Learning Pauli Commuting Local Hamiltonians

Yupan Liu. Hebrew University of Jerusalem, Israel

THE HEBREW UNIVERSITY OF JERUSALEM

Abstract

Learning an unknown Hamiltonian from local measurements is an increasingly important task in the NISQ era. Recent work by [BAL19] proposed an approach to learn *non-commuting* local Hamiltonians, though their method fails for *commuting* Hamiltonians.

We provide a method to learn *Pauli commuting local Hamiltonians*, which is a subclass of general CLHs. Given $\exp(n)$ copies of the Gibbs state ρ of a Pauli commuting local Hamiltonian H on n qubits, one can learn such a Hamiltonian by

1. Applying a *linear-depth* Clifford circuit on given copies;
2. Performing classical post-processing.

Our result sheds light on learning general commuting local Hamiltonians using local measurements.

Problem Statement: A Quantum Perspective

Consider an "inverse problem" of finding ground states by given a Hamiltonian (i.e., the *local Hamiltonian problem*), namely

m copies of the Gibbs state ρ → Leaning a local Hamiltonian by an algorithm $\mathcal{A}$ → Coefficients of a local Hamiltonian

Q1: What's the *sample complexity* m?

Q2: What's the *time complexity* of a learning algorithm $\mathcal{A}$?

Problem Statement: A Classical Perspective

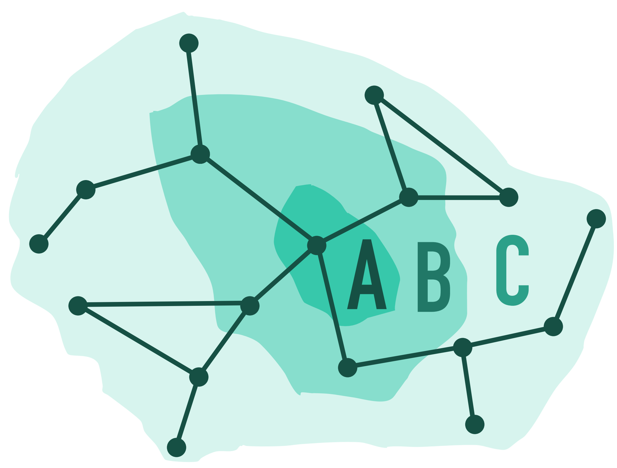
The classical analog is learning a Markov random field (e.g. [KM17]):

Gibbs distribution defined on $z = (z_1, \cdots, z_n) \in \{\pm 1\}^n$,

$$\Pr[Z = z] \propto \exp(-H_c) := \exp\left(\sum_{i \neq j \in [n]} A_{ij} z_i z_j + \sum_{i \in [n]} \theta_i z_i\right).$$

Configuration graph $G = ([n], E)$ where $(i,j) \in E$ if $A_{i,j} \neq 0$.

Let $\mathbf{X_i} := \{z_j | j \neq i, (i,j) \in E\}$ and $Y_i := (1 - z_i)/2$ be random variables. Note $\mathbf{X_i}$ is only dependent on *the neighbors of* i due to the *Markovianity* $\Pr[A|B] = \Pr[A, C|B]$.



Task. Given m random samples $(\mathbf{X_i}, Y_i)$ satisfying $\mathbb{E}[Y_i | \mathbf{X_i} = \mathbf{x}] = \sigma(\mathbf{A_j} \cdot \mathbf{x} + \theta_\mathbf{i})$ where $\sigma(x) = 1/(1 + e^{-x})$, recover $\mathbf{A_j}$ and $\theta_\mathbf{i}$.

Main Algorithm: Learning a Pauli CLH

m copies of the Gibbs state of Pauli CLH

Mapping Gibbs states of a Pauli CLH into Gibbs states of a classical Hamiltonian [AG04,JST+20]

Learning candidates of local terms in Hamiltonian using O(log n) samples [BAL19,CW19]

Draw samples from the classical Gibbs distribution

local measurements

Classical post-processing with exp(n) samples: Learning a classical Hamiltonian

[KM17]

Mappling a classical Hamiltonian back into a Pauli CLH

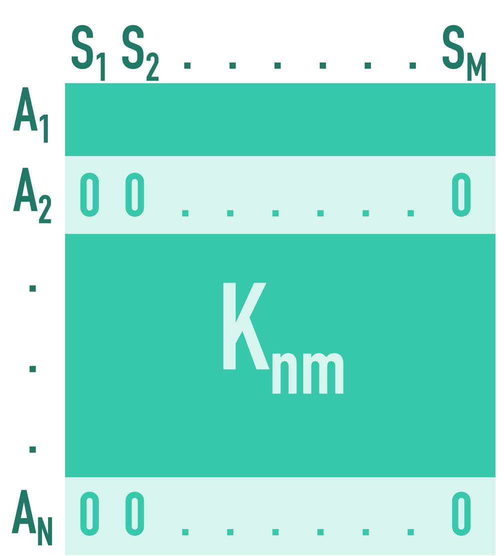
[AG04,JST+20]

Coefficients (associated with local terms) of a Pauli CLH

Proof Technique: Learning candidates of local terms

Consider $H_L = \sum_{m=1}^{M} c_m S_m$ defined on a local patch L with a Gibbs state ρ, [BAL19] implies that $\forall$ local observables A_n inside L, $\sum_{m=1}^{M} c_m \mathrm{Tr}(\mathbf{i}\rho[S_m, A_n]) = 0$ and $|\{A_n \text{ on } L\}| \leq \mathrm{poly}(n)$. Let matrix $K_{nm} := \mathrm{Tr}(\mathbf{i}\rho[S_m, A_n])$.

