TROTTERIZATION

ZEESHAN AHMED, SHREYAS PRADHAN, HRISHI NARAYANAN

CONTENTS

Simulation of quantum systems is one of the most fruitful and promising domains of quantum computing in the near future. Due to a limited array of gates that are available, one needs to find efficient ways of approximating the system that is to be simulated. Trotterization is one of the procedures that can be used to construct efficient circuits for the simulation of a class of Hamiltonians. We present an overall study to understand Trotterization, its different variants, and a brief understanding of Trotter Error presented by Childs et al.

1 HAMILTONIAN SIMULATION & ITS IMPORTANCE

First established by Richard Feynman, Hamiltonian Simulation was established as a requirement to simulate nature and perform experiments.

> "...nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem because it doesn't look so easy" - Richard Feynman

The universe is of quantum nature, which means that to simulate particles or any other natural entities you need to simulate its Hamiltonian. However, efficient computation of evolution given the Hamiltonian has not been achieved for a general Hamiltonian.

Due to the discrete nature of classical computers, not all Hamiltonian Simulations can be simulated efficiently. This gives quantum computers an edge since they can work with qubits to match the quantum nature of the required system and simulate the dynamics. This makes Hamiltonian Simulation one of the most impacting and significant contributions of quantum computing.

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

1.1 Applications

Although it has been stated that Hamiltonian Simulation is BQP-complete, there are problems that require the direct use of Hamiltonian Simulation without reduction. The most notable and impacting problem is that of Ground-state preparation.

1.1.1 *Ground-state Preparation*

Consider Hamiltonian H and it's spectral decomposition $H = \sum_{i=0}^{n} \lambda_i |\psi_i\rangle \langle\psi_i|$. Without loss of generality we can arrange the terms in a way that they satisfy the condition $\lambda_i \leqslant \lambda_{i+1}$. The state $|\psi_0\rangle$ is referred to as the ground-state and it's corresponding eigenvalue λ_0 is the ground energy.

Constructing the ground-state and further using it to estimate the ground energy are very important problems in condensed matter physics, quantum chemistry and quantum information. This problem of calculating ground energy further opens up a large number of useful applications of Hamiltonian Simulations.

1.1.2 *Protein Folding*

Protein folding involves the physical process by which a protein chain is translated to its native three-dimensional structure, typically a "folded" conformation by which the protein becomes biologically functional. However, finding this folded conformation is an NP-Hard problem.

Since the protein itself is a quantum system, the evolution of the protein into its native state will be described by some hamiltonian H. Hence, finding the native structure of the protein corresponds to the problem of finding the ground state of H.

This has a lot of significance in speeding up drug discovery.

1.1.3 Graph Coloring

Graph coloring is a way of labeling the vertices of a graph with colors such that no two adjacent vertices are assigned the same color. Finding the chromatic number of the graph, ie, the smallest number of colours required to colour a graph, is in general NP-hard. This is modelled as a minimisation problem with a cost function of the form:

$$f(x) = C \sum_{v=1}^{n} (1 - \sum_{i=1}^{k} x_{v,i})^2 + D \sum_{v,w=1}^{n} \sum_{i=1}^{k} A_{vw} x_{v,i} x_{w,i}$$

This cost function can be mapped to a hamiltonian of the form [8]:

$$H = C \sum_{v=1}^{n} (2I - \sum_{i=1}^{k} (1 - \sigma_{v,i}^{z})) + D \sum_{v,w=1}^{n} \sum_{i=1}^{k} A_{vw} (I - \sigma_{v,i}^{z})(I - \sigma_{w,i}^{z})$$

Hence, the problem of finding the chromatic number of a graph (minimising the cost function) can be mapped to finding the ground state energy of the corresponding hamiltonian (minimising the energy). This has many uses in data mining, image segmentation, clustering, image capturing, networking etc.

2 DYNAMICS OF A SYSTEM

The evolution dynamics for a given system is governed by it's Hamiltonian and the Schrodinger's Equation [9] . Consider a system whose Hamiltonian at time t is $H(t)$ and it's initial state $|\psi_0\rangle$. For time t

$$|\psi_t\rangle = U(t) |\psi_0\rangle \tag{1}$$

Where we get $U(t)$ from Schrodinger's Equation:

$$\frac{d}{dt} U(t) = -\iota H(t) U(t) \tag{2}$$

Solving the Equation 2, we obtain

$$U(\tau) = \exp(-\iota \int_0^{\tau} dt H(t)) \tag{3}$$

Lemma 1 (Time-ordered evolution in the interaction picture)**.** *Let $\mathscr{H}(\tau) = \mathscr{A}(\tau) + \mathscr{B}(\tau)$ be an operator-valued function defined for $\tau \in \mathbb{R}$ with continuous summands $\mathscr{A}(\tau)$ and $\mathscr{B}(\tau)$. Then*

$$\exp_{\mathcal{T}} \left(\int_0^t d\tau \mathscr{H}(\tau) \right) = \exp_{\mathcal{T}} \left(\int_0^t d\tau \mathscr{A}(\tau) \right) \cdot \exp_{\mathcal{T}} \left(\int_0^t d\tau_1 \exp_{\mathcal{T}}^{-1} \left(\int_0^{\tau_1} d\tau_2 \mathscr{A}(\tau_2) \right) \mathscr{B}(\tau_1) \exp_{\mathcal{T}} \left(\int_0^{\tau_1} d\tau_2 \mathscr{A}(\tau_2) \right) \right)$$

This can describe how a time-dependent Hamiltonian is exponentiated if it is expressed as a sum of two Hamiltonians.

