

# Hamiltonian Simulation Circuit

David Ponarovsky

February 22, 2023

## Abstract

In this work we deal with the problem of designing efficient circuit to simulate the development in time of the initial state by the given  $\Delta$ -local Hamiltonian. We suggest an heuristics and compare the circuit depth obtained by them to the circuit which is yielded by naive approach.

## 1 The Problem.

Generate a circuit, using no more than 10 qubits, that approximates the unitary  $e^{-iH}$  where  $H$  is the qubit Hamiltonian of a **LiH** (lithium hydride) molecule. The **LiH** Hamiltonian is composed of 276 Pauli strings, and can be found [HERE](#). The approximation error is defined in the next section, and should be less than 0.1. The circuit should be composed of the  $CX$  and single qubit gates only.

**Definition 1.** Consider the following relation  $\prec$ . We will say that  $H_i \prec H_j$  if  $H_j$  could obtained by applying the replacements  $I \mapsto X$  or  $I \mapsto Z$ . For example, if  $H_1 = IXIXZ$  and  $H_2 = IXZZXZ$  then  $H_1 \prec H_2$ , observe also that  $I^{\otimes n} \prec H_i$  for any  $H_i$ . Note that this relation defines an order (transitive).

**Definition 2** (Order Tree). Consider the directed graph  $G = (V, E)$ , such that the vertices associated with the Hamiltonian terms, and there is a directed edge between  $H_i$  and  $H_j$  if  $H_i \prec H_j$ . We will say that  $G$  is the ordered tree of  $H$ .

**Lemma 1.** There is a classical algorithm which compute for given  $H$  with  $m$  terms over  $n$  qubits the associated ordered tree, at  $O(m^2n)$  classical time.

*Proof.* Algorithm.

---

**Algorithm 1:** Construct the ordered tree from given local terms  $H_1, H_2..H_m$ .

---

```
1
2 if  $H \prec H_v$  then
3    $H \leftrightarrow H_v$ 
4   call in recursion  $(v, H_v)$ 
5 end
6 else
7   Let  $L$  be an empty list.
8   Let  $w \leftarrow \emptyset$ 
9   for  $\{v, u\} \in E$  do
10    if  $H_v \prec H_u$  then
11       $L \leftarrow L \cup \{H_u\}$ 
12    end
13    else if  $H_u \prec H_v$  then
14       $w \leftarrow u$ 
15    end
16  end
17  if  $L \neq \emptyset$  then
18    Define new vertex  $v'$ , and edge  $\{v, v'\}$ 
19    Define the edges  $\{v', u\}$  for any  $u \in L$ 
20    Remove the edges  $\{v, u\}$  for any  $u \in L$ 
21  end
22  else
23    call in recursion  $(w, H)$ 
24  end
25 end
```

---

□

**Definition 3.** Let  $H_i$  be a single term of the Hamiltonian, we will define the  $\text{support}(H_i)$  to be a vector  $v \in \mathbb{F}_2^n$  such that  $v_j = 1$  if  $H_i$  act non-trivially on the  $j$ th qubit and 0 else.

Noitce that each term  $H_i$  such that  $\text{support}(H_i) = 1^a 0^b = (1, 1, \dots, 1, 0, \dots, 0)$  can be simulated in parallel with term  $H_j$  with complementary support (i.e  $0^a 1^b$ ). In addition, if the support isn't a continous segment of 1's then one could perform rotate the base by permutation at a cost of the "mixing size". [\[COMMENT\]](#) Change  $G$  to be the graph which each node  $v = (P, \text{base change})$ .

**Definition.** Denote by  $m$  the number of local Hamiltonian terms and consider the graph  $G = (\{H_i\}, E)$  which is a wighted complte graph over  $m$  vertices. Such that the wight function defined by  $w(u, v) = \# \{j | H_v^j \neq H_u^j \text{ and } H_v^j, H_u^j \neq I\}$ . We will call to such  $G$  the Qubic embedding or shortly just  $G$ .

**Theorem.** Given a  $\Delta$ -local Hamiltonain over quabits and  $m$  terms such that  $n > \frac{m}{\Delta+1}$  then there is a circuit which simulate a single Turturize step at depth:

$$\begin{aligned} D(n, m) &\leq \log_{\frac{\Delta+1}{\Delta}}(m) \cdot \frac{\Delta}{1+\Delta} n \cdot (2 + 2\Delta) \\ &= \log_{\frac{\Delta+1}{\Delta}}(m) \cdot 2\Delta n \end{aligned}$$

Let us abuse the notation and denote by  $H$  also the random variable which count the support size of local term which is sampled uniformly from the total Hamiltonian.

**Theorem.** [\[COMMENT\]](#) Change to uniform drown  $\Delta$ -local Hamiltonian. Given an Hamiltonian with expected support  $\mathbb{E}[H] := \mathbb{E}_{\sim_i} [\text{Ham}(\text{support}(H_i))]$  then there is a circuit which simulate a single step  $e^{iH\Delta t}$  at depth:

$$\begin{aligned} D(n, m) &\leq m(1 + \mathbb{E}_{\sim_i}[H]) \\ &\quad + 4w(\text{MST}(G)) + \left(2 - \frac{1}{m}\right) n^2 \end{aligned}$$

Where  $w(\text{MST}(G))$  is the weight of the minimal spanning tree of  $G$ . Please, noitce that in our case  $n = 10, m = 276$ .

Before continuing to the proof let's analyze the naive approach, just handle each of Hamiltonian terms one by one. Assume the the hamming weight of the support of  $H_i$  equal to  $d_i$ . so we pay two steps to rotate each wire into the parity base and uncompute it in the end. then we could apply the  $CX$  from each qubit in the support to a chosen one which will sum the wires parity and finally will propogate the sum trough the  $RZ(\theta\Delta t)$  gate (rotation by the coefficient of the term and the step size). So in total we pay  $2 + 2d_i$  for each term and therefore the whole circuit will require:

$$\begin{aligned} D^{\text{naive}}(n, m) &= \sum_i 2 + 2d_i = m \sum_i \frac{2 + 2d_i}{m} = \\ &= 2m(1 + \mathbb{E}_{\sim_i}[H]) \end{aligned}$$

Hence the theorem inequality could be written as:

$$\begin{aligned} D(n, m) &\leq \frac{1}{2} D^{\text{naive}}(n, m) \\ &\quad + 4w(\text{MST}(G)) + \left(2 - \frac{1}{m}\right) n^2 \end{aligned}$$

And therefore, for an Hamiltonians which have a grater number of terms than the number of the qubits which they act on, we expect to see an improvement which is almost an half of the naive solution. In fact this improvement is much more significant than might seem as simulating up to  $\varepsilon$ -error requires  $\text{Poly}(\frac{1}{\varepsilon})$  repetitions [\[COMMENT\]](#) add a citation, so for full simulation we have obtained a  $\sim (\frac{1}{2})^{\text{poly}(\frac{1}{\varepsilon})}$  speed-up factor.

**Solution.** Let's formulate the claim which mentioned earlier, suppose that  $H_i, H_j$  are terms which their supports are disjoint and also they lay on continues segments. Then we can simulate the terms in parallel.

**Lemma 2.** let  $H_i, H_j$  and integers  $a, b, c, d$  such that  $0^a 1^b, 1^c 0^d$ . Then we can simulate  $e^{iH_i + H_j}$  with circuit with depth  $2 + \max_{i,j} 2 \cdot \mathbf{Ham}(\text{support}(H_i))$ .

*Proof.* It's clear that the first rotation, and the final uncompute stages remain unchanged. So we should deal only with 2-qubits operators (the cnot's). By the  $a < c$  restriction it follows that each  $CX$  gate in the naive implementation of the  $H_i$  cross only elements which  $H_j$  act trivially on. Hence if at time-step  $k$  the original operations in each of the naive implementations were  $CX(i_1, i_2), CX(j_1, j_2)$  then  $j_1, j_2 \notin [i_1, i_2]$ . Repeating on the above argument, but consider the  $d < b$  constraint instead yields that  $i_1, i_2 \notin [j_1, j_2] \Rightarrow [i_1, i_2] \cap [j_1, j_2] = \emptyset$  and therefore we can put those gates on the same time step  $\square$

**Corollary.** Assume that  $H_i, H_j$  have a disjoint supports. Then by swapping the wires (including phase swap) into continues segments, we obtain a simulation at cost of :  $\text{cost}(\text{Permutation}) + 2 + \max_{i,j} 2 \cdot \mathbf{Ham}(\text{support}(H_i))$ .

The permutation which swapping between  $v_1, v_2 \dots v_k \leftrightarrow u_1, u_2 \dots u_k$  where  $v_i, u_j \in [n]$  are wires/qubits indices could be done by swapping each of the pair separately. Namely,  $\mathbf{SWAP}(v_i, u_i)$  for each of the pairs. Figure 1 presents the two qubit gate **SWAP** and Figure 2 is an example of the generalization obtained by concatenation such gates.

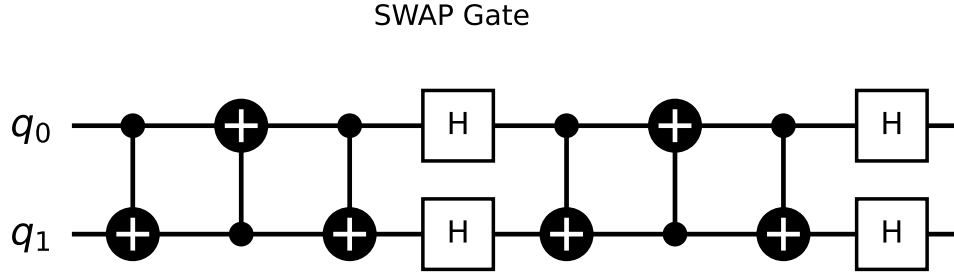


Figure 1: The implementation of the two-qubit **SWAP** gate. The left part replace the bit values of the qubits, then we are rotate into the dual space and swap the phases finally we return back to the original space.

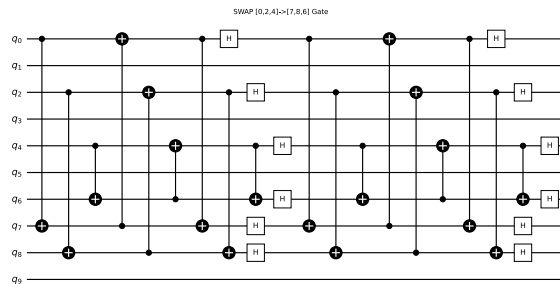


Figure 2: Example for the general swap gate **SWAP**, In that example  $v = (e_1, e_2, e_4)$  and  $u = (e_7, e_8, e_6)$ . And notice that the mapping swap between  $e_4 \leftrightarrow e_6$  wires.

Now Let  $v \in \mathbb{F}_2^n$ , we will use the notation  $\mathbf{SWAP}^v$  to name the family of circuits such that for every  $v_i = 1$  the  $i$ 'th wire will be swapped with  $j$ 'th wire such that  $j \leq \mathbf{Ham}(v)$ . In other words, we could think about a circuit  $C \in \mathbf{SWAP}^v$  as the one who lifting (advancing) the interesting wires to the top of the circuit.

In similar way for  $u, v \in \mathbb{F}_2^n$  we will name by  $\mathbf{SWAP}^{v,u}$  the family of gates such that for every  $C_1 \in \mathbf{SWAP}^v$  and  $C_2 \in \mathbf{SWAP}^{v,u}$  we get that  $C_2 C_1 \in \mathbf{SWAP}^u$ .

To understand the rational behind those notations consider the following possible terms:

- $H_1 = IXXII$
- $H_2 = XIIZI$
- $H_3 = IIXIX$
- $H_4 = XIIZI$

In case we have decided to simulate first  $H_1, H_2$  which can be done in parallel after swapping the first and the third qubit. Then we already have made half of the way through the rotation in which  $H_3, H_4$  can be compute in parallel. That insight bring us to the next Lemma.

**Definition.** Let  $G'$  be the the graph which is obtained from  $G$  by indentify each vertex with his support.

**Lemma 3.** Let  $v, u$  be vertices in  $G'$  which defined as which by definition are also match to a pair of vectors in  $\mathbb{F}_2^n$ . Then:

$$\text{Depth}(\mathbf{SWAP}^{v,u}) \leq \mathbf{Ham}(v, u) \cdot \text{Depth}(\mathbf{SWAP})$$

*Proof.* By induction over  $w(v, u)$ . For the base case consider  $u, v$  such  $u$  differ from  $v$  only by single coordinate. Denote it by  $j$ . Observes that if  $\mathbf{Ham}(u) < \mathbf{Ham}(v)$  then  $\mathbf{SWAP}^{v,u}$  is just the identity. So we can assume that  $\mathbf{Ham}(u) = \mathbf{Ham}(v) + 1$  and that  $v_j = 0, u_j = 1$ , denote by  $j'$  the wire which  $\mathbf{SWAP}^v$  sends  $j$  into. then it's sufficient to apply the gate  $\mathbf{SWAP}(\mathbf{Ham}(v) + 1, j')$ .

Assume the correctness of the lemma for any  $d' \leq d$ , denote again by  $j$  one of the coordinate in which  $v, u$  act different, And consider the vector  $u' = u \oplus v_j \Rightarrow u'_j = v_j$ , if  $\mathbf{Ham}(u, v) = d$  then  $\mathbf{Ham}(u', v) = d - 1$  and therefore there exist a circuit  $C^{v,u'} \in \mathbf{SWAP}^{v,u'}$  at depth  $\mathbf{Ham}(v, u') \cdot \text{Depth}(\mathbf{SWAP})$ . By the fact that  $\mathbf{SWAP}^{v,u} = \mathbf{SWAP}^{v,u'} \mathbf{SWAP}^{u',u}$  and because  $\mathbf{Ham}(u', u) = 1$  we get that:

$$\begin{aligned} D(\mathbf{SWAP}^{v,u}) &= D(\mathbf{SWAP}^{v,u'}) + D(\mathbf{SWAP}^{u',u}) \\ &\leq (\mathbf{Ham}(v, u') + \mathbf{Ham}(u, u')) \cdot \text{Depth}(\mathbf{SWAP}) \\ &= \mathbf{Ham}(v, u) \cdot \text{Depth}(\mathbf{SWAP}) \end{aligned}$$

