Hw5

Thursday, April 14, 2022 10:35 PM

Problem Set 5

AS.171.402: Applied Quantum Information [Spring 2022]

Due Date: April 19, 2022 (11:59 pm)

---

1. Hamiltonian Simulation for the time-dynamics of a Spin Chain.

In this problem, we will utilize the Hamiltonian Simulation algorithm to simulate the dynamics of the *Heisenberg model*; this model explains experimental observations in a variety of magnetic materials at low temperatures. This model of magnetism is highly relevant in the field of Quantum Condensed Matter Physics, and in this problem our goal is to analyze a quantum algorithm that can be used to simulate the time-dynamics of this model.

We will simulate a toy-version of the one-dimensional spin-1/2 Heisenberg model consisting of four spins, which is described by the following Hamiltonian,

$$\hat{H} = \frac{1}{2}\sum_{\alpha\in\{x,y,z\}}\sum_{i=1}^{3} J\ \hat{\sigma}_i^\alpha \hat{\sigma}_{i+1}^\alpha + \frac{1}{2}\sum_{\alpha\in\{x,y,z\}}\sum_{i=1}^{4} B_\alpha\ \hat{\sigma}_i^\alpha \tag{1}$$

This model depends on four parameters : (J, B_x, B_y, B_z), and the spin degree of freedom on each site represented as the two basis states $\{|\uparrow\rