (JACM)* 56.6 (2009), pp. 1–40 (cit. on pp. 3, 11).

[RY21] Gregory Rosenthal and Henry Yuen. *Interactive Proofs for Synthesizing Quantum States and Unitaries*. 2021. arXiv: 2108.07192 [quant-ph]. URL: https://arxiv.org/abs/2108.07192 (cit. on pp. 42, 44).

[Sha79] Adi Shamir. "How to share a secret". In: *Commun. ACM* 22.11 (Nov. 1979), pp. 612–613. ISSN: 0001-0782. DOI: 10.1145/359168.359176. URL: https://doi.org/10.1145/359168.359176 (cit. on p. 3).

[Smi06] Graeme Stewart Baird Smith. "Upper and lower bounds on quantum codes". AAI3235592. PhD thesis. USA: California Institute of Technology, 2006. ISBN: 9780542893360 (cit. on pp. 4, 10, 20).

[Ste93] Jacques Stern. "A New Identification Scheme Based on Syndrome Decoding". In: *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*. Ed. by Douglas R. Stinson. Vol. 773. Lecture Notes in Computer Science. Springer, 1993, pp. 13–21. DOI: 10.1007/3-540-48329-2_2. URL: https://doi.org/10.1007/3-540-48329-2%5C_2 (cit. on p. 3).

[Uhl76] Armin Uhlmann. "The "transition probability" in the state space of a $*$-algebra". In: *Reports on Mathematical Physics* 9.2 (1976), pp. 273–279. ISSN: 0034-4877. DOI: 10.1016/0034-4877(76)90060-4 (cit. on p. 13).

[Val84] Leslie G. Valiant. "A Theory of the Learnable". In: *Proceedings of the 16th Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1984, Washington, DC, USA*. Ed. by Richard A. DeMillo. ACM, 1984, pp. 436–445. DOI: 10.1145/800057.808710. URL: https://doi.org/10.1145/800057.808710 (cit. on p. 47).

[Var64] R. R. Varshamov. "Estimation of number of signals in codes with correction of non-symmetric errors". In: *Avtomat. i Telemekh* 25.11 (1964), pp. 1628–1629. URL: https://www.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=at&paperid=11783&option_lang=eng (cit. on pp. 3, 10).

[Wat06] John Watrous. "Zero-knowledge against quantum attacks". In: *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*. Ed. by Jon M. Kleinberg. ACM, 2006, pp. 296–305. DOI: 10.1145/1132516.1132560. URL: https://doi.org/10.1145/1132516.1132560 (cit. on p. 45).

[WE16] Joel J. Wallman and Joseph Emerson. "Noise tailoring for scalable quantum computation via randomized compiling". In: *Physical Review A* 94.5 (Nov. 2016). ISSN: 2469-9934. DOI: 10.1103/physreva.94.052325. URL: http://dx.doi.org/10.1103/PhysRevA.94.052325 (cit. on p. 8).

[Wil13] Mark M. Wilde. *Quantum Information Theory*. Cambridge University Press, 2013 (cit. on pp. 4, 14).

[Yan22] Jun Yan. "General Properties of Quantum Bit Commitments (Extended Abstract)". In: *Advances in Cryptology – ASIACRYPT 2022*. Ed. by Shweta Agrawal and Dongdai Lin. Cham: Springer Nature Switzerland, 2022, pp. 628–657. ISBN: 978-3-031-22972-5. DOI: 10.1007/978-3-031-22972-5_22 (cit. on pp. 10, 52).

[YK17] Beni Yoshida and Alexei Kitaev. *Efficient decoding for the Hayden-Preskill protocol*. 2017. arXiv: 1710.03363 [hep-th]. URL: https://arxiv.org/abs/1710.03363 (cit. on p. 4).

[YZ21] Yu Yu and Jiang Zhang. "Smoothing Out Binary Linear Codes and Worst-Case Sub-exponential Hardness for LPN". In: *Advances in Cryptology – CRYPTO 2021: 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16–20, 2021, Proceedings, Part III*. Berlin, Heidelberg: Springer-Verlag, 2021, pp. 473–501. ISBN: 978-3-030-84251-2. DOI: 10.1007/978-3-030-84252-9_16. URL: https://doi.org/10.1007/978-3-030-84252-9_16 (cit. on pp. 3, 5, 7, 10).