From this point on-wards, unless explicit we assume that the Hamiltonian for a given system is time Independent. The state at time t is given by:

$$|\psi(t)\rangle = e^{-\iota t H} |\psi_0\rangle \tag{4}$$

2.1 Simulation Problem

To simulate the above system, one of the approaches is to construct an operator of evolution U on a set of qubits that is equivalent to applying the above evolution on the given system. If the required precision of approximation is ϵ, then U must satisfy:

$$\|U - e^{-\iota t H}\| < \epsilon \tag{5}$$

Construction of U efficiently on any computing model would solve the simulation problem. As of now, there are a few notable methods for constructing U. Qubitization, Quantum Walks, etc. are the current algorithms that are being explored.

3 GENERAL METHODS OF SIMULATION

3.1 Taylor Expansion

We can write e^{-iHt} as it's Taylor expansion:

$$e^{-iHt} = \sum_{n=0}^{\infty} \frac{(-iHt)^n}{n!} = I - iHt - \frac{H^2t^2}{2} + \frac{iH^3t^3}{6}... \tag{6}$$

For short periods we can truncate the latter terms.

An efficient quantum algorithm to simulate the Hamiltonian by truncating the latter terms is described in [6].

3.2 Quantum Walks

Szegedy quantum walks allow us to define a unitary quantum walk on a directed graph. A single step in the Szegedy quantum walk is defined by the unitary:

$$U_{walk} = S(2\Pi - I) \tag{7}$$

where $S = \sum |j,k\rangle \langle k,j|$, $\Pi = \sum_i |\psi_j\rangle \langle \psi_j|$ and $|\psi_j\rangle = \sum_k \sqrt{(P_{jk})}\, |j,k\rangle$.

P_{jk} represents the probability of making a transition to j from k and thus the matrix P will be a stochastic matrix which can be modified to represent the hamiltonian for any given system. Once, this unitary is prepared, applying it is approximately equal to the required Hamiltonian evolution [3].

The above methods are for a general Hamiltonian. The remaining study will focus on a specific class of Hamiltonians and use Trotterization for efficient simulation.

4 TROTTERIZATION

The Trotterization method focuses on constructing a circuit with efficient gate complexity for the target system whose Hamiltonian can be decomposed of the following form:

$$H = \sum_{i=1}^{\tau} H_i \tag{8}$$

where H_i is a hermitian that can be exponentiated with $O(1)$ cost in terms of gate complexity. Note that $O(1)$ gate complexity implies that the number of gates must not depend on the number of qubits.

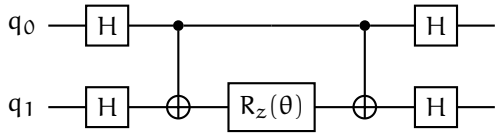
4.1 k-local Hamiltonian

A k-local Hamiltonian is a Hamiltonian that operates on only k out of the n qubits in a system, suggesting that for a given k-local Hamiltonian, the gate complexity is independent of n, satisfying the required criterion.

Consider a simple example: $H_{12} = \sigma_x \otimes \sigma_x \otimes I_{2^3}$ which is a 2-local Hamiltonian in a 5-qubit system. Simulation of H_{12} would require the construction of unitary operator $e^{-\iota t H_{12}} = e^{-\iota t \sigma_x \otimes \sigma_x}$. Intuitively it can be seen that H_{12} is a rotation applied to the first two qubits on the basis $\{|00\rangle, |11\rangle\}$. It takes the exact form:

$$e^{-\iota t \sigma_x \otimes \sigma_x} = R_{XX}(\theta) = \begin{pmatrix} \cos(\frac{\theta}{2}) & 0 & 0 & -i\sin(\frac{\theta}{2}) \\ 0 & \cos(\frac{\theta}{2}) & -i\sin(\frac{\theta}{2}) & 0 \\ 0 & -i\sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) & 0 \\ -i\sin(\frac{\theta}{2}) & 0 & 0 & \cos(\frac{\theta}{2}) \end{pmatrix}$$

This can be easily implemented as shown:



Circuit for $R_{XX}(\theta)$

4.2 Decomposition of a given Hamiltonian

Decomposition of a given Hamiltonian into sum of simpler Hamiltonians can be done in multiple ways. One simple approach that can be used is to decompose it into the Extended Pauli Basis. An n-qubit Hamiltonian H can be decomposed as follows:

$$H = \sum_{i_1, i_2, \ldots, i_n} \frac{1}{2^n} h_{i_1, i_2, \ldots, i_n} \sigma_{i_1} \otimes \sigma_{i_2} \otimes \ldots \otimes \sigma_{i_n} \tag{9}$$

where each $i_k \in \{0, 1, 2, 3\}$ and $h_{i_1, i_2, \ldots, i_n} = \frac{1}{2^n} Tr((\sigma_{i_1} \otimes \sigma_{i_2} \otimes \ldots \otimes \sigma_{i_n})H)$

This sum will have 4^n terms. But the advantage this basis provides is that they can be exponentiated in $O(1)$ gate complexity assuming that they are k-local Hamiltonians.

5 TROTTERIZATION ORDER

Trotterization is an approximate simulation of a Hamiltonian using circuits. There exist different orders of approximation, with greater order we have a lesser error rate but the gate complexity grows. The p^{th} order of Trotterization is represented using L_p. First we study a simple case of first order Trotterization.

5.1 First Order Trotterization

Let's consider a simple system $H = A + B$ such that A, B can be implemented in $O(1)$ gate complexity. The required unitary evolution as shown in 10 [5].

$$U(t) = e^{-\iota t(A+B)} \tag{10}$$

The first order approximation of the above unitary would be 11

$$L_1(t) = e^{-\iota tA} e^{-\iota tB} \tag{11}$$

Now we can compare 11 & 10 and understand when the operators will not match. $U(t)$ can be understood as evolution of the Hamiltonian H for time t. Where as $L_1(t)$ is evolution of of the system based on B for time t followed by another evolution based on A for time t.

