# JOGGING A HAMILTONIAN:

## Simulating Hamiltonians with Quantum Walks and other methods

RUTVIJ [1] AND ARJO [2]

Apr 28, 2022


CONTENTS



[1] *Rutvij Menavlikar*, CQST, IIITH
[2] *Alapan Chaudhuri*, CQST, IIITH

## ABSTRACT

The problem of efficiently simulating a Hamiltonian is one of the long standing problems in computational physics. Today, the problem of Hamiltonian simulation stands as one of the most impacting and significant contribution of quantum computers. Furthermore since it is a BQP-complete problem, devising any efficient classical algorithm for the same is believed to be intractable.

In this project, we plan to study and analyse Hamiltoninan simulation using quantum walks. We will also compare the efficiency of these algorithms with other methods such as trotterization and quantum signal processing.


# 1 INTRODUCTION

Hamiltonian simulation serves as the basis for many problems in science. The universe is of quantum nature, which means that to simulate particles or any other natural entities you need to simulate its Hamiltonian. However, computing the evolution given the Hamiltonian is also not very straightforward.

Due to the discrete nature of classical computers, it is not possible to efficiently simulate any of the above on the same. This gives quantum computers an edge since they can work with qubits to simulate the dynamics of a system given its Hamiltonian. This makes Hamiltonian Simulation the most impacting and significant contribution of quantum computers.

Furthermore, simulating hamiltonian is a BQP-complete problem. Thus, it is believed to be intractable by classical computers. Devising any efficient classical algorithm for hamiltonian simulation would mean proving that $P = BQP$.

## 1.1 Why simulate Hamiltonians?

Computational simulation of the physical systems provides important insights and acts as a bridge between theory and experiments.

Moreover, it has more than a handful of applications, namely:

- Drug design and protein folding: Finding the native structure of the protein is equivalent to the problem of finding the ground state of the system.
- Graph theoretic problems: Several graph theory related problems such as Graph Coloring (which is NP-complete) can be cleverly mapped to finding ground states of some classes of Hamiltonians.
- Note that finding the ground state of an Hamiltonian is equivalent to finding the $\min \langle \phi | H | \phi \rangle, \forall\ |\phi\rangle$.
- Furthermore, Hamiltonian simulation can be used as a subroutine for implementing continuous-time quantum walks and solving linear equations (), amongst others.

## 1.2 Why simulate Hamiltonians in quantum computers?

Computational simulation of the physical systems provides important insights and acts as a bridge between theory and experiments. But can a quantum system be efficiently simulated by a classical computer? The answer is certainly (almost), 'No!' as said by Bell. Even simulation using pseudo-random variables has exponential

computational overhead.

In the case where the Hamiltonian consists of a sum of interaction terms between small subsystems, the simulation is thought to be exponentially more efficient than classical simulation.

## 2 THE HAMILTONIAN

A Hamiltonian can be time-independent (then you are lucky) or time-dependent.

- Time-independent: $U(t) = e^{-iHt}$
- Time-dependent: $U(t) = e^{-i\int_0^t H(s)}$

### 2.1 Local Hamiltonians

$$H = \sum_{j=1}^{n} H_j$$

Here, $H$ denotes a local hamiltonian where each $H_j$ acts on $k = O(1)$ qubits.

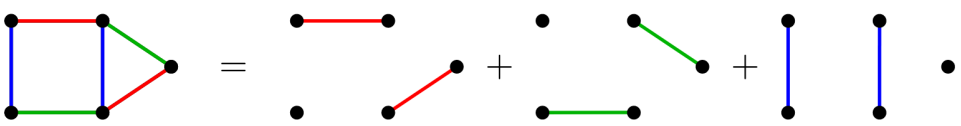
### 2.2 Sparse Hamiltonians

Consider the matrix form of the Hamiltonian, $H$, to be $N \times N$. Then, $H$ is said to be sparse if it has at most $d$ non zero entries per row $d = \text{poly}(\log N)$. Some interesting points on sparse Hamiltonians:

- There should exist some way for efficiently computing the location and value of the $j^{th}$ nonzero entry $\forall\ r \in$ rows of $H$. This is called row computability.
- A $k$-local Hamiltonian with $n$ terms is $d$-sparse with $d = 2^k n$.
- A $k$-local Hamiltonian with $n$ terms can be expressed as a linear combination of $\leqslant n4^k$ Pauli operators, each of which are unitaries.

### 2.3 Problem of efficient decomposition

Finding the efficient decomposition of a given Hamiltonian is not an easy task. One approach for dealing with the same for a sparse Hamiltonian ($H$) is by representing $H$ in terms of a graph and having its edges colored. Now, given that a sparse graph can be efficiently colored using only local information, we can conclude that this procedure will result in a decomposition yielding efficient simulations. In layman terms, the simulation breaks into small pieces that are easy to handle (shown below diagramatically).



## 3 ALGORITHMS FOR SPARSE HAMILTONIAN SIMULATION

Today, there exists several frameworks for simulating Hamiltonians, namely:

- **Trotterization**: This was described by Zeeshan, Shreyas and Hrishi, in quite some detail.
- **Truncated Taylor Series**: We shall describe this algorithm in some depth, before moving on with quantum walks.
- **Quantum Walks**: The main focus of our project which we describe below in quite some detail.
- **Quantum Signal Processing**: This procedure gives us the best known theoretical bounds through a framework called qubitization.