Claim. If A_n is a local term in H_L, then $\forall S_m$, $[S_m, A_n] = 0$, i.e., the n-th row of K is all-zero.



Defintion: Pauli *Commuting* Local Hamiltonians

A k-CLH $H = \sum_{i=1}^{m} H_i$ on n-qubit is a Pauli CLH if it satisfies

- $\forall i \in [m]$, $H_i = a_i \sigma_1^{(i)} \otimes \cdots \otimes \sigma_n^{(i)}$, where $\sigma_j^{(i)} := \mathbf{i}^{x_j^{(i)} z_j^{(i)}} X^{x_j^{(i)}} Z^{z_j^{(i)}}$.
- $\forall i, j \in [m]$, $[H_i, H_j] = 0$.

A Pauli CLH can be described by *a Stabilizer tableau* [AG04], namely each local term can be represented by a $(2n+1)$-tuple:

	a_i	$x_1^{(i)}$	$\cdots$	$x_n^{(i)}$	$z_1^{(i)}$	$\cdots$	$z_n^{(i)}$	
H_1	a_1	$x_1^{(1)}$	$\cdots$	$x_n^{(1)}$	$z_1^{(1)}$	$\cdots$	$z_n^{(1)}$	$\mathbf{Z}^{(1)}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	
H_m	a_m	$\mathbf{X}^{(m)}$ $x_1^{(m)}$	$\cdots$	$x_n^{(m)}$	$z_1^{(m)}$	$\cdots$	$z_n^{(m)}$	
	(p)		(A)	$\mathbf{X}_n$	$\mathbf{Z}_1$	(B)		

Commutation. $\forall i, j$, $[H_i, H_j] = 0 \Leftrightarrow \mathbf{X}^{(i)} \cdot \mathbf{Z}^{(j)} \oplus \mathbf{X}^{(j)} \cdot \mathbf{Z}^{(i)} = 0$.

Linear *column* operations on the tableau. It is equivalent to apply Clifford gates [AG04], such as Hadamard, S, CNOT:

- Had_k: swap $\mathbf{X}_k$ with $\mathbf{Z}_k$ and $\mathbf{p}' := \mathbf{p} \oplus (\mathbf{X}_i \odot \mathbf{Z}_i)$.
- S_k: $\mathbf{Z}'_k := \mathbf{Z}_k \oplus \mathbf{X}_k$ and $\mathbf{p}' := \mathbf{p} \oplus (\mathbf{X}_i \odot \mathbf{Z}_i)$.
- $\mathrm{CNOT}_{i,j}$: $\mathbf{X}'_j := \mathbf{X}_i \oplus \mathbf{X}_j$, $\mathbf{Z}'_i := \mathbf{Z}_i \oplus \mathbf{Z}_j$ and $\mathbf{p}' := \mathbf{p} \oplus (\mathbf{X}_i \odot \mathbf{Z}_j \odot (\mathbf{X}_j \oplus \mathbf{Z}_i \oplus \mathbf{1}))$.

Proof Technique: Mapping Pauli CLHs into Classical Hamiltonians

Applying a $O(n/\log n)$-depth Clifford circuit [AG04, JST+20]:

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

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

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

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

An efficient classical post-processing condition:

$\mathrm{rank}(A|B) = m$ and all rows in $(A|B)$ are *linearly independent*.

Now we obtain a 1-local classical Hamiltonian since $B_2^{(3)} = 0$. It can be learned *efficiently* in both time complexity and sample complexity.

Open Problem: Towards an Efficient Classical Post-processing

A classical algorithm for learning classical Hamiltonians. [KM17] provides an algorithm for learning a k-local classical Hamiltonian with run-time $n^{\Theta(k)}$ and sample complexity $n^{O(k)}$.

Main issues. The resulting classical Hamiltonian H' is *non-necessarily local* since $B_2^{(3)} \neq 0$ in general, so applying [KM17] for H' directly requires $\exp(n)$ run-time and $\exp(n)$ samples. Could we learn a classical Hamiltonian obtained from a Pauli CLH *efficiently*?

Open Problem: Learning CLHs by Matrix MWU Methods

[KM17] is based on *multiplicative weight updates (MWU)* and *Markov property* of Gibbs distributions. Notice Markov property holds for commuting local Hamiltonians due to the *Koashi-Imoto decomposition* [KI02]. Could we learn CLHs using *Matrix* MWU methods?

Acknowledgment. The author thanks Itai Arad for introducing the problem that considers here and helpful discussion.

References

[AG04] S. Aaronson and D. Gottesman. Improved simulation of stabilizer circuits. *Phys Rev A*, 70(5):052328, 2004.

[BAL19] E. Bairey, I. Arad, and N. H Lindner. Learning a local hamiltonian from local measurements. *Phys Rev Lett*, 122(2):020504, 2019.

[CW19] J. Cotler and F. Wilczek. Quantum overlapping tomography. *arXiv:1908.02754*, 2019.

[JST+20] J. Jiang, X. Sun, S. Teng, B. Wu, K. Wu, and J. Zhang. Optimal space-depth trade-off of cnot circuits in quantum logic synthesis. In *40th SODA*, pages 213–229. SIAM, 2020.

[KI02] Masato Koashi and Nobuyuki Imoto. Operations that do not disturb partially known quantum states. *Phys Rev A*, 66(2):022318, 2002.

[KM17] A. Klivans and R. Meka. Learning graphical models using multiplicative weights. In *2017 IEEE 58th Annual Symposium on FOCS*, pages 343–354. IEEE, 2017.