Exponentiation of a matrix is equivalent to exponentiation of the eigenvalues in it's spectral decomposition. From this it can be seen that $e^{A+B} = e^A e^B$ only in the case where A, B are simultaneously diagnolizable. For A, B to be simultaneously diagnolizable only when they commute, i.e. $[A, B] = 0$ [9].

5.1.1 *Commuting Case*

For the case where $[A, B] = 0$, $U(t) = L_1(t)$. The approximation made is accurate with no error. This case is a trivial case which can be directly obtained from the decomposition and does not require Trotterization.

5.2 General Case

In a general scenario, A, B need not commute. In this scenario, the expression can be expanded using Baker-Campbell-Hausdorff formula [5]

$$e^{-\iota tA} e^{-\iota tB} = e^{-\iota t(A+B) - \frac{t^2}{2}[B,A] + \iota\frac{t^3}{12}[B,[B,A]] + \ldots} \tag{12}$$

The study shows that the error can be bounded as shown in 13.

$$\|L_1(t) - U(t)\| \leqslant \frac{t^2}{2}\|[B, A]\| \tag{13}$$

The above error bound can be extended to a Hamiltonian with more sum terms. The paper constructs an error bound for p^{th} order which we will look into.

The following lemma describes the trotter error with 1-norm scaling.

Lemma 2 (Trotter error with 1-norm scaling). *Let* $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ *be an operator consisting of* Γ *summands and* $t \geqslant 0$. *Let* $\Psi(t) = \prod_{\nu=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} e^{t a_{(\nu,\gamma)}} H_{\pi_\nu(\gamma)}$ *be a pth-order product formula. Then,*

$$\left\| \mathscr{S}(t) - e^{tH} \right\| = \mathcal{O}\left(\left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| \, t \right)^{p+1} e^{t\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|} \right).$$

Furthermore, H_γ *are anti-Hermitian,*

$$\left\| \mathscr{S}(t) - e^{tH} \right\| = \mathcal{O}\left(\left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| \, t \right)^{p+1} \right).$$

5.3 General Trotterization

For a n qubit Hamiltonian with τ terms in it's decomposition, the $2l^{th}$ order Trotterization is described recursively in 14

$$\begin{aligned} L_2(t) &= e^{\frac{t}{2}H_1} \cdots e^{\frac{t}{2}H_\tau} \cdots e^{\frac{t}{2}H_1} \\ L_{2l}(t) &= L_{2l-2}(u_l t)^2 \; L_{2l-2}((1-4u_k)t) \; L_{2l-2}(u_l t)^2 \end{aligned} \tag{14}$$

Here $u_l = 1/(4 - 4^{1/(2l-1)})$.

6 TROTTER ERROR

Childs present three forms of error for Trotterization. In terms of simulation, they are all said to be equivalent. We will continue using the above form of error which has been extended to any order using Commutators.

$$Err(t) = \|L(t) - e^{itH}\| \tag{15}$$

6.1 α Commutator

The error for p^{th} order is given using the α commutator as show in 16.

$$Err(t) = O(\tilde{\alpha}_{comm} t^{p+1}) \tag{16}$$

Where α_{comm} is given by the Nested commutator of all the permutations of Hamiltonians in the decomposed sum.

$$\tilde{\alpha}_{comm} = \sum_{\gamma_1,\ldots\gamma_p+1=1}^{\tau} \|[H_{\gamma_{p+1}}, \cdots [H_{\gamma_2}, H_{\gamma_1}]]\| \tag{17}$$

Lemma 3 (Trotter error with commutator scaling)**.** *Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be an operator consisting of Γ summands and let $t \geqslant 0$. Let $\mathscr{S}(t)$ be a pth-order Υ-stage product formula. Define $\widetilde{\alpha}_{\text{comm}} = \sum_{\gamma_1,\gamma_2,\ldots,\gamma_{p+1}=1}^{\Gamma} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1}\right]\right] \right\|$, where $\|\cdot\|$ is the spectral norm. Then the additive error $\mathscr{A}(t)$ and the multiplicative error $\mathscr{M}(t)$, defined respectively by $\mathscr{S}(t) = e^{tH} + \mathscr{A}(t)$ and $\mathscr{S}(t) = e^{tH}(I + \mathscr{M}(t))$, can be asymptotically bounded as*

$$\|\mathscr{A}(t)\| = \mathcal{O}\left(\widetilde{\alpha}_{comm}\, t^{p+1} e^{2t\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|}\right), \quad \|\mathscr{M}(t)\| = \mathcal{O}\left(\widetilde{\alpha}_{\text{comm}} t^{p+1} e^{2t\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|}\right).$$

6.2 Constant Accuracy

For the given system $H = \sum_{i=0}^{\tau} H_i$, the gate complexity required to achieve constant accuracy is mentioned in 18

$$O(\tau \tilde{\alpha}_{comm} t^{p+1} t^{1+1/p}) \tag{18}$$

6.3 Error for k-local Hamiltonians

The following lemma describes the error and gate complexity for simulating a k-local Hamiltonian.