Now, for implementation purposes over near term NISQ models of quantum computation, Trotterization appears to be the way to go, even though it is theoretically slower with respect to algorithms such as qubitization. This is because it is simpler to implement and uses way less ancilla qubits.

Now, it is also worth noting that theoretically Trotterization in itself is not inefficient. On the contrary, it is quite efficient achieving a complexity only slightly superlinear in $t$. Why is that good? Well, because the no-fast-forwarding theorem implies that simulation in sublinear time is not possible.

#### 3.0.1 No-fast-forwarding Theorem

Fast forwarding is the ability to simulate (using a quantum computer) the evolution of a given system governed by a certain Hamiltonian $H$ to within time $t$, but such that the simulation takes time which is much less than $t$.

The no-fast-forwarding theorem states that fast forwarding any physically realistic Hamiltonian (which can be efficiently simulated by a quantum circuit) is highly unlikely. This is proven by showing that even for 2-sparse row-computable Hamiltonians the possibility of exponential fast-forwarding implies $\mathsf{BQP} = \mathsf{PSPACE}$.

### 3.1 Complexity comparision for sparse Hamiltonian simulation

| *Method* | *Query Complexity* | *Gate Complexity* |
|---|---|---|
| Lee-Suzuki Trotter | $O\left(d^3 t \left(\frac{dt}{\epsilon}\right)^{\frac{1}{2}k}\right)$ | $O\left(\frac{t^2}{\epsilon}\right)$ |
| Truncated Taylor Series | $O\left(\frac{d^2\|\|H\|\|_{max} \log \frac{d^2\|\|H\|\|_{max}}{\epsilon}}{\log\log \frac{d^2\|\|H\|\|_{max}}{\epsilon}}\right)$ | $O\left(\frac{t \log\left(\frac{t}{\epsilon}\right)}{\log\log \frac{t}{\epsilon}}\right)$ |
| Quantum Walks | $O\left(d\|\|H\|\|_{max} \frac{t}{\sqrt{\epsilon}}\right)$ | $O\left(\frac{t}{\sqrt{\epsilon}}\right)$ |
| Qubitisation | $O\left(td\|\|H\|\|_{max} + \frac{\log \frac{1}{\epsilon}}{\log\log \frac{1}{\epsilon}}\right)$ | $O\left(t + \log \frac{1}{\epsilon}\right)$ |

### 3.2 Unsolved questions

1. How well do the bounds reflect upon implementation over NISQ frameworks?
2. Quantum walks serve as a framework for universal computation. In that case, can we leverage Hamiltonian simulation using quantum walks to solve unrelated graph theoretic and combinatorial optimization problems?
3. The problem of efficient decomposition is required to be tackled for general cases.

4. Given possible frameworks for Hamiltonian simulation and a given Hamiltonian with a favoured decomposition, which framework will be most suitable for simulating it? Can this problem be solved using decision trees/random forests by keeping calculating information gained about the original Hamiltonian by studying how it is deemed to affect chosen sets of qubits?
5. How to deal with non-sparse Hamiltonians?

# 4 TRUNCATED TAYLOR SERIES

Any Hamiltonian can be decomposed as a linear combination of unitary matrices. Now, let $H$ be some given Hamiltonian. Then, we have the following:

- $H = \sum_{j=1}^{n} \alpha_j H_j$
- Here, each $H_j$ is unitary and a mechanism is available for implementing that unitary.
- Local Hamiltonians: These can be decomposed into a sum of tensor products of Pauli matrices where each term acts nontrivially on a constant number of qubits.
- We can use this algorithm, broadly, for all families of sparse Hamiltonians.

## 4.1 The Simulation

Consider $U(t) = e^{-iHt}$ and we wish to simulate this evolution within error $\epsilon$.

- Divide the evolution time into $r$ segments.
- Within each segment we have $U_r = e^{-iHt/r} \approx \sum_{k=0}^{K} \frac{1}{k!}(-iHt/r)^k$, where the Taylor series is truncated to $K$.
- For $U_r$, the corresponding error should be $\leqslant \epsilon/r$.
- Given, $r \geqslant \|H\|t$, we can choose $K = O(\frac{\log(r/\epsilon)}{\log\log(r/\epsilon)})$.
- Overall complexity $\approx rK$.

$$\begin{aligned} U_r &= \sum_{k=0}^{K} \frac{(-iHt/r)^k}{k!} \\ &= \sum_{k=0}^{K} \sum_{l_1, l_2, \ldots, l_k = 1}^{L} \frac{(t/r)^k}{k!} \alpha_{l1}, \ldots, \alpha_{lk} (-i)^k H_{l1} \ldots H_{lk} \\ &= \sum_{j=0}^{J} \beta_j V_j \end{aligned}$$

Here, $U_r$ is a LCU of $V_j = (-i)^k H_{l1} \ldots H_{lk}$ with the coefficients $\beta_j$ described above. Now, without loss of generality we can set each of $\alpha_l \geqslant 0 \implies \beta_j > 0$. Note that, $U_r = \tilde{U}(t/r)$ where $\tilde{U}$ denotes truncation.

$$\|U(\tilde{t/r}) - U(t/r)\| \leqslant e^{\alpha t/r} \frac{(\alpha t/r)^{K+1}}{(K+1)!}$$

Now, here we have the following as well.

- $\alpha = \sum_l \alpha_l$
- Given $\epsilon$, $K = O\left(\frac{\log \frac{\alpha t}{\epsilon}}{\log\log \frac{\alpha t}{\epsilon}}\right)$
- Required complexity = $O(rK)$

### 4.2 Implementation

In order to be able to implement the given sum of unitary operators, we introduce a few subroutines. The first subroutine we introduce is the SELECT such that $\text{SELECT}(V)\,|j\rangle\,|\psi\rangle = |j\rangle\,V_j\,|\psi\rangle$. Thus, SELECT serves as some form of reflection.

Now, in order to simulate $\tilde{U}$, we first introduce a $J+1$ dimensional unitary $B$ to prepare the required ancillary state (where $s = \sum_{j=0}^{J} \beta_j$).

$$B\,|0\rangle = \frac{1}{\sqrt{s}} \sum_{j=0}^{J} \sqrt{\beta_j}\,|j\rangle$$

Now, let us define the operator $W = (B^\dagger \otimes \mathbb{I})\,\text{SELECT}(V)(B \otimes \mathbb{I})$ then we have the following for some $|\phi\rangle$ whose ancillary state is supported in the subspace orthogonal to $|0\rangle$.

$$W\,|0\rangle\,|\psi\rangle = \frac{1}{s}\,|0\rangle\,\tilde{U}\,|\psi\rangle + \sqrt{1-\frac{1}{s^2}}\,|\phi\rangle \implies P(W\,|0\rangle\,|\psi\rangle) = \frac{1}{s}\,|0\rangle\,\tilde{U}\,|\psi\rangle$$

Here, $P$ denoted the projection operator $|0\rangle\langle 0| \otimes \mathbb{I}$. The value of $s$ can be adjusted by choosing the size of the segments. Now, using the *Approximate segment lemma* and *Oblivious amplitude amplification*, we obtain that if $\tilde{U}$ is unitary then it can be implemented using $W$, $W^\dagger$ and with the $(J+1)$-dimensional ancilla reflection $R = 1 - 2P$.

To conclude, if $\tilde{U}$ is unitary and $A = -WRW^\dagger RW$ then, $A\,|0\rangle\,|\psi\rangle = |0\rangle\,\tilde{U}\,|\psi\rangle$.

### 4.3 Possible future work

Now, it is worth noting that in this formulation, the truncation error exists even if the Hamiltonian consists of mutually commutative terms. Thus, we can ideally come up with faster and less error prone algorithms by further exploiting commutative and anti-commutative relations.

## 5 TIME-DEPENDENT HAMILTONIANS

We observe time dependent Hamiltonians when our system has external classical control on it. then, we can not represent the corresponding unitary evolution as a closed form expression. Thus, $U$ remains as following.

$$U(t) = e^{-i\int_0^t H(s)}$$

For this time-dependent case, we do have an infinite-series representation - an analog for Taylor series in the time-independent case. This representation is called the Dyson series expression.

### 5.1 Dyson expansion of $U$

$$U(t) = 1 + (-i)\int_0^t dt_1 H(t_1) + (-i)^2 \int_0^t \int_0^{t_1} dt_1\,dt_2 H(t_1)H(t_2) + \dots$$

Note that, in this case ordering highly matters because when we are considering $H(t)$, it may not always commute with itself (because it is time dependent).

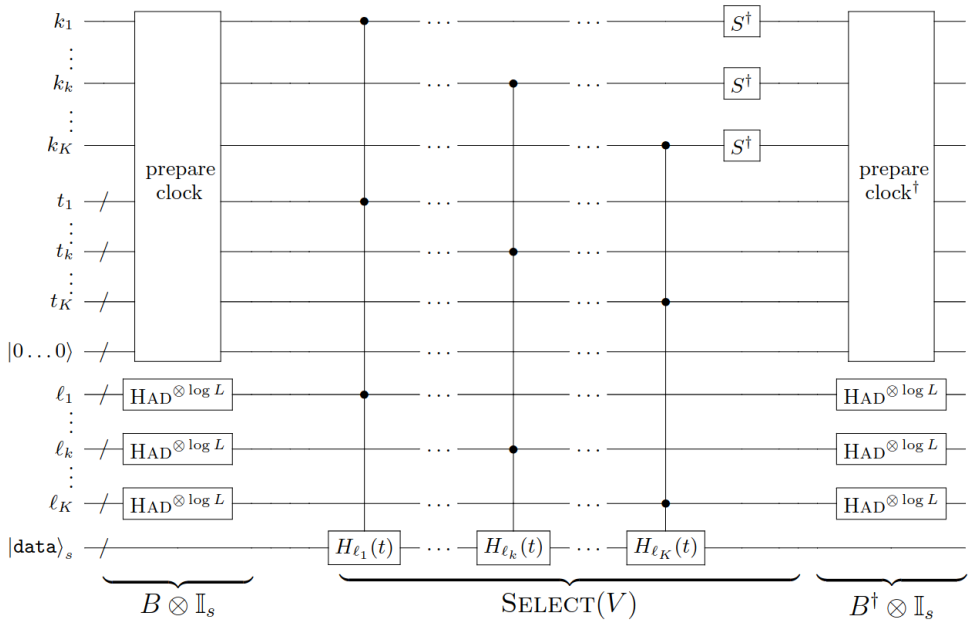



**Figure 1:** Truncated Dyson series implementation for time-dependent Hamiltonian simulation

# 6 QUANTUM WALKS

Quantum walks, the quantum mechanical counterpart of classical random walks are an advanced tool for building quantum algorithms that have been recently shown to constitute a universal model of quantum computation. In contrast to the classical random walk, where the walker occupies definite states and the randomness arises due to stochastic transitions between states, in quantum walks randomness arises through:

1. Quantum superposition of states,
2. Non-random, reversible unitary evolution
3. Collapse of the wave function due to state measurements.

The quantum dynamics of any discrete system can be captured by a quantum walk on a network, which is a universal model for quantum computation. Besides being a useful primitive to design quantum algorithms, quantum walks are a powerful tool to model transport in quantum systems such as the transfer of excitations in light-harvesting systems. Studying the long-time dynamics of quantum walks on networks is crucial to the understanding of these diverse problems.

## 6.1 Szegedy Quantum Walk

Szegedy defined a discrete-time quantum walk corresponding to an arbitrary discrete-time classical Markov chain in [**??**]. Based on his work, Andrew Childs described a discrete-time quantum walk corresponding to a general $N \times N$ Hermitian matrix $H$.

First, Childs fixed an orthonormal basis $\{|j\rangle : j = 1, 2, \ldots, N\}$ of $\mathbb{C}^N$, and

$$\text{abs}(H) = \sum_{j,k=1}^{N} \left|H_{jk}\right| |j\rangle \langle k|$$

is the elementwise absolute value of H. Now we define the principal eigenvector of abs(H), i.e., the eigenvector with eigenvalue ||abs(H)|| to be

$$|d\rangle = \sum_{j=1}^{N} d_j |j\rangle$$

Now Childs defines N quantum states $|\psi_1\rangle, \ldots, |\psi_N\rangle \in \mathbb{C}^N \times \mathbb{C}^N$ as

$$|\psi_j\rangle = \frac{1}{\sqrt{||abs(H)||}} \sum_{k=1}^{N} \sqrt{H^*_{jk} \frac{d_k}{d_j}} |j,k\rangle$$

**Theorem 1.** *The quantum states $|\psi_1\rangle, \ldots, |\psi_N\rangle$ are orthonormal.*

*Proof.*

$$\langle\psi_j|\psi_k\rangle = \frac{1}{||abs(H)||} \sum_{l_1,l_2=1}^{N} \sqrt{H_{jl_1} H^*_{kl_2} \frac{d_{l_1} d_{l_2}}{d_j d_k}} \langle j,l_1|k,l_2\rangle$$

Thus, it is clear that $\langle\psi_j|\psi_k\rangle = 0$ when $j \neq k$, that is the quantum states are orthogonal.

$$\begin{aligned}
\langle\psi_j|\psi_j\rangle &= \frac{1}{||abs(H)||} \sum_{k_1,k_2=1}^{N} \sqrt{H_{jk_1} H^*_{jk_2} \frac{d_{k_1} d_{k_2}}{d_j^2}} \langle j,k_1|j,k_2\rangle \\
&= \frac{1}{||abs(H)||} \sum_{k=1}^{N} |H_{jk}| \frac{d_k}{d_j} \\
&= 1, \quad \text{since } abs(H)\,|d\rangle = ||abs(H)||\,|d\rangle
\end{aligned}$$

Hence, the quantum states are orthonormal. □

### 6.1.1 Swap Operator

The function of the swap operator S is

$$S\,|j,k\rangle = |k,j\rangle$$

Thus, $S = S^\dagger$ and $SS^\dagger = S^2 = \mathbb{I}$.

### 6.1.2 Discrete Time Quantum Walk

Childs defines the discrete-time quantum walk (Szegedy) corresponding to H as the unitary operator obtained by first reflecting about $\text{span}\{|\psi_j\rangle\}$ and then exchanging the two registers with the swap operator S. To formally define this, we first define the isometry mapping from $|j\rangle \in \mathbb{C}^N$ to $|\psi_j\rangle \in \mathbb{C}^N \times \mathbb{C}^N$ as

$$T = \sum_{j=1}^{N} |\psi_j\rangle \langle j|$$

Then $TT^\dagger$ is the projector onto $\text{span}\{|\psi_j\rangle\}$. And thus the walk operator is defined as $S(2TT^\dagger - \mathbb{I})$

**Theorem 2.** *Suppose that $\frac{H}{||abs(H)||}|\lambda\rangle = \lambda|\lambda\rangle$. Then the unitary operator*

$$U = iS(2TT^\dagger - \mathbb{I})$$

*has two normalised eigenvectors*

$$|\mu_\pm\rangle = \frac{\mathbb{I} - e^{\pm i \arccos\lambda} S}{\sqrt{2(1-\lambda^2)}} T\,|\lambda\rangle$$

*with eigenvalues $\mu_\pm = \pm e^{\pm i \arcsin\lambda}$.*

*Proof.* Considering the action of $U$ on the vector $T|\lambda\rangle$, we have

$$\begin{aligned} UT|\lambda\rangle &= iS(2TT^\dagger - \mathbb{I})T|\lambda\rangle \\ &= 2iSTT^\dagger T|\lambda\rangle - iST|\lambda\rangle \\ &= iST|\lambda\rangle \qquad\qquad (\text{since } T^\dagger T = \mathbb{I}) \end{aligned}$$

Considering the effect of the swap operator on the inner product of two of our orthonormal states we get

$$\begin{aligned} \langle\psi_j|S|\psi_k\rangle|\psi_j|S|\psi_k\rangle &= \frac{1}{||abs(H)||}\sum_{l_1,l_2=1}^{N}\sqrt{H_{jl_1}H^*_{kl_2}\frac{d_{l_1}d_{l_2}}{d_j d_k}}\,\langle j,l_1|l_2,k\rangle \\ &= \frac{H_{jk}}{||abs(H)||} \end{aligned}$$

Thus, we also get the following.

$$\begin{aligned} T^\dagger ST &= \sum_{j,k=1}^{N}|j\rangle\langle\psi_j|S|\psi_k\rangle\langle k| \\ &= \frac{H}{||abs(H)||} \end{aligned}$$

And now, we have:

$$\begin{aligned} UST|\lambda\rangle &= iS(2TT^\dagger - \mathbb{I})ST|\lambda\rangle \\ &= 2iSTT^\dagger ST|\lambda\rangle - iS^2T|\lambda\rangle \\ &= 2\lambda iST|\lambda\rangle - iT|\lambda\rangle \end{aligned}$$

The unnormalised eigenvector, let it be $|\mu\rangle$, is

$$\begin{aligned} |\mu\rangle &= \left(\mathbb{I} - e^{i\arccos\lambda}S\right)T|\lambda\rangle \\ &= \left(\mathbb{I} - ie^{i\arcsin\lambda}S\right)T|\lambda\rangle \\ &= T|\lambda\rangle + i\mu ST|\lambda\rangle \end{aligned}$$

Thus, we can now obtain an expression by equating $U|\mu\rangle = \mu|\mu\rangle$

$$\begin{aligned} U|\mu\rangle = \mu|\mu\rangle &\iff UT|\lambda\rangle + i\mu UST|\lambda\rangle = \mu T|\lambda\rangle + i\mu^2 ST|\lambda\rangle \\ &\iff iST|\lambda\rangle + i\mu\left(2\lambda iST|\lambda\rangle - iT|\lambda\rangle\right) = \mu T|\lambda\rangle + i\mu^2 ST|\lambda\rangle \\ &\iff i(1+2i\lambda\mu)ST|\lambda\rangle = i\mu^2 ST|\lambda\rangle \\ &\iff \mu^2 = 1 + 2i\lambda\mu \end{aligned}$$

Thus, solving the quadratic equation obtained, we obtain the unnormalised eigenvalues

$$\mu = i\lambda \pm \sqrt{1-\lambda^2} = ie^{\arccos\lambda} = \pm e^{\arcsin\lambda}$$

And the normalisation factor is

$$\begin{aligned} \langle\mu|\mu\rangle &= \left\langle\lambda|T^\dagger T|\lambda\right\rangle + i\mu\left\langle\lambda|T^\dagger ST|\lambda\right\rangle - i\mu^*\left\langle\lambda|T^\dagger S^\dagger T|\lambda\right\rangle + |\mu|^2\left\langle\lambda|T^\dagger S^\dagger ST|\lambda\right\rangle \\ &= 1 + i\mu\lambda - i\mu^*\lambda + |\mu|^2 \\ &= 2(1-\lambda^2), \quad \text{since } \mu = i\lambda \pm \sqrt{1-\lambda^2} \end{aligned}$$

Hence, by combining these results we get our the expressions for eigenvalues and eigenvectors that we wanted to prove. □

# 7 HAMILTONIAN SIMULATION USING QUANTUM WALK

No-fast-forwarding theorem states that the optimal number of steps required to simulate an evolution for time $t$ is linear in $t$. This can be done using Szegedy Quantum Walks and phase estimation.

Firstly, Childs mentioned the representation of isometry mapping on eigenvector of $\frac{H}{||abs(H)||}$ in terms of eigenvectors of the quantum walk operator $U$. We derive this expression. Let

$$T|\lambda\rangle = x|\mu_+\rangle + y|\mu_-\rangle$$

Then,

$$\begin{aligned} T|\lambda\rangle &= x|\mu_+\rangle + y|\mu_-\rangle \\ &= \frac{x+y}{\sqrt{2(1-\lambda^2)}} T|\lambda\rangle - \frac{xe^{i\arccos\lambda} + ye^{-i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}} ST|\lambda\rangle \end{aligned}$$

Thus, we get a system of linear equations as follows.

$$x+y = \sqrt{2(1-\lambda^2)}, xe^{i\arccos\lambda} + ye^{-i\arccos\lambda} = 0$$

And we solve this system to get the coefficients to be

$$x = \frac{1-\lambda e^{-i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}, y = \frac{1-\lambda e^{i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}$$

and from that we have,

$$T|\lambda\rangle = \frac{1-\lambda e^{-i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_+\rangle + \frac{1-\lambda e^{i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_-\rangle$$

Thus, for any state $|\psi\rangle \in \mathbb{C}^N \times \mathbb{C}^N$, we can write $T|\psi\rangle$ as

$$\begin{aligned} T|\psi\rangle &= \sum_\lambda \psi_\lambda T|\lambda\rangle \\ &= \sum_\lambda \psi_\lambda \left( \frac{1-\lambda e^{-i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_+\rangle + \frac{1-\lambda e^{i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_-\rangle \right) \end{aligned}$$

To simulate $H$ using the corresponding discrete-time quantum walk operator $U$ a phase of $e^{i\lambda t}$ is to be introduced for the $\lambda$ term of the superposition observed in the above expression. The procedure used for simulation is:

1. Apply $T$ to the input state $|\psi\rangle$
2. Perform phase estimation of $U$, estimating a phase $\pm e^{\pm i\arcsin\lambda}$ for the component $T|\lambda\rangle$
3. Use estimate of $\arcsin\lambda$ to get estimate of $\lambda$, $\tilde{\lambda}$.
4. Induce the phase $e^{-i\tilde{\lambda}t}$ and uncompute $\tilde{\lambda}$ by performing phase estimation in reverse.
5. Finally apply reverse isometry $T^\dagger$

Now for an eigenstate-eigenvalue pair $|\theta\rangle, e^{i\theta}$ of the discrete-time quantum walk, let $P$ denote the isometry that performs phase estimation on this walk. Then $P$ can be defined as follows

$$P|\theta\rangle = \sum_\phi a_{\phi|\theta}|\theta,\phi\rangle$$

where $a_{\phi|\theta}$ is the amplitude for the estimate $\phi$. And let $F_t$ be the unitary operation that applies the desired phase to a given value-estimate state. That is,

$$F_t |\theta, \phi\rangle = e^{-it \sin\phi} |\theta, \phi\rangle$$

Childs claims that the simulation of Hamiltonian evolution $e^{iHt}$ is $T^\dagger P^\dagger F_t P T$. To prove this claim, Child computes the inner product between the expected evolved state, $e^{-iHt}|\psi\rangle$, and the simulated evolved state, $T^\dagger P^\dagger F_t P T |\psi\rangle$. First, it is observed that

$$\begin{aligned}
PTe^{-iHt}|\lambda\rangle &= PT\left(\sum_{\lambda'} e^{-i\lambda' t} |\lambda'\rangle\langle\lambda'|\right)|\lambda\rangle \\
&= e^{-i\lambda t} PT|\lambda\rangle \\
&= e^{-i\lambda t} P\left(\frac{1-\lambda e^{-i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_+\rangle + \frac{1-\lambda e^{i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_-\rangle\right) \\
&= e^{-i\lambda t}\sum_\phi \left(a_{\phi|\arcsin\lambda}\frac{1-\lambda e^{-i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_+,\phi\rangle + \right. \\
&\qquad \left. a_{\phi|\pi-\arcsin\lambda}\frac{1-\lambda e^{i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_-,\phi\rangle\right)
\end{aligned}$$

And thus,

$$\begin{aligned}
F_t PT|\lambda\rangle = \sum_\phi e^{-it\sin\phi}\left(a_{\phi|\arcsin\lambda}\frac{1-\lambda e^{-i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_+,\phi\rangle + \right. \\
\left. a_{\phi|\pi-\arcsin\lambda}\frac{1-\lambda e^{i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}|\mu_-,\phi\rangle\right)
\end{aligned}$$

Using orthonormality of eigenstates, and the previous observation, we compute the required inner product.

$$\begin{aligned}
\left\langle\psi|e^{iHt}T^\dagger P^\dagger F_t PT|\psi\right\rangle &= \sum_\lambda |\psi_\lambda|^2 \left\langle\lambda|e^{iHt}T^\dagger P^\dagger F_t PT|\lambda\right\rangle \\
&= \sum_{\lambda,\phi}|\psi_\lambda|^2 e^{i(\lambda-\sin\phi)t}\left(\left|a_{\phi|\arcsin\lambda}\right|^2\left|\frac{1-\lambda e^{-i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}\right|^2 + \right. \\
&\qquad \left.\left|a_{\phi|\pi-\arcsin\lambda}\right|^2\left|\frac{1-\lambda e^{i\arccos\lambda}}{\sqrt{2(1-\lambda^2)}}\right|^2\right) \quad (\langle\mu_+|\mu_-\rangle = 0) \\
&= \sum_{\lambda,\phi}|\psi_\lambda|^2 e^{i(\lambda-\sin\phi)t}\frac{\left|a_{\phi|\arcsin\lambda}\right|^2 + \left|a_{\phi|\pi-\arcsin\lambda}\right|^2}{2}
\end{aligned}$$

Thus, the fidelity of this simulation is

$$\begin{aligned}
\left|\left\langle\psi|e^{iHt}T^\dagger P^\dagger F_t PT|\psi\right\rangle\right| &\geqslant \min_\lambda \sum_\phi \cos((\lambda-\sin\phi)t)\frac{\left|a_{\phi|\arcsin\lambda}\right|^2 + \left|a_{\phi|\pi-\arcsin\lambda}\right|^2}{2} \\
&\geqslant \min_{\theta\in[0,2\pi)}\sum_\phi \cos((\sin\theta-\sin\phi)t)\left|a_{\phi|\theta}\right|^2 \\
&\geqslant 1 - \min_{\theta\in[0,2\pi)}\frac{1}{2}\sum_\phi((\sin\theta-\sin\phi)t)^2|a_{\phi|\theta}|^2 \\
&\geqslant 1 - \frac{t^2}{2}\min_{\theta\in[0,2\pi)}\sum_\phi(\theta-\phi)^2|a_{\phi|\theta}|^2
\end{aligned}$$

and this holds because, $|\sin\theta - \sin\phi| \leqslant |\theta - \phi|$.

To further quantify the fidelity of the simulation, evaluation of performance of phase estimation is required. In standard phase estimation modulo M, we begin from the state of equal superposition, that is $\frac{1}{\sqrt{M}}\sum_{x=0}^{M-1}|x\rangle$. Although there are some other states that, starting from whom give better results. One particular state that Childs mentioned is

$$\sqrt{\frac{2}{M+1}}\sum_{x=0}^{M-1}\sin\left(\frac{\pi(x+1)}{M+1}\right)|x\rangle$$

He claims, and goes on to prove that this initial state minimizes the variance of the estimate and gives the best bound for the fidelity of the simulation.

The probability distribution of estimates of $\theta$ with this initial state is

$$\left|a_{\theta+\Delta|\theta}\right|^2 = \frac{\cos^2\left(\Delta\frac{M+1}{2}\right)\sin^2\left(\frac{\pi}{M+1}\right)}{2M(M+1)\sin^2\left(\frac{\Delta}{2}+\frac{\pi}{2(M+1)}\right)\sin^2\left(\frac{\Delta}{2}-\frac{\pi}{2(M+1)}\right)}$$

where the estimated phase is $\frac{2\pi j}{M} = \theta + \Delta$ for some integer j. A range of angles can be chosen such that $\Delta = \Delta_0 + \frac{2\pi j}{M}$, where $0 \leqslant \Delta_0 \leqslant \frac{2\pi}{M}$ and $-\lceil\frac{M}{2}\rceil + 1 \leqslant j \leqslant \lfloor\frac{M}{2}\rfloor$.

With these assumptions, and for sufficiently large M, we have the following.

$$\begin{aligned}
|a_{\theta+\Delta|\theta}|^2 &\leqslant \frac{\pi^2}{2M^4\sin^2\left(\frac{\Delta}{2}+\frac{\pi}{2(M+1)}\right)\sin^2\left(\frac{\Delta}{2}-\frac{\pi}{2(M+1)}\right)} \\
&\leqslant \frac{128\pi^2}{M^4\Delta^4\left(1-\frac{\pi^2}{\Delta^2(M+1)^2}\right)^2} \\
&\leqslant \frac{512\pi^2}{9M^4\Delta^4}
\end{aligned}$$

Here, the last step assumes that $\Delta \geqslant 2\pi/M$. Now, since we have $2\pi j/M \leqslant \Delta \leqslant 2\pi(j+1)/M$, the following holds.

$$\begin{aligned}
\sum_{\phi}(\theta-\phi)^2|a_{\phi|\theta}|^2 &\leqslant \frac{4\pi^2}{M^2} + 2\sum_{j=1}^{\infty}\frac{128(j+1)^2}{9M^2j^2} \\
&= \frac{4\pi^2}{M^2} + \frac{256}{9M^2}\left(\frac{15\pi^2+\pi^4+180\zeta(3)}{90}\right) \\
&\leqslant \frac{186}{M^2}
\end{aligned}$$

Thus, using the fidelity bound, we obtain a simulation fidelity of at least $1 - 93t^2/M^2$. Thus we find that $M = O(t/\sqrt{\delta})$ steps suffice to obtain fidelity at least $1-\delta$.

## 8 EXAMPLE IMPLEMENTATION

We will explain the quantum walk approach using a toy Hamiltonian during the presentation. The explanation will also be added here while submitting the final draft.

# 9 QUBITIZATION

Qubitization provides us with a general framework to understand a high-dimensional space as the direct sum of a large number of 2-dimensional spaces on which we act simultaneously and without their interfering with each other.

Qubitization is found to perform better than older Hamiltonian simulation techniques for most cases. But, its use cases are not constrained to the simulation of Hamiltonians. It can be applied to understand and discover new algorithms and properties especially in the field of quantum walks.

## 9.1 Description

Before we delve into qubitization, we should note that in quantum computing, embedding in a 2-dimensional space is computationally expensive. A single auxiliary qubit is enough to double the dimension of the vector space used to represent the quantum information being processed. This is way treating qubits classically results in dealing with exponential space since we require a vector space of dimension $2^n$ to represent a quantum state defined by $n$ qubits.

### *9.1.1 About the routine*

- Qubitization is based on the block encoding technique.

$$U = \begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$

  with the top-left block of size $2^n \times 2^n$ and the bottom-right block of size $2^n \times 2^n$.

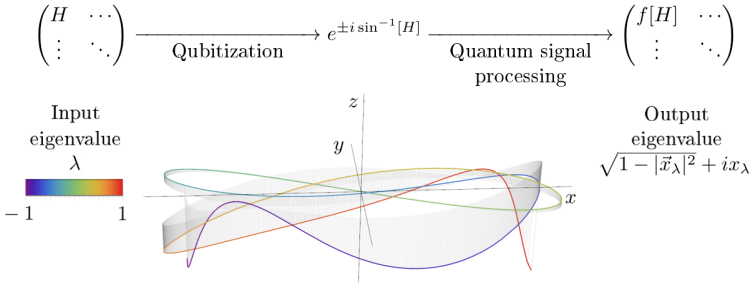
- It relies on the decomposition of the $2^{n+1}$ dimensional space on which $U$ acts into $2^n$ spaces, each of dimension 2.
- Each such space of dimension 2 is generated by an eigenvector of $A$ and another associated vector.
- Thus, qubitization refers to this decomposition in spaces of dimension 2.
- Each eigenspace of $A$ is immersed in a space of dimension 2, i.e. in the qubit space.
- Let $\tilde{U}$ be the result of qubitizing $U$.

  $$\tilde{U} = \oplus_{\lambda \in \Lambda(A)} W_{\text{qubit},\lambda}$$

  Thus, $\tilde{U}$ acts as the direct sum of $2^n$ unitary matrices of dimension 2. We have one unitary of dimension 2 per eigenvalue of $A$.
- For the purpose of Hamiltonian simulation, $A$ will be some Hamiltonian, $H$.
- $W_{\text{qubit},\lambda}$ is a $2 \times 2$ diagonalizable unit matrix. Hence, $\forall\, \lambda_W \in \Lambda(W)$, $|\lambda_W| = 1$. Furthermore, $\lambda_W$s are determined by $\lambda \in \Lambda(H)$. $W$ is referred to as the walk operator.

  $$W_{\text{qubit},\lambda} \approx \begin{bmatrix} e^{i \cos^{-1} \lambda} & 0 \\ 0 & e^{-i \cos^{-1} \lambda} \end{bmatrix}$$

- Now, the next step involves transforming the eigenvalues of the walk operator from $e^{\pm i \cos^{1} \lambda}$ to $e^{i\lambda t}$.
- This above transformation is possible by using any QSVT method (in the given diagram, we can see that QSP has been used).
- This method is suitable for finding static quantities such as the ground state of the associated $H$ (Hamiltonian).
- For estimating ground state, we can just apply QPE directly to $W$.




## 9.2 Wow qubitization rocks! Right? ... umm it does right?

Now that we have a vague idea of how to go about implementing qubitization, it is worth noting that we require a few (2) subroutines (namely SELECT and PREPARE) to serve as interfaces for the Hamiltonian.

Now unlike as in Trotterization, these subroutines are quantum in nature (and not classical). Thus, they require ancilla qubits (to be exact, logarithmically more qubits when compared to Trotterization) and hence might be infeasible for useful implementations in extremely near term NISQ-era. This same problems persists, to some extent, in the case of "Truncated Taylor Series" method and "Szegedy Quantum Walk", as well.

However that said, qubitization is way better in simulating chemistry problems where there do not exist any self evident symmetries that can be taken advantage of.

## 9.3 Implementation

Let $H = \sum_j h_j H_j$ for unitary and Hermitian $H_j$ and $h_j \geqslant 0$. Thus, we need to implement $W = e^{\pm i \cos^{-1}(H/|h|_1)}$ where $|h|_1 = \sum_j |h_j|$.

Now, consider the following subroutines:

- **Reflection**: We have $\text{SELECT} \, |j\rangle \, |\psi\rangle = |j\rangle \, H_j \, |\psi\rangle$. Now, $\text{SELECT}^2 \, |j\rangle \, |\psi\rangle = |j\rangle \, |\psi\rangle$ and the operator has eigenvalues $\pm 1$.
- Purpose of SELECT is to be able to access each of the $H_j$s.
- Now, we shall define the operator PREPARE to access the coefficients $h_j$s.

$$\text{PREPARE} \, |0\rangle = \sum_j \sqrt{\frac{h_j}{|h|_1}} \, |j\rangle$$

- Let us define $\mathcal{M}$ as follows.

$$\mathcal{M}\left|0\right\rangle^{\otimes n} = \begin{cases} -\left|x\right\rangle & \text{if } x = 0 \\ \left|x\right\rangle & \text{otherwise} \end{cases}$$

- Thus, $\mathcal{M}$ is a multiply controlled phase gate.
- **Another reflection**: $R = 1 - 2\,\text{PREPARE}\left|0\right\rangle\left\langle 0\right|\text{PREPARE}^{\dagger}$.
- Thus, we have $R = \text{PREPARE}(\mathcal{M})\,\text{PREPARE}^{\dagger}$

Thus, the walk operatoor can be implemented as $W = (\text{SELECT})\ (R)$ which again can be seen to implement an operator that is equivalent (up to an isometry) to $e^{\pm i\cos^{-1}(H/|h|_1)}$.

## REFERENCES


[1] Richard Feynman. Simulating physics with computers.

[2] Wikipedia contributors. Quantum walk — Wikipedia, the free encyclopedia, 2022. [Online; accessed 28-April-2022].

[3] Salvador Elías Venegas-Andraca. Quantum walks: a comprehensive review. *Quantum Information Processing*, 11(5):1015–1106, jul 2012.

[4] Shantanav Chakraborty, Kyle Luh, and Jeremie Roland. How Fast Do Quantum Walks Mix? *Physical Review Letters*, 124(5), feb 2020.

[5] Andrew M. Childs. On the Relationship Between Continuous and Discrete Time Quantum Walk. *Communications in Mathematical Physics*, 294(2):581–603, oct 2009.

[6] Dominic W. Berry and Andrew M. Childs. Black-Box Hamiltonian Simulation and Unitary Implementation. *Quantum Info. Comput.*, 12(1–2):29–62, jan 2012.

[7] M. Szegedy. quantum speed-up of markov chain based algorithms. In *45th Annual IEEE Symposium on Foundations of Computer Science*.

[8] Laura Clinton, Johannes Bausch, and Toby Cubitt. Hamiltonian simulation algorithms for near-term quantum hardware. *Nature Communications*, 12(1), aug 2021.

[9] Yulong Dong, K. Birgitta Whaley, and Lin Lin. A Quantum Hamiltonian Simulation Benchmark, 2021.

[10] Guang Hao Low and Isaac L. Chuang. Hamiltonian Simulation by Qubitization. *Quantum*, 3:163, July 2019.

[11] Dominic W. Berry, Andrew M. Childs, and Robin Kothari. Hamiltonian simulation with nearly optimal dependence on all parameters. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. IEEE, oct 2015.

[12] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu. Theory of trotter error with commutator scaling. *Physical Review X*, 11(1), feb 2021.

[13] Jonathan Wei Zhong Lau, Tobias Haug, Leong Chuan Kwek, and Kishor Bharti. Nisq algorithm for hamiltonian simulation via truncated taylor series, 2021.

[14] Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma. Exponential improvement in precision for simulating sparse hamiltonians. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*. ACM, may 2014.