On learning Pauli commuting local Hamiltonians

Yupan Liu*

School of Computer Science and Engineering, The Hebrew University of Jerusalem

October 6, 2020

Abstract

Learning a local Hamiltonian given samples from its Gibbs (thermal) state is a quantum analog of the classical problem known as learning graphical models or Boltzmann machines, which is a well-studied question in machine learning and statistics. In this note, we propose an algorithm for learning a Pauli commuting local Hamiltonian, namely a sub-class of commuting local Hamiltonians. In particular, our algorithm first conjugately applies a linear-depth Clifford circuit on given copies, then performs classical post-processing. Our approach is both sample-efficient and time-efficient under two specific conditions, which known results are either not working for commuting cases [BAL19], or only sample-efficient [AAKS20]. The proof utilizes the tableau representation of Paulis [AG04]. Our result enlightens on the problem of learning a general commuting local Hamiltonian with efficient time complexity.

1 Introduction

Learning graphical models, such as Boltzmann machines or Markov random fields, is one of the central problems in machine learning theory and statistical inference. Attempts to understand Boltzmann machines' learnability could date back to [AHS85] by Hinton et al. in the early 1980s. In recent years, significant progress emerges prominently in such fields – efficient provable learning algorithms for graphical models with optimal sample and time complexity, in particular, results regarding sparse and bounded-degree graphs [Bre15, VMLC16, KM17].

Learning Boltzmann machines: a classical example. Boltzmann machines are a specific sub-class of 2-wise Markov random fields, defined by *an interaction graph* such that each vertex i corresponds to a random variable x_i, and there is an edge between vertex i and j iff the coefficient associated with $x_i x_j$ is non-zero. Concentrating on this model, we could define a natural probability distribution as below:

$$\Pr\left[X = x\right] = \frac{1}{Z} \exp\left(\sum_{i \sim j} J_{ij} x_i x_j + \sum_i h_i x_i \right),$$

where J_{ij} and h_i are real coefficients, and the normalization factor Z is called the partition function. This probability distribution is also as known as the *Gibbs distribution*. We are aware of

*Email: yupan.liu@gmail.com

a series of results regarding Boltzmann machines' learnability, specifically, how to output an estimate of coefficients J_{ij} and h_i, utilizing given samples from the Gibbs distribution. Recent works, such as [Bre15, KM17, VMLC16], provide efficient classical algorithms for such a problem, viz., the running-time is a quadratic function of the number of vertices on the graph.

As learning graphical model problems from classical Hamiltonians play a crucial role in the learning theory with many known great results, an exciting connection arises from the interplay between these techniques and open problems in quantum computing and many-body physics. Quantum machine learning for quantum data signifies a new angle to resolve such open problems, which seems even more natural than quantum PAC learning [AdW17, AdW18] based on semi-quantum data. Could we provide provable efficient algorithms for such a learning local Hamiltonian problem?

Learning local Hamiltonians: previous results. There are several provable proposals for such a problem in the past. In [QR19, BAL19], the authors considered learning the Hamiltonian from local measurements. Their idea is to use a linear equation system determined by the outcomes of measuring the commutator between different candidate local terms on the Gibbs state. By solving this linear system, we can approximately recover the Hamiltonian corresponding to the given Gibbs state. Consequently, this approach managed to solve the Hamiltonian learning problem for non-commuting cases with the constraint matrix's non-vanishing spectral gap.

In addition, a sample-efficient algorithm utilizes *sufficient statistics*, which is based on the strong convexity of the quantum log-partition function, recently proposed by [AAKS20]. It is worth pointing out that their proof circumvents the quantum Markov property [HJPW04, BP12] (namely, conditional independence[1] of the Gibbs state), making their approach quite different from solutions to the classical problem. Whereas we cannot employ quantum Markov property generally, [APS19] suggests a time-efficient algorithm for learning 2-local commuting Hamiltonians, namely, efficiently reducing this problem to the problem of learning Markov random field. Utilizing the structure lemma in [BV05], they propose a locality-preserving constant-depth circuit mapping from a such a Hamiltonian to a classical Hamiltonian.

However, an algorithm for general Hamiltonian problem with efficient time complexity is still unknown.

1.1 Main result: learning a Pauli commuting local Hamiltonian

In this note, we propose an approach to learn a sub-class of commuting local Hamiltonians, namely, Pauli commuting k-local Hamiltonians. The famous Stabilizer codes inspire such a sub-class; notably, each local terms in the Hamiltonian is a tensor product of k Pauli matrix. We thus start from a formal definition of Pauli commuting local Hamiltonians (Pauli CLH).

Definition 1.1 (Pauli CLH). *We said that a k-local Hamiltonian $H(a) = \sum_{i=1}^{m} H_i(a_i)$ defined on n qubits is a Pauli commuting local Hamiltonian if the following holds:*

- $\forall i \in [m]$, $H_i(a_i) = a_i \sigma_1^{(i)} \otimes \cdots \otimes \sigma_n^{(i)}$ *where* $a_i \in \mathbb{R}$, $\sigma_j^{(i)} \in \{X, Y, Z, I\}$ *and* $|\{j : \sigma_j^{(i)} \neq I\}| \leq k$.
- $\forall i, j \in [m]$, $[H_i, H_j] = 0$.