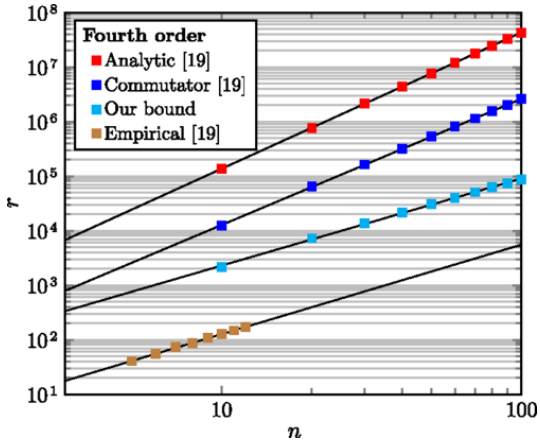


Figure 1: Gate Complexity for Molecule Simulation

Lemma 4 (Product-formula simulation of k-local Hamiltonians)**.** *Let H be a k-local Hamiltonian on n qubits (62). Let $\mathcal{S}(t)$ be a pth-order product formula. Then, the Trotter error has the scaling*

$$\left\| \mathcal{S}(t) - e^{-itH} \right\| = \mathcal{O}\left(\|H\|_1^p \|H\|_1 \right)$$

To simulate with accuracy ϵ, it thus suffices to choose a Trotter number of

$$r = \mathcal{O}\left(\frac{\|H\|_1 \|H\|_1^{1/p} t^{1+1/p}}{\epsilon^{1/p}} \right)$$

Choosing p sufficiently large, letting ϵ be constant, and implementing each Trotter step using $\mathcal{O}\left(n^k\right)$ gates, we have the gate complexity

$$n^k \|H\|_1 \|H\|_1^{o(1)} t^{1+o(1)}$$

6.4 Bound Tightness

A good approximation algorithm is to be accompanied by a tight error and complexity bound on it's performance when compared to the required answer. Post-Trotter techniques for simulation have a relatively tight bound in comparision to that of the Trotterization. For a simple Heisenberg model, the required value r for getting a constant accuracy has been plotted in 1

The paper derives error bounds for the Lie-Trotter formula and the second-order Suzuki formula. These are given as propositions below.

Proposition 1 (Tight error bound for the first-order Lie-Trotter formula)**.** *Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be a Hamiltonian consisting of Γ summands and $t \geqslant 0$. Let $\mathcal{S}_1(t) = \prod_{\gamma=1}^{\Gamma} e^{-itH_\gamma}$ be the first-order Lie-Trotter formula. Then, the additive Trotter error can be bounded as*

$$\left\| \mathcal{S}_1(t) - e^{-itH} \right\| \leqslant \frac{t^2}{2} \sum_{\gamma_1=1}^{\Gamma} \left\| \left[\sum_{\gamma_2=\gamma_1+1}^{\Gamma} H_{\gamma_2}, H_{\gamma_1} \right] \right\|$$

Proposition 2 (Tight error bound for the second-order Suzuki formula)**.** *Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be a Hamiltonian consisting of Γ summands and $t \geqslant 0$.*

Let $\mathscr{S}_2(t) = \prod_{\gamma=\Gamma}^{1} e^{-i\frac{t}{2}H_\gamma} \prod_{\gamma=1}^{\Gamma} e^{-i\frac{t}{2}H_\gamma}$ *be the second-order Suzuki formula. Then, the additive Trotter error can be bounded as*

$$\left\|\mathscr{S}_2(t) - e^{-itH}\right\| \leqslant \frac{t^3}{12} \sum_{\gamma_1=1}^{\Gamma} \left\| \left[\sum_{\gamma_3=\gamma_1+1}^{\Gamma} H_{\gamma_3}, \left[\sum_{\gamma_2=\gamma_1+1}^{\Gamma} H_{\gamma_2}, H_{\gamma_1} \right] \right] \right\| + \frac{t^3}{24} \sum_{\gamma_1=1}^{\Gamma} \left\| \left[H_{\gamma_1}, \left[H_{\gamma_1}, \sum_{\gamma_2=\gamma_1+1}^{\Gamma} H_{\gamma_2} \right] \right] \right\|$$

6.5 Dependency

The α_{comm} value is dependent on the commutativity amongst the Hamiltonians in the decomposition. Due to the sum format that is used to calculate α_{comm}, it can also be seen that there is an implicit dependency of error rate on number of decomposition terms, τ. This indicates that with growing number of terms in the decomposition, the error rate will implicitly increase. The gate complexity has a direct dependency on τ, which means overall gate complexity would be $O(\tau^2)$.

This dependency is negligible as long as the Hamiltonian is sparse which makes Suzuki-Trotterization a great Hamiltonian Simulation Algorithms. But for dense Hamiltonians, we will have to look into other approaches.

We will now continue the study of Trotterization and look at a different approach from the traditional Suzuki-Trotter approach to eliminate the dependency on τ.

7 RANDOMIZED TROTTERIZATION

Apart from Suzuki-Trotterization, Campbell [2] provides a different approach through randomization to overcome the τ dependency. The Hamiltonians that are relevant in quantum chemistry have a dependency between the number of qubits and τ. It is often the case that $\tau = O(n^4)$. This create a $O(n^8)$ scaling factor which is unacceptably high.

The study gives a random algorithm qDRIFT, to construct the circuit. We understand the Hamiltonian decomposition and it's new representation to comprehend the construction.

$$H = \sum_i h_i \hat{H}_i \tag{19}$$

7.1 Normalizing Decomposition Terms

The original decomposition 8 puts no constraint on the term H_i other than the requirement that it must be hermitian and must be exponentiated in $O(1)$ gate complexity.

We construct $\hat{H}_i$ such that $H_i = h_i \hat{H}_i$ where $|\hat{H}_i| = 1$ and $h_i = |H_i|$. Each term of Hamiltonian decomposition is replaced with a coefficient and a normalized Hamiltonian without changing the overall Hamiltonian. The coefficient h_i is referred to as the strength of the term $\hat{H}_i$

7.2 Picking Total Gates

Similar to Trotter number r in Suzuki-Trotter, we pick the number of gates in qDRIFT which is denoted by N. This number will also be used to determine the new strength of the individual terms in the Hamiltonian, rather than using h_i, we will set the strength of all the Hamiltonians to be $\Gamma = t\lambda/N$ where λ is an upper bound on the largest singular value of H given by 20. So for simulating each term we will have to apply the operation $e^{\iota\Gamma H_j}$.

$$\lambda = \sum_i h_i \tag{20}$$

7.3 Picking Terms

As constructed above, we have set the strength of all the Hamiltonian terms to be Γ. Intuitively it can be seen that in our initial Hamiltonian 19, the strength of Hamiltonian determines the effect of the term on the overall simulation, a term with larger strength will have a larger impact on the evolution of the system governed by H. If the terms are being picked at random then the strength must play a role of deciding the odds of each term.

To weigh in the strength, the probability distribution of the terms are the normalized values of the strength. The probability p_i of picking H_i is given by 21

$$p_i = h_i/\lambda \tag{21}$$

7.4 Algorithm

The final algorithm for constructing the circuit with fixed number of gates N is given by algorithm 1

Algorithm 1 qDRIFT ALgorithm

$i \leftarrow 0$
$\lambda \leftarrow \sum_j h_j$
$V_{list} \leftarrow \{\}$
while $i < N$ **do**
  $i \leftarrow i + 1$
  $j \leftarrow$ SAMPLE()
  $V_{list} \leftarrow V_{list}.append(e^{\iota\lambda H_j/N})$
end while

The operators in V_{list} are applied in the order they are appended, the study then shows that for constant accuracy, the N is chosen based on λ and t.

$$N = \left\lceil \frac{2\lambda^2 t^2}{\epsilon} \right\rceil \tag{22}$$

The study performs analysis for propane and other such molecules and presents the improvement of gate count shown by randomization over deterministic Suzuki-Trotterization in 2.

The above section concludes introductory theory to Trotterization and it's different kinds of variants. One of the open problems in Hamiltonian Simulation is de-

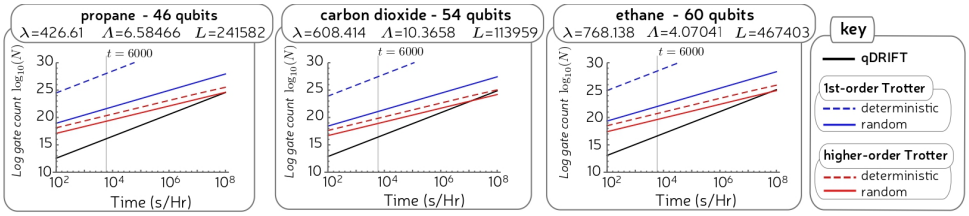


Figure 2: Gate Complexity for Molecule Simulation

ciding the most effective simulation method. The following sections cover systems that can be simulated using Trotterization along with their physical significance.

8 SYSTEMS

Now that Trotterization has been established for a group of Hamiltonians and its error has been understood, we can look at a few example systems that can be simulated well and further understand what Trotterization can offer.

8.1 Nearest-Neighbour Lattice Hamiltonian

A lot of physical systems can be modeled in the nearest neighbor lattice form. This is natural since most of the interactions in a system will be with the closest particle. Systems of spins such as the Ising model, Heisenberg models, etc are modeled with lattice Hamiltonians [1]. From the perspective of Trotterization, we focus on the systems that have interactions only among their neighboring particles. The Hamiltonians can generally be stated as:

$$H = \sum_{i=1}^{n-1} H_{j,j+1}$$

Here, $H_{j,j+1}$ is a hamiltonian that only acts on j and $j+1$ qubit. The paper showed that the trotter error would be of the order $O(nt^2)$ assuming that the order of Trotterization is constant. We can arrive at this result by calculating $\tilde{\alpha}_{comm}$.

$$\tilde{\alpha}_{comm} = \sum_{\gamma_1,\ldots\gamma_p+1=1}^{n} \|[H_{\gamma_{p+1}},\cdots[H_{\gamma_2},H_{\gamma_1}]]\|$$

It can noticed that for H_{γ_1}, there are only constant number of H_{γ_2} with which it does not commute. Continuing this reasoning, we can see for constant order p, $\tilde{\alpha}_{comm} = O(n)$.

8.2 Ising and Heisenberg Models

In these models, spins, which can be interpreted as magnetic fields of individual atoms, are arranged in a graph, usually a lattice format, which allows a spin to interact with its nearest neighbours. We can describe the interactions between these neighbours using a Hamiltonian. Since only 2 neighbours interact at a time with each other, this will be a 2-local Hamiltonian which will describe (a) How strongly they interact - decided by the magnitude of the coefficient of the interaction terms and (b) Are aligned spins or anti-aligned spins preferred - decided by the sign of

the coefficient (negative means aligned spins are preferred as it will decrease the energy). The Hamiltonian can also describe the effect of an external magnetic field.

The following models [7] are some 2-local Hamiltonians, which are commonly studied models in condensed matter physics, usually to analyse magnetic properties of matter.

8.2.1 Quantum Ising Model

This model describes the nearest-neighbour interactions determined by spin projections along the z-axis in a lattice. It also describes the effect of an external magnetic along the x-axis.

Its hamiltonian is given by

$$H = -J \sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z - g \sum_i \sigma_i^x$$

8.2.2 Quantum Heisenberg Model

Similar to the quantum Ising model, this model also describes nearest-neighbour interactions in a lattice. However, now the interactions are determined by spin projections in the x, y and z directions. The external magnetic field is described along the z-axis (this axis doesn't matter as all three directions are symmetric in this model).

Its hamiltonian is given by

$$H = -\sum_{\langle i,j \rangle} (J_x \sigma_i^x \sigma_j^x + J_y \sigma_i^y \sigma_j^y + J_z \sigma_i^z \sigma_j^z) + h \sum_i \sigma_i^z$$

8.2.3 Anti-ferromagnetic Heisenberg model

This is just a special case of the Heisenberg model, to get an idea of the model's physical significance. It prefers all spins to be anti-aligned which physically represents anti-ferromagnetism.

$$H = \sum_{\langle i,j \rangle} \sigma_i^x \sigma_j^x + \sigma_i^y \sigma_j^y + \sigma_i^z \sigma_j^z$$

A ferromagnetic Heisenberg model, which prefers all spins to be aligned, would be represented by $-H$.

9 IMPLEMENTATION

Using IBM's Qiskit, we implemented a module that can convert a Hamiltonian or it's decomposition into a circuit for it's simulation using different variants of Trotterization. The source code can be found here.

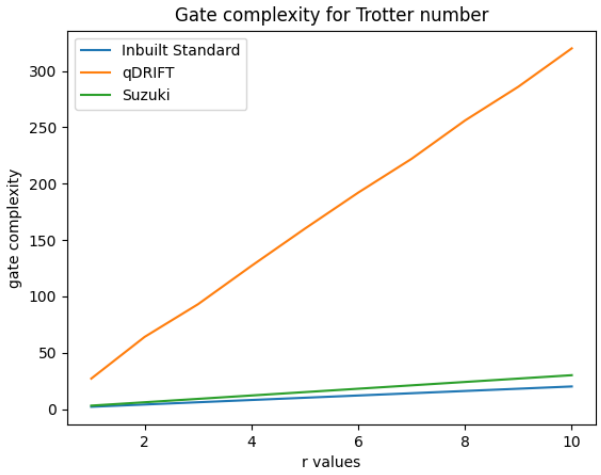


Figure 3: Plot for Gate Complexity vs Trotter number for different variants of Trotterization

9.1 Decomposition

If the given input is a Hamiltonian of size $2^n \times 2^n$, we decompose it into the Extended Pauli Basis to get the form shown in 23

$$H = \sum_j \otimes_k \sigma_a^i \tag{23}$$

Here, $\sigma_a \in \{I, \sigma_x, \sigma_y, \sigma_z\}$ and σ_a^i acts on i^{th} qubit. The Hamiltonian is assumed to be k-local for given value k.

9.2 Trotterization

The above study has shown different variants of Trotterization, we apply and give out the circuit for each of the variants. For example given trotter number r, we construct the Lie Trotter Expression as shown in 24.

$$e^{-itH} \approx (e^{-i(t/r)H_1} e^{-i(t/r)H_1} ... e^{-i(t/r)H_L})^r \tag{24}$$

9.3 Circuit Construction & Benchmarking

We can easily construct 2-qubit gates like R_{xx} as we saw in 4.1. This can be generalised to k-qubit gates corresponding to $e^{-it(\otimes\sigma_i^a)}$.

Gate Complexity

For overall analysis of Trotterization Algorithms, we have plotted in 3 the gate complexity trend that is observed for a range of trotter number r for different variants of Trotterization.

9.4 Result

With this circuit we can take an initial state $|\psi_0\rangle$ that evolves according to the hamiltonian H and obtain:

$$|\psi_t\rangle \approx e^{-itH} |\psi_0\rangle$$

10 FUTURE WORK

Exploring Trotterization and Hamiltonian Simulation helped us discover a lot of the problems that are unsolved in the field of Simulation. In this section, we shed light on a few related problems.

10.1 Best Simulation Method

In the post-Trotter era, there have been plenty of simulation techniques that give advantage in one form or the other based on the system. It is still unclear on what the best Hamiltonian Simulation algorithm is.

Even if we assume that there is no best algorithm for a general case, determining what the best technique would be for a given Hamiltonian is non-trivial[4].

10.2 Ideal Decomposition

Although we have provided a decomposition for a given Hamiltonian in section 4.2, there are plenty of other gates in our basis, which means that our Hamiltonian can be decomposed as sum of Hermitians that can be exponentiated in $O(1)$ complexity in multiple ways. Finding an ideal decomposition is still an open problem.

10.3 Analytical Bounds

For a simple system and first order, Baker-Campbell-Hausdorff formula presented analytical tight bound in section 5.2. Although it has been achieved for second order Trotterization, there is no bound of such form for general Trotterization.

10.4 Loose Bounds

Trotterization suffers from being neglected in the pool of Hamiltonian Simulation algorithms due to the high complexity the current bounds display. But as shown in 1, there is a severe gap between the experimental complexity and the complexity provided even by the newest α_{comm} bound.

10.5 Space for Randomization

qDRIFT is a Randomized Trotterization method that outperforms Suzuki-Trotter for Hamiltonians that have a large decomposition. Building up on qDRIFT using

randomization for the construction could lead to better simulation techniques for other subclasses of Hamiltonians.

In general, tweaking of current Hamiltonian simulation techniques for creating better algorithms for relevant Hamiltonians is a field that could be explored. Similar to how qDRIFT works well for quantum chemistry, simulation of other disciplines such as Condensed matter physics can be implemented with better performance.

10.6 Scope for topologial optimisation

On a quantum computer, we are often limited by the topology of the system. For example, it might be the case that only neighbouring qubits interact. Hence, for interactions between qubits that are not physically connected, SWAP gates would be needed.

There is scope for optimisation here in terms of number of SWAP gates. Though, even after minimizing the number of SWAP gates, there will still be some of them due to the interactions given the Hamiltonian. However, there is more scope for minimizing the SWAP gates by ignoring some interactions from the Hamiltonian. As a result of this, there will be a trade-off between accuracy and speed. The weakest interactions in a Hamiltonian can be ignored if they reduce the amount of SWAP gates considerably.

Deciding which of the interactions to ignore based on the topology of a quantum computer is a problem worth considering.

11 SIMULTONIAN – OUR FUTURE WORK

Based on the concept and problems we learnt from this project, we have decided to make our own open-source framework called 'Simultonian'.