And that finishes the proof □

**Corollary.** Let  $P = \langle v_1, v_2 \dots v_l \rangle$  a vertices path in the graph  $G$  as defined above. And let  $U$  be circuit such that for each  $v_i \in P$   $U$  apply the circuit  $U_i$  at the [\[COMMENT\] I should define it](#) base. Then there exist an implementation for that circuit at depth:

$$\begin{aligned} D(U) &= \sum_i D(U_i) + D(\mathbf{SWAP}^{v_i, v_{i+1}}) \\ &= \sum_i D(U_i) + \mathbf{Ham}(v_{i-1}, v_i) \cdot \text{Depth}(\mathbf{SWAP}) \\ &= \sum_i D(U_i) + w(v_{i-1}, v_i) \cdot \text{Depth}(\mathbf{SWAP}) \\ &= w(P) \cdot \text{Depth}(\mathbf{SWAP}) + \sum_i D(U_i) \end{aligned}$$

**Expectation of intersect terms.** Denote by  $X_i^j$  the indicator which assign 1 to the event that the  $i$  term act non trivially on the  $j$ th qubit. Then it's clear that

$$\mathbb{E}[\text{intersects}] \leq \mathbb{E} \left[ \sum_{i,k}^m n \cdot X_i^j X_k^j \right] = \binom{m}{2} np^2$$

If we sample the terms form a uniform distribution over the  $\Delta$ -local terms then  $p = \binom{n-1}{\Delta-1} / \binom{n}{\Delta} = \frac{\Delta}{n}$ . And therefore the expectation of the intersections is less then  $\binom{m}{2} \frac{\Delta^2}{n}$ . Hence by markov inequality we obtain that

$$\Pr \left[ \text{intersects} \geq \frac{1}{2} \binom{m}{2} \right] \leq \frac{2\Delta^2}{n}$$

**The Algorithm.**

**The general concept.** We will identify each pair of

---

**Algorithm 2:** Simulate a single step  $e^{iH\Delta t}$

---

```

1
2  $|\psi\rangle \leftarrow$  the initial state.  $T \leftarrow \text{MST}(G)$ .
3  $P \leftarrow$  ordering of the vertices  $\langle v_1, \dots, v_l \rangle$  according to the DFS scanning of  $T$ .
4
5 for  $v \in P$  do
6    $|\psi\rangle \leftarrow \text{SWAP}^{v_{i-1}, v_i}$ 
7   while  $\exists H_i, H_j$  s.t.  $\text{supp}(H_i) = v = \overline{\text{supp}(H_j)}$  do
8      $|\psi\rangle \leftarrow U(H_i)_v \otimes U(H_j)_{\bar{v}} |\psi\rangle$ 
9     mark  $H_i, H_j$ .
10  end
11 end
12
13 complete the left terms by the naive approach.
14 return  $|\psi\rangle$ 

```

---

**Lemma 4.** Let  $\{X_i\}$  be a monotonic set of numbers bounded by  $2\mu$  with expectation  $\mu$  and variance  $\sigma^2$ . Then:

$$\sum \frac{1}{m} \max \{X_i, X_{i+\frac{n}{2}}\} \leq \mu + \frac{1}{2}\sigma^2$$

*Proof.* Define  $Y : \Omega \rightarrow \mathbb{R}$  to be the random variable which is the hist of the  $X$  by  $\mu$  i.e  $Y = X - \mu$ . From the linearity of expectation we get that  $\mathbb{E}[X] = \mu + \mathbb{E}[Y]$ . let's note by  $Y^s$  and  $Y^{as}$  the symmetric and asymmetric parts of  $Y$ . the  $\square$ .

$$\mathbb{E}[X] = \frac{1}{2}M + \mathbb{E}[X^s - M] = \frac{1}{2}M + \mathbb{E}[(X - M)^s + (X - M)^{as}] = \frac{1}{2}M + \frac{1}{2}\mu + \mathbb{E}[(X - M)^{as}] \quad \square$$

**Lemma 5.** Let  $H = \frac{1}{m} \sum H_i$  be Hamiltonian such each term  $H_i$  is a generalized pauli operator. Let  $P$  be random pauli which sampled uniformly. Then:

**Lemma 6.** Majoritiy principle. We can always assume that the more then quarter of the paulis are identity. *[COMMENT] doesn't correct, but can be passed if we change the definition of the solid tensor to include only differ coordinates.*

$$U^t = \sqrt{X}^\dagger \sqrt{X} U \sqrt{X} \sqrt{X}^\dagger U \sqrt{X}^\dagger \sqrt{X} \dots U \sqrt{X}^\dagger \sqrt{X}$$