With the help of the tableau representation of Paulis [AG04], we could construct a mapping from a quantum Hamiltonian to a classical Hamiltonian by applying a specific linear-depth Clifford

[1] Informally, partitioning a classical physical system associated with a Gibbs distribution p into three regions A, B, C where B shields A from C, then such a system satisfying *the Markov property* iff $p(A, C|B) = p(A|B)p(C|B)$. A quantum analog of such a decomposition property is more involved, and only knowns to hold for commuting local Hamiltonians [HJPW04, BP12].

circuit to given quantum Gibbs states. Such a mapping procedure corresponds to column-operation Gaussian elimination on the tableau. The resulted classical Hamiltonian will be local or sparse only under some conditions stated as Theorem 1.2.

Theorem 1.2. *Given* $\mathrm{poly}(n)$ *copies of the Gibbs state of a Pauli commuting local Hamiltonian defined on* n *qubits, one can recover the Hamiltonian by applying a nearest-neighbor linear-depth Clifford circuit and classical post-processing, under one of the following conditions:*

- *The tableau has full rank, namely, the number of local terms is at most the number of qubits, and all rows in the tableau are linearly independent;*
- *The tableau is sparse, i.e., the number of non-zero elements in each row and column is* $O(\log n)$*.*

The limitation of the approach presented in this note is that the resulting classical Hamiltonian's locality could be exponentially dependent on the Clifford circuit mapping's depth. Because we have to make either X block or Z block in the tableau be all zeros. Still, towards a practical quantum algorithm in the NISQ era, a provable shallow-depth quantum algorithm for learning Pauli commuting local Hamiltonians would be promising.

Paper organization. In Section 2, we will present our main algorithm. Section 3 concentrates on techniques for analyzing such an algorithm: the tableau representation of Pauli (Section 3.1), a procedure of learning local terms (Section 3.2), the mapping procedure (Section 3.3), and classical post-processing (Section 3.4). Section 4 will discuss our approach's limitations and some promising ideas to circumvent such obstacles.

2 Main algorithm

We hereby present the main algorithm that infers Theorem 1.2.

2.1 Algorithm overview

The algorithm's flow as described below in Figure 1.

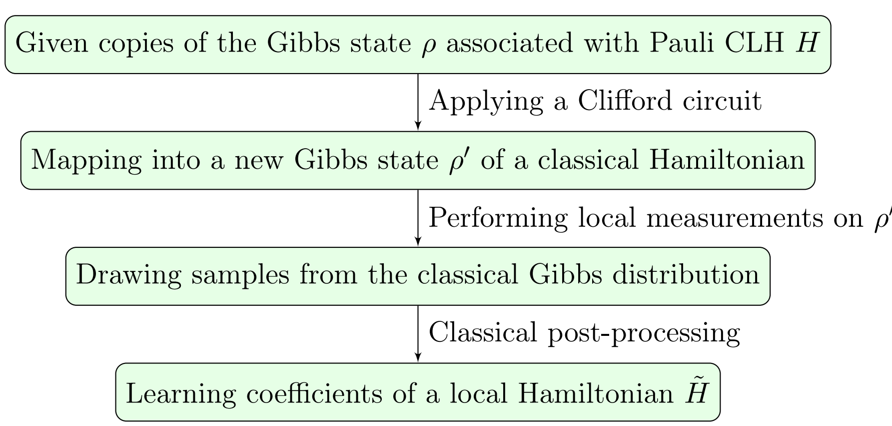

Figure 1: How to learn a Pauli commuting local Hamiltonian

To illuminate the intuition behind the main algorithm as Figure 1, consider the Hamiltonian H below and the Gibbs state ρ_H corresponded to it (setting $\beta := 1$ for simplicity),

$$H = a_1XX + a_2YY + a_3ZZ \text{ and } \rho_H = \exp(-H).$$

By applying two Clifford gates $\text{CNOT}_{1\to2}$ and H_1 on ρ_H, we observe that

$$\begin{aligned}\rho_{H'} &= (\text{H}_1\text{CNOT}_{1\to2})\,\rho_H\,(\text{H}_1\text{CNOT}_{1\to2})^\dagger \\ &= \exp(-a_1'ZI - a_2'ZZ - a_3'IZ).\end{aligned}$$

As the resulting local Hamiltonian H' is on Z basis (i.e., a classical Hamiltonian), one can draw samples from the corresponding Gibbs state $\rho_{H'}$ by performing $Z \otimes Z$ product measurements. Then, combining with a classical algorithm that learns a Markov random field [VMLC16, KM17], we have derived coefficients associated with H's local terms within desired errors. Then combining a classical algorithm for learning a Markov random field [VMLC16,KM17], we have derived coefficients associated with local terms in H within desired errors.

This illuminating example raises a natural question: *Could we extend this approach as Figure 1 to learn any Pauli commuting local Hamiltonian?*

2.2 Our main algorithm

We indeed could generalize this approach to the following algorithm:

(1) **Learning local terms.** By performing measurements of local observables constructed by the constraint matrix in [BAL19] on the Gibbs state ρ_H, one can learn local terms H_i such that its coefficient $|a_i| \geq \epsilon$, where ϵ is a small constant such as 10^{-3}. Since local observables here are tensor products of Paulis, it can be achieved by $O(\log n)$ measurement using overlapping tomography [CW20].

(2) **Mapping into a classical Hamiltonian.** By applying a Clifford circuit hinted by the tableau representation of Paulis [AG04], we result in a new Gibbs state $\rho_{H'}$ corresponding to a classical Hamiltonian H'. Owing to a recent result by Bravyi and Maslov [BM20], we only require a linear-depth nearest-neighbor Clifford circuit.

(3) **Drawing samples by local measurements.** As the resulting Hamiltonian H' is classical (but not necessarily local), one can draw m samples of the classical Gibbs distribution corresponding to H' by performing a product measurement on $Z^{\otimes n}$ on m copies of $\rho_{H'}$, where m is a polynomial of n under one of the conditions in Theorem 1.2.

(4) **Learning a classical Hamiltonian.** We consequently learn the resulting classical Hamiltonian H' by an algorithm that learns a Markov random field. This step probably is not efficient[2] except for the scenarios stated in Theorem 1.2.

(5) **Recovering a good approximation of coefficients associated with H.** Employing a solution of a linear equations system derived from both the mapping Clifford circuit and H', one could recover a good approximation of the coefficients of local terms in H.

It is worthwhile to mention that the tableau representation of Paulis [AG04] only works for commuting local Hamiltonians. Therefore it is required to ensure all local terms are indeed commuting at the first step.

[2] In a broad sense, it might take an exponential time because of either the non-locality of resulting Hamiltonian H' (such as [KM17]) or $O(n)$ degree on the interaction graph corresponding to H' due to an information-theoretic sample complexity lower bound proved in [SW12].

3 Proof techniques

3.1 Tableau representation of Paulis

The celebrated Gottesman-Knill theorem states that stabilizer states, namely the resulting state after applying a Clifford circuit on a computational basis state, are classically efficiently simulatable. Aaronson and Gottesman [AG04] provide a quadratic-time algorithm for simulating stabilizer states using a Boolean matrix representation of Pauli matrices. We thus recap their construction.

Overview of the tableau representation. Given an n-qubit Pauli commuting local Hamiltonian (see Definition 1.1), each local term $H_i = a_i \sigma_1^{(i)} \otimes \cdots \otimes \sigma_n^{(i)}$ is proportion to an n-fold tensor product of Paulis. We could represent such a tensor product of Paulis by a $(2n+1)$-tuple $(a_i, x_1^{(i)}, \cdots, x_n^{(i)}, z_1^{(i)}, \cdots, z_n^{(i)})$ where $\sigma_j^{(i)} = \mathbf{i}^{x_j^{(i)} z_j^{(i)}} X^{x_j^{(i)}} Z^{z_j^{(i)}}$. Such being the case, we could further represent a Pauli commuting local Hamiltonian by a tableau as Figure 2:

	a_i	$x_1^{(i)}$	$\cdots$	$x_n^{(i)}$	$z_1^{(i)}$	$\cdots$	$z_n^{(i)}$	
H_1	a_1	$x_1^{(1)}$	$\cdots$	$x_n^{(1)}$	$z_1^{(1)}$	$\cdots$	$z_n^{(1)}$	$\mathrm{Z}^{(1)}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	
H_m	a_m $\mathrm{X}^{(m)}$	$x_1^{(m)}$	$\cdots$	$x_n^{(m)}$	$z_1^{(m)}$	$\cdots$	$z_n^{(m)}$	
	p			X_n	Z_1			

Figure 2: Tableau representation of Paulis

Such a tableau of the tensor product of Paulis is referred to as *the tableau representation of Paulis*. Additionally, we denote this tableau (without the coefficients vector $\mathbf{p}$) by M_H.

Note the signed n-qubit Pauli group is closed under conjugation of Clifford circuits. This tableau representation indicates that applying a Clifford gate, such as Hadamard (H), Phase (S), and Controlled NOT (CNOT), conjugately on a tensor product of Paulis is correspondent to applying a column operation on this tableau, as summarized in Figure 3, where the left block is associated with Pauli Xs; and the right block corresponds to Pauli Zs.

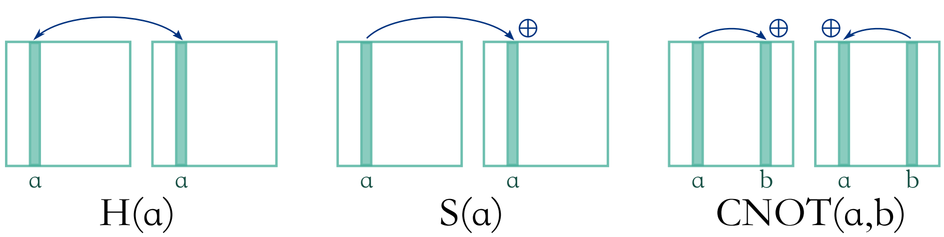

Figure 3: The correspondence between Clifford gates and column operations on the tableau

Useful lemmas. We now then prove three useful lemmas regarding the tableau representation. For simplicity, we shall utilize the following vectors from the tableau (see Figure 2). Let $\mathbf{p} := (a_1, \cdots, a_m)$ be the coefficient vector on the tableau. Let $\mathbf{X}_i := \left(x_i^{(1)}, \cdots, x_i^{(m)}\right)$ be the vector corresponds to the i-th column, and let $\mathbf{X}^{(j)} := \left(x_1^{(j)}, \cdots, x_n^{(j)}\right)$ be the vector associated with the

j-th row. Likewise, the vectors $\mathbf{Z}_i$ and $\mathbf{Z}^{(j)}$ correspond to the Z-block on the tableau and are defined similarly.

Lemma 3.1 follows from a direct calculation.

Lemma 3.1 (Commutation criterion)**.** *For any $i \neq j \in [m]$, two local terms H_i and H_j are commuting, i.e. $[H_i, H_j] = 0$ if and only if $\mathbf{X}^{(i)} \cdot \mathbf{Z}^{(j)} \oplus \mathbf{X}^{(j)} \cdot \mathbf{Z}^{(i)} = 0$.*

Lemma 3.2 (Tableau's rank upper bound)**.** *For any k-local Pauli CLH $H = \sum_{i=1}^{m} a_i P_i$ on n qubits,* $\mathrm{rank}(M_H) \leq n$*.*

Proof. Notice that all rows in M_H are linearly independent if and only if these rows form a set of a stabilizer's generators. For simplicity, each P_i is needless local. For the first row P_1, there are $4^n - 1$ choices because this row cannot be an identity. The second row P_2 must commute with P_1 and cannot be either identity or P_1; Due to Lemma 3.1 and the fact that P_2 cannot be any operator in the group generated by identity and P_1, there are $4^n/2$ terms commutes with P_1. Inductively, the k-th row P_k have to commute with $\{P_i\}_{1 \leq i < k}$ and cannot be generated by $\{I, P_1, \cdots, P_{k-1}\}$; hence, there are $4^n/2^k - 2^k$ choices of such row. It is straightforward to see that $4^n/2^k - 2^k \geq 0$ if and only if $0 \leq k \leq n$, which infers $\mathrm{rank}(M_H) \leq n$. □

Finally, we prove the correspondence indicated in Figure 3.

Lemma 3.3 (Correspondence between Clifford gates and column operations)**.** *A Pauli commuting local Hamiltonian H is closed under the conjugation of any Clifford circuit U, namely, the resulting Hamiltonian $UHU^\dagger$ remains to be a Pauli CLH. Moreover, each Clifford gate (such as Hadamard, Phase, Controlled NOT) corresponds to a column operation on the tableau matrix M_H.*

Proof. Recall the conjugation relation between Paulis X, Y, Z and Clifford gates H, S, CNOT:

$$HXH^\dagger = Z, HYH^\dagger = -Y, HZH^\dagger = X;$$
$$SXS^\dagger = Y, SYS^\dagger = X, SZS^\dagger = Z;$$
$$\mathrm{CNOT}_{1\to 2}(X^{a_1}Z^{b_1}) \otimes (X^{a_2}Z^{b_2})\mathrm{CNOT}_{1\to 2}^\dagger = (-1)^{a_1 b_2 (a_2 \oplus b_1 \oplus 1)}(X^{a_1}Z^{b_1 \oplus b_2}) \otimes (X^{a_1 \oplus a_2}Z^{b_2}).$$

Remark that a Clifford gate applying conjugately on a tensor-product of Paulis could only change each entry's sign in the coefficient vector $\mathbf{p}$. We thus have derived the correspondence between Clifford gates and column operation on the tableau matrix M_H:

- $\mathrm{CNOT}_{i\to j}$ (i-th qubit: control; j-th qubit: target):

$$\mathbf{p}' := \mathbf{p} \oplus (\mathbf{X}_i \odot \mathbf{Z}_j \odot (\mathbf{X}_j \oplus \mathbf{Z}_i \oplus \mathbf{1}))\,;\ \mathbf{X}'_j := \mathbf{X}_i \oplus \mathbf{X}_j \text{ and } \mathbf{Z}'_i := \mathbf{Z}_i \oplus \mathbf{Z}_j.$$

- H on i-th qubit: $\mathbf{p}' := \mathbf{p} \oplus (\mathbf{X}_i \odot \mathbf{Z}_i)$ and swap $\mathbf{X}_i$ with $\mathbf{Z}_i$.
- S on i-th qubit: $\mathbf{p}' := \mathbf{p} \oplus (\mathbf{X}_i \odot \mathbf{Z}_i)$ and $\mathbf{Z}'_i := \mathbf{X}_i \oplus \mathbf{Z}_i$.

Notice $\odot$ denotes entry-wise multiplication between two vectors. It is apparent to verify that Clifford operations acting on a Pauli CLH preserve the commutation criterion in Lemma 3.1. □

As a remark, combining Lemma 3.2 and Lemma 3.3, we are able to perform a Gaussian elimination using only column operations on the tableau matrix. It is crucial for the mapping procedure in Section 3.3.

3.2 Learning local terms by local measurements

We exhibit below an efficient algorithm for learning *commuting* local terms associated with given Gibbs states using only local measurement. As in [BAL19], the starting point is the fact $\forall n, \mathrm{Tr}(\mathbf{i}\rho[H_L, A_n]) = 0$, where H_L is a Hamiltonian associated with a local patch L and $\{A_n\}$ is a set of all local observables inside the local patch L. By rewriting H_L as $H_L = \sum_{m=1}^{M} c_m S_m$, we have derived a constraint for the Gibbs state ρ:

$$\forall A_n, \sum_{m=1}^{M} c_m \mathrm{Tr}(\mathbf{i}\rho[S_m, A_n]) = 0.$$

In the scenario of k-local Pauli CLH, the set of all possible S_m consists of all n-fold tensor products of Paulis $\sigma_1^{(m)} \otimes \cdots \sigma_n^{(m)}$ such that the number of non-identity entries is at most k. Likewise, we could obtain the set $\{A_n\}$ of local observables inside L. Because of the cyclicity $\mathrm{Tr}(A[B,C]) = \mathrm{Tr}(C[A,B])$, we further infer $\mathrm{Tr}(\mathbf{i}\rho[S_m, A_n]) = \mathrm{Tr}(\mathbf{i}A_n[\rho, S_m])$. Obviously, if S_m is indeed a local term in H, then $[\rho, S_m] = 0$ indicates that $\forall n, \mathrm{Tr}(\mathbf{i}\rho[S_m, A_n]) = 0$ because S_m is supposed to commute with all other possible local terms.

Let the constraint matrix K be $(K)_{mn} := \mathrm{Tr}(\mathbf{i}\rho[S_m, A_n])$, which is defined in [BAL19], we summarize this construction as Figure 4:

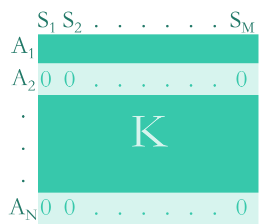

Figure 4: Learning local terms by constructing the constraint matrix

Informally, the algorithm learns all local terms by finding all-zero rows in the constraint matrix K, where outcomes of local measurements constitute the constraint matrix concerning desired local observables. We henceforth write this procedure of learning local terms formally below as Lemma .

Lemma 3.4 (Learning local terms by local measurements)**.** *Given* $\mathrm{poly}(n)$ *copies of the Gibbs state* ρ_H *corresponding to an* n*-qubit Pauli commuting local Hamiltonian* H*, an efficient algorithm for learning the candidate set* $\mathcal{S}$ *consists of all* k*-local tensor products of Paulis generated by the stabilizer group of* H*. Moreover, let the candidate Hamiltonian* H' *be* $H' := \sum_{S_i \in \mathcal{S}} a_i S_i$*, where* $\{a_i\}_{S_i \in \mathcal{S}}$ *are unknown coefficients.*

Proof. We first provide an algorithm that learns possible local terms in H:

(1) For all commutators $[S_m, A_n]$, where S_m and A_n are k-local tensor products of Paulis on n qubits, measuring $2k$-local observables $\mathbf{i}[S_m, A_n]$ provided $[S_m, A_n] \neq 0$; otherwise, let $(K)_{mn} := \mathrm{Tr}(\mathbf{i}\rho[S_m, A_n]) = 0$.

(2) For each k-local tensor product of Paulis S_m on n-qubit, adding S_m into the candidate set $\mathcal{S} = \mathcal{S} \cup \{S_m\}$ if $\mathrm{Tr}(\mathbf{i}\rho[S_m, A_n]) = 0$ for any n.

(3) For each $S_m \in \mathcal{S}$, remove S_m from the candidate set $\mathcal{S}$ if S_m is not commuting with all local terms in $\mathcal{S}$.

Let $\hat{\mathcal{S}}$ be the candidate set of local terms after the step (3).

It remains to prove the correctness of such an algorithm. Owing to $\mathrm{Tr}(\mathbf{i}\rho[S_m, A_n]) = \mathrm{Tr}(\mathbf{i}A_n[\rho, S_m])$, we obtain $[\rho, S_m] = 0$ if S_m is a local term in H. Such equality hints that $\mathrm{Tr}(\mathbf{i}A_n[\rho, S_m]) = 0$ holds for any local observable A_n inside the local patch L. The set of all such local terms S_m, as stated above, is candidate set $\mathcal{S}$ of local terms at the step (2).

It is evident that all local terms in a Pauli CLH commute; however, these local terms found by step (2) of the algorithm probably not commute. Hence, an additional step as the step (3) is required. Specifically, we have to remove local terms in $\mathcal{S}$ that is not commuting with other local terms in $\mathcal{S}$. Still, we cannot get rid of local terms that is a product of local terms in H. $\mathcal{S}$ thus is a set of all k-local tensor product of Paulis in the stabilizer associated with the Hamiltonian of given Gibbs state ρ. Remark that the number of n-qubit tensor product of Paulis with locality k is $\mathrm{poly}(n)$. We immediately deduce such an algorithm requires the expectation value of $\mathrm{poly}(n)$ local observables. One can calculate such expectation values using the result of local measurements, so the classical post-processing is efficient. □

Furthermore, employing perfect hash families mentioned in [CW20], we could further reduce the number of product measurements from $\mathrm{poly}(n)$ to $\log(n)$. In particular, as the commutator $[A_n, S_m]$ is a $2k$-local tensor product of Paulis[3], we could utilize the overlapping tomography scheme [CW20].

3.3 Mapping a Pauli CLH into a classical Hamiltonian

We now provide a procedure that maps a Pauli commuting local Hamiltonian H into a classical Hamiltonian H' by applying a nearest-neighbor linear-depth Clifford circuit. Such mapping produce is essentially performing *a Gaussian elimination on the tableau matrix's X block* (or Z block), and then resulting Hamiltonian is clearly classical.

For simplicity, we would first assume that the associated tableau M_H has rank n, then we will show an efficient procedure to construct a full-rank variant of M_H by adding additional local terms in Hamiltonian.

Mapping procedure. Consider a Pauli commuting LH $H = \sum_{i=1}^{m} a_i H_i$, which can be represented as the coefficient vector $\mathbf{p} \in \mathbb{R}^m$ and the tableau $M_H \in \mathbb{F}_2^{2n\times m}$. Employing the correspondence between Clifford gates and column operations on the tableau, namely Lemma 3.3, we can mapping H into a classical Hamiltonian on the Z basis. W.O.L.G., we further assume that the X-block of the tableau M_H has rank n, since any tableau M_H satisfying $\mathrm{rank}(M_H) = n$ could be transformed into such form by the procedure in Lemma 6 in [AG04].

We thus proceed with an explicit mapping procedure.

(1) Performing Gaussian elimination on the X-block of M_H, which requires $O(n^2)$ CNOT gates. Also, such a step will change the Z-block simultaneously.

$$\left(\begin{array}{c|c|c} p & A & B \end{array} \right) \xrightarrow{\text{CNOT}} \left(\begin{array}{c|c|c} p^{(1)} & \begin{array}{c} I \\ A_2 \end{array} & \begin{array}{c} B_1 \\ B_2 \end{array} \end{array} \right)$$

(2) Making the Z-block has full-rank. Lemma 7 in [AG04] achieves this by applying S gates.

$$\left(\begin{array}{c|c|c} p^{(1)} & \begin{array}{c} I \\ A_2 \end{array} & \begin{array}{c} B_1 \\ B_2 \end{array} \end{array} \right) \xrightarrow{\text{S}} \left(\begin{array}{c|c|c} p^{(2)} & \begin{array}{c} I \\ A_2 \end{array} & \begin{array}{c} B_1^{(1)} \\ B_2^{(1)} \end{array} \end{array} \right) = \left(\begin{array}{c|c|c} p^{(2)} & \begin{array}{c} I \\ A_2 \end{array} & \begin{array}{c} NN^T \\ B_2^{(1)} \end{array} \end{array} \right)$$

[3] It is because all A_n and S_m are k-local tensor products of Paulis.

The last equality is owing to the commutation criterion (i.e., Lemma 3.1), as it implies that $B_1^{(1)}$ is a rank-n symmetric matrix.

(3) Multiplying I by N on the Z-black by applying $O(n^2)$ CNOT gates. Likewise, NN^T on the X-block will concurrently multiply $\left(N^T\right)^{-1}$ and result in N.

$$\left(\begin{array}{c|c|c} p^{(2)} & I & NN^T \\ & A_2 & B_2^{(1)} \end{array}\right) \xrightarrow{\text{CNOT}} \left(\begin{array}{c|c|c} p^{(3)} & N & N \\ & A_2^{(1)} & B_2^{(2)} \end{array}\right)$$

(4) Cancelling N on the X-block by applying S gates.

$$\left(\begin{array}{c|c|c} p^{(3)} & N & N \\ & A_2^{(1)} & B_2^{(2)} \end{array}\right) \xrightarrow{\text{S}} \left(\begin{array}{c|c|c} p^{(4)} & 0 & N \\ & A_2^{(2)} & B_2^{(2)} \end{array}\right)$$

(5) Performing Gaussian elimination on Z-block by applying $O(n^2)$ CNOT gates.

$$\left(\begin{array}{c|c|c} p^{(4)} & 0 & N \\ & A_2^{(2)} & B_2^{(2)} \end{array}\right) \xrightarrow{\text{CNOT}} \left(\begin{array}{c|c|c} p^{(5)} & 0 & I \\ & A_2^{(3)} & B_2^{(3)} \end{array}\right) = \left(\begin{array}{c|c|c} p^{(5)} & 0 & I \\ & 0 & B_2^{(3)} \end{array}\right).$$

The last equality due to the commutation criterion (namely, Lemma 3.1).

Let H' be the classical Hamiltonian associated with $p^{(5)}$. We complete the analysis of this mapping procedure by Lemma 3.5.

Lemma 3.5 (Mapping procedure)**.** *There exists an efficient mapping procedure that maps a Pauli commuting k-local Hamiltonian H into a classical Hamiltonian H'. Such a procedure only needs to apply a nearest-neighbor linear-depth Clifford circuit.*

Proof. It suffices to analyze the algorithm presented in this section. It is straightforward to see that implementing the step (2) and the step (4) requires only depth-1 circuits. Regarding other steps, utilizing the canonical form proposed in [BM20], implementing these steps needs a nearest-neighbor linear-depth Clifford circuit. We hereby conclude that the depth of the mapping Clifford circuit is indeed $O(n)$. □

Furthermore, if all rows in the tableau are linearly independent, then it is simple to see $B_2^{(3)} = 0$, namely, the resulting classical Hamiltonian H' is 1-local. In other words, we could learn such a Hamiltonian by approximating the expectation value of local observables $\{\mathrm{Tr}(\rho Z_i)\}_{1\leq i\leq n}$, which only requires $O(\log n)$ product measurement on Z basis.

Making H full-rank by adding additional local terms. If the stabilizer tableau M_H has $\mathrm{rank}(M_H) = r < n$, then the mapping procedure stated as Lemma 3.5 is still useful. Notice there are r linearly independent rows that only have one Z, we can add $n-r$ additional rows such that there is only one Z and all n rows is an $n \times n$ identity matrix on the Z-block. To find such $n-r$ additional local terms, one can perform the Clifford circuit that corresponds to the algorithm presented in this section reversely on these additional local terms, and then we can make H full rank.

3.4 Classical post-processing

Learning a classical Hamiltonian. As the resulting Hamiltonian H' in Section 3.3 is classical, we could learn the coefficients by employing a classical algorithm for learning a Markov random field. It is worthwhile to mention that the classical post-processing seemingly requires exponential time, provided either the resulting Hamiltonian H' is non-local, such as [KM17]; or the degree of the interaction graph associated with H' is $O(n)$ (see [SW12] for further discussion). That is why our approach only learns a Pauli commuting local Hamiltonian efficient under specific conditions.

Conditions of efficient learning Pauli CLH. Our approach's limitation arises from the difference between Gaussian elimination using only column operations and the regular Gaussian elimination. The equivalence between the column-operation-only Gaussian elimination and the regular one only holds for *full-rank tableau matrices*. Specifically, for a Pauli CLH with only linearly independent local terms, the resulting classical Hamiltonian is 1-local, as stated in Section 3.3.

Otherwise, the resulting Hamiltonian's locality is exponentially dependent on the depth of the mapping Clifford circuit. So the resulting Hamiltonian is *sparse* only if we begin with a Hamiltonian corresponding to a sparse tableau. That explains the conditions required in Theorem 1.2.

Recover an approximation of H's coefficients. Certainly, the magnitude of coefficients associated with local terms in H remains the same after applying a Clifford circuit. Utilizing the correspondence between Clifford gates and column operations on the tableau (i.e., Lemma 3.3), we have derived a linear equations systems $A\mathbf{p} = \mathbf{p}'$, where $\mathbf{p}$ is the coefficient vector corresponding to the original Hamiltonian H, and $\mathbf{p}'$ is the coefficient vector associated with the resulting Hamiltonian H'. Since the matrix A is precisely following the mapping procedure in Section 3.3, we could approximately recover the coefficient of the original Hamiltonian H by solving such a linear equations system.

4 Discussion

Caveats for efficient classical post-processing. Because the resulting classical Hamiltonian in Section 3.3 is probably non-local, our approach is efficient (both in time and sample complexity) only under two specific conditions, as stated in Theorem 1.2. In general, the classical post-processing procedure could take exponential time due to the sample complexity lower bound $2^{\Omega(t)}$ for learning t-wise Markov random field [SW12].

It seems hopeless to efficiently learn resulting classical Hamiltonians, since this column-operation-only Gaussian elimination approach is quite brute-force and destroys the local structure of the original Hamiltonian.

Learning Pauli CLH with a shallow-depth Clifford circuit. The mapping Clifford circuit produced by the mapping procedure in Section 3.3) could be linear depth. It raises a natural question:

Open Problem 4.1. *Could we learn a Pauli commuting local Hamiltonian by applying a shallow-depth quantum circuit on the Gibbs state?*

In [APS19], the authors suggest positive evidence, in particular, we can learn the modified Toric code model[4] without applying any quantum circuit. Moreover, classical post-processing is even

[4]The coefficients of this modified Toric code model is not necessarily 1.

efficient. This observation leads to a relaxed commutation criterion – *X-Z commutation criterion*, namely, replacing $\mathbf{X}^{(i)} \cdot \mathbf{Z}^{(j)} \oplus \mathbf{X}^{(j)} \cdot \mathbf{Z}^{(i)} = 0$ (in Lemma 3.1) by $\mathbf{X}^{(i)} \cdot \mathbf{Z}^{(j)} = \mathbf{X}^{(j)} \cdot \mathbf{Z}^{(i)} = 0$. Particularly, all local terms on the X-block in the tableau commute with all local terms on the Z-block. This condition further infers a new mapping in Proposition 4.2:

Proposition 4.2 (Mapping into primal-dual Markov random fields with hidden variables)**.** *For any Pauli commuting local Hamiltonian H satisfying the X-Z commuting criterion, namely, $\forall i, j \in [m]$, $\mathbf{X}^{(i)} \cdot \mathbf{Z}^{(j)} \oplus \mathbf{X}^{(j)} \cdot \mathbf{Z}^{(i)} = 0$. We can map this family of Hamiltonians to a primal-dual pair of Markov random fields (H_X, H_Z) with hidden variables, where H_X corresponds to the distribution*

$$\Pr\left[X = (x_1, \cdots, x_n)\right] \propto \sum_{z_1, \cdots, z_n \in \{\pm 1\}} \exp\left(-\sum_{i \in [m]} H_i(x_1, \cdots, x_n, z_1, \cdots, z_n)\right),$$

and H_Z corresponds to the distribution

$$\Pr\left[Z = (z_1, \cdots, z_n)\right] \propto \sum_{x_1, \cdots, x_n \in \{\pm 1\}} \exp\left(-\sum_{i \in [m]} H_i(x_1, \cdots, x_n, z_1, \cdots, z_n)\right).$$

(H_X, H_Z) are so-called a primal-dual pair if visible variables $x_1, \cdots, x_n$ of H_X are hidden variables of H_Z, and visible variables $z_1, \cdots, z_n$ of H_Z are hidden variables of H_X.

The proof follows from a straightforward calculation, namely, as the distribution corresponding to H_X (the same to H_Z) follows $\langle x_1, \cdots, x_n | \exp(-\sum_{i \in [m]} H_i) | x_1, \cdots, x_n \rangle$.

Furthermore, it is worthwhile to mention that our setting differs from known results on the learning Boltzmann machine with hidden variables [BKM19, BB20]. The obstacles showed by [BKM19] arises from a single Markov random field with hidden variables. In contrast, our problem circumvents such issues since a primal-dual pair with hidden variables could complete each other's information.

Even though this tableau representation of Paulis specified in Section 3.1 seems helpful, it is unclear how to take advantage of such techniques yet.

Acknowledgments

The author thanks Itai Arad for introducing the problem, insightful discussion, and suggestions of the earlier version of this note. The author also thanks Hsin-Yuan (Robert) Huang and Anurag Anshu for helpful discussion. This work was partially done while the author visited the Centre for Quantum Technologies at National University Singapore in Summer 2019.

References

[AAKS20] Anurag Anshu, Srinivasan Arunachalam, Tomotaka Kuwahara, and Mehdi Soleimanifar. Sample-efficient learning of quantum many-body systems. *arXiv preprint arXiv:2004.07266*, 2020.

[AdW17] Srinivasan Arunachalam and Ronald de Wolf. Guest column: A survey of quantum learning theory. *ACM SIGACT News*, 48(2):41–67, 2017.

[AdW18] Srinivasan Arunachalam and Ronald de Wolf. Optimal quantum sample complexity of learning algorithms. *The Journal of Machine Learning Research*, 19(1):2879–2878, 2018.

[AG04] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A*, 70(5):052328, 2004.

[AHS85] David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowski. A learning algorithm for boltzmann machines. *Cognitive science*, 9(1):147–169, 1985.

[APS19] Itai Arad, Anupam Prakash, and Miklos Santha. Unpublished result. 2019.

[BAL19] Eyal Bairey, Itai Arad, and Netanel H Lindner. Learning a local hamiltonian from local measurements. *Physical review letters*, 122(2):020504, 2019.

[BB20] Guy Bresler and Rares-Darius Buhai. Learning restricted boltzmann machines with few latent variables. *arXiv preprint arXiv:2006.04166*, 2020.

[BKM19] Guy Bresler, Frederic Koehler, and Ankur Moitra. Learning restricted boltzmann machines via influence maximization. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 828–839, 2019.

[BM20] Sergey Bravyi and Dmitri Maslov. Hadamard-free circuits expose the structure of the clifford group. *arXiv preprint arXiv:2003.09412*, 2020.

[BP12] Winton Brown and David Poulin. Quantum markov networks and commuting hamiltonians. *arXiv preprint arXiv:1206.0755*, 2012.

[Bre15] Guy Bresler. Efficiently learning ising models on arbitrary graphs. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 771–782, 2015.

[BV05] Sergey Bravyi and Mikhail Vyalyi. Commutative version of the local hamiltonian problem and common eigenspace problem. *Quantum Information & Computation*, 5(3):187–215, 2005.

[CW20] Jordan Cotler and Frank Wilczek. Quantum overlapping tomography. *Physical Review Letters*, 124(10):100401, 2020.

[HJPW04] Patrick Hayden, Richard Jozsa, Denes Petz, and Andreas Winter. Structure of states which satisfy strong subadditivity of quantum entropy with equality. *Communications in mathematical physics*, 246(2):359–374, 2004.

[KM17] Adam Klivans and Raghu Meka. Learning graphical models using multiplicative weights. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 343–354. IEEE, 2017.

[QR19] X. Qi and D. Ranard. Determining a local hamiltonian from a single eigenstate. *Quantum*, 3:159, 2019.

[SW12] Narayana P Santhanam and Martin J Wainwright. Information-theoretic limits of selecting binary graphical models in high dimensions. *IEEE Transactions on Information Theory*, 58(7):4117–4134, 2012.

[VMLC16] Marc Vuffray, Sidhant Misra, Andrey Lokhov, and Michael Chertkov. Interaction screening: Efficient and sample-optimal learning of ising models. In *Advances in Neural Information Processing Systems*, pages 2595–2603, 2016.