The idea is to create a framework built on qiskit to simulate Hamiltonians for given quantum systems efficiently and benchmark different strategies for simulation.

11.1 What will it consist of?

Some of the main features and components of Simultonian that we are planning to make are:

Hamilutor

A library primarily focused on implementing a collection of Hamiltonian Simulation techniques.

Optimisation Heuristics

The Hamilutor library will be accompanied by some implemented heuristics that pick the optimal simulation method with the aim of reducing the cost of simulation

which includes the gate complexity, ancilla qubits, and the error given a Hamiltonian.

MenchBarker

A library to perform benchmarking on the existing Hamiltonian Simulation techniques.

12 APPENDIX

In this section, we present the detailed analysis of a system where Trotterization can is applied.

12.1 Trotterized adiabatic simulation for matrix diagonalisation

The paper [10] proposes a method of reaching the ground state of a boson Hamiltonian with bilinear interactions. Finding the ground state of a Hamiltonian is QMA-hard which in special cases can be reduced to diagonalisation problem in certain Hilbert spaces. The matrix diagonalisation problem is NP-hard through classical computation.

The protocol for Trotterized-AQC (TAQC) is described as follows:

1. Prepare the ground state $|\psi_0\rangle$ of Hamiltonian H_0.
2. Find the problem Hamiltonian H_p whose ground state encodes the solution
3. Set the total Hamiltonian as $H(t) = f(t)H_0 + g(t)H_p$ where $f(t)$ and $g(t)$ are slow-varying control function(eg: $f(t) = 1 - \frac{t}{T}$ and $g(t) = \frac{t}{T}$, where T is the time for the entire evolution)

The evolution operator is decomposed into a sequence of steps using the Trotter-Suzuki formula, which is the key ingredient and given by:

$$U(T) := exp[-i \int_0^T H(t)\ dt] \approx \prod_{a=0}^{k-1} exp[-iH(a)]$$

An optical implementation of a Trotterized adiabatic simulation (a type of TAQC) using linear optical elements is proposed by the paper. The linear optic system in consideration involves the encoding of the logical qubit into the spatial modes of the photon, which is preserved for a relatively long time and is controllable. The operations of the system are static so that the dynamics are discretized.

12.2 The Model

The generalised model considered for this case is:

$$H_0 = \sum_s \epsilon_s b_s^\dagger b_s, \;\; H_p = \sum_l \epsilon_l b_l^\dagger b_l + \sum_{m \neq n} J_{mn} b_m^\dagger b_n$$

where, $b_i^\dagger$ and b_i are the creation and annihilation operators respectively (ladder operators) for the ith bosonic mode (spatial mode of the photon), J_{mn} is the coupling coefficient between the mth and nth mode.

The creation and annihilation operators have commutators such that

$$[b_i, b_j^\dagger] = \delta_{ij}$$

and

$$[b_i^\dagger, b_j^\dagger] = [b_i, b_j] = 0$$

For a one-photon subspace, the Hamiltonian system given above can represent a matrix which has no additional constraints other than being Hermitian. So the process of finding its ground state is equivalent to diagonalizing a general Hermitian matrix.

As mentioned before, the bosonic modes are mapped to the spatial modes of photons. Hence, $b_i^\dagger$ corresponds to a photon propagating along an optical path labeled by i, and bi corresponds to the absence of the photon from the path.

12.3 Formulation

Decomposing the Hamiltonian, we get:

$$U(t) = \prod_{a=0}^{k-1} e^{-i\left((1-\frac{a}{k})\tau \sum_s \epsilon_s b_s^\dagger b_s + (\frac{a}{k})\tau\left(\sum_l \epsilon_l b_l^\dagger b_l + \sum_{m\neq n} J_{mn} b_m^\dagger b_n\right)\right)}$$

By the Hermiticity of J, $J_{mn} = J_{nm}^*$, Hence:

$$\begin{aligned}\sum_{m\neq n} J_{mn} b_m^\dagger b_n &= \sum_{m<n} (J_{mn} b_m^\dagger b_n + J_{mn}^* b_m b_n^\dagger) \\ &= \sum_{m<n} (Re(J_{mn})(b_m^\dagger b_n + b_m b_n^\dagger) + Im(J_{mn})(b_m^\dagger b_n - b_m b_n^\dagger))\end{aligned}$$

Thus, by Trotter-Suzuki formulation, we can further decompose the exponential operators as:

$$\begin{aligned}e^{-i(1-\frac{a}{k})\tau \sum_s \epsilon_s b_s^\dagger b_s} &= \prod_s e^{-i(1-\frac{a}{k})\tau \epsilon_s b_s^\dagger b_s} \\ e^{-i(\frac{a}{k})\tau \sum_l \epsilon_l b_l^\dagger b_l} &= \prod_l e^{-i(\frac{a}{k})\tau \epsilon_l b_l^\dagger b_l} \\ e^{-i(\frac{a}{k})\tau \sum_{m\neq n} J_{mn} b_m^\dagger b_n} &\approx \prod_{m<n} e^{(\frac{a}{k})\tau Im(J_{mn})(b_m^\dagger b_n - b_m b_n^\dagger)} e^{-i(\frac{a}{k})\tau Re(J_{mn})(b_m^\dagger b_n + b_m b_n^\dagger)}\end{aligned}$$

Implementation

Two types of linear gate elements are used in the implementation:

1. **Phase shifters (PSs)** represented by the unitary:

$$U_{ps}(\phi) = e^{-i\phi c^\dagger c}$$

2. **Beam splitters (BSs)** represented by the unitary:

$$U_{bs}(\theta) = e^{\theta(c^\dagger d - c d^\dagger)}$$

The are diagrammatically represented below:

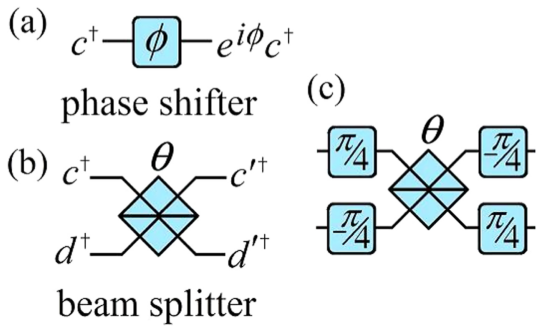


Figure 4: Optical elements and their combinations used for simulation. (a) Phase shifter (b) beam splitter and (c) combination for the simulation of the real part of the interaction.Output modes $c\prime^\dagger$ and $d\prime^\dagger$ in (b) are defined by $c\prime^\dagger = c^\dagger \cos\theta + d^\dagger \sin\theta$, $d\prime^\dagger = -c^\dagger \sin\theta + d^\dagger \cos\theta$

Each of the factors in the mentioned in the previous section can be implemented as follows:

1. $e^{-i(1-\frac{a}{k})\tau\sum_s \epsilon_s b_s^\dagger b_s}$ and $e^{-i(\frac{a}{k})\tau\sum_l \epsilon_l b_l^\dagger b_l}$ can be implemented by the phase shifters $U_{ps}^s((1-\frac{a}{k})\tau\epsilon_s)$ and $U_{ps}^l((\frac{a}{k})\tau\epsilon_l)$ where s and l are optical modes.

2. $e^{(\frac{a}{k})\tau Im(J_{mn})(b_m^\dagger b_n - b_m b_n^\dagger)}$ can be implemented by the beam splitter $U_{bs}^{mn}((\frac{a}{k})\tau Im(J_{mn}))$

3. $e^{-i(\frac{a}{k})\tau Re(J_{mn})(b_m^\dagger b_n + b_m b_n^\dagger)}$ can be implemented by the combination of four PSs and one BS $U_{bs}^m(\frac{-\pi}{4})U_{bs}^n(\frac{\pi}{4})U_{bs}^{mn}((\frac{a}{k})\tau Re(J_{mn}))U_{bs}^m(\frac{\pi}{4})U_{bs}^n((\frac{-\pi}{4})$

Thus we get the circuit for calculating nearest-neighbour case as:

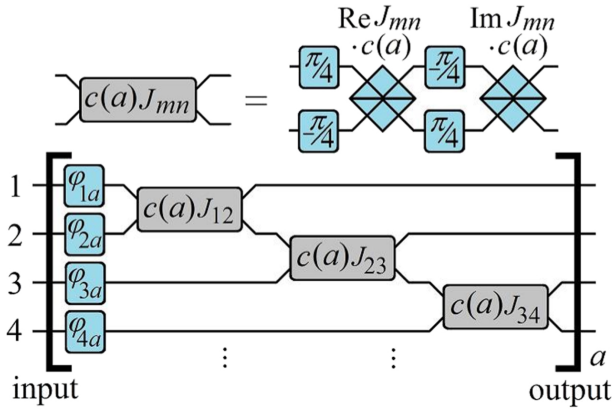


Figure 5: Sketch of the whole simulation of the adiabatic evolution (for only nearest-neighbor interactions). The parameters of each element are shown in the figure. Square brackets labeled by a mark the unit cell which periodically repeats along the propagation direction of the photons (from input to output) with $a = 0, \cdots, k-1$. Function $c(a) = \tau a/k$. The phase function of the PS $\varphi_{na} = (1 - a/k)\tau_a \epsilon_n + (a/k)\tau_a \epsilon_n$

.

This can be expanded to all bosonic interactions as:

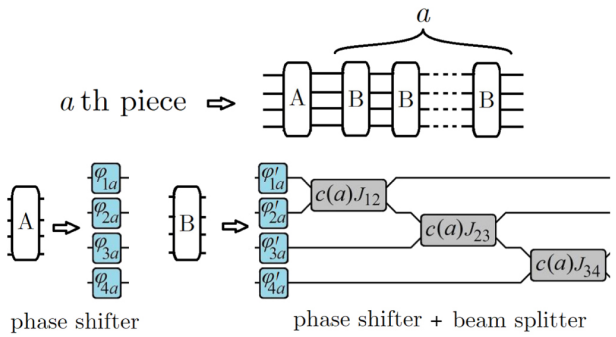


Figure 6: A simplification of the Trotterization circuit. The upper panel shows the a^{th} piece represented by block A and B. The lower panel shows the detailed schemes in block A and B

REFERENCES

[1] Yuan Su Andrew M. Childs. Nearly optimal lattice simulation by product formulas, 2019.

[2] Earl Campbell. Random compiler for fast hamiltonian simulation. *Physical Review Letters*, 123(7), aug 2019.

[3] Andrew M. Childs. On the relationship between continuous- and discrete-time quantum walk, 2008.

[4] Andrew M. Childs, Dmitri Maslov, Yunseong Nam, Neil J. Ross, and Yuan Su. Toward the first quantum simulation with quantum speedup. *Proceedings of the National Academy of Sciences*, 115(38):9456–9461, sep 2018.

[5] 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.

[6] Dominic W. Berry et al. Simulating hamiltonian dynamics with a truncated taylor series, 2015.

[7] Sevag Gharibian et al. Quantum hamiltonian complexity, 2015.

[8] Zsolt Tabi et al. Quantum optimization for the graph coloring problem with space-efficient embedding, 2020.

[9] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.

[10] Mark S Byrd Yifan Sun, Jun-Yi Zhang and Lian-Ao Wu. Trotterized adiabatic quantum simulation and its application to a simple all-optical system. *New Journal of Physics*, 22, May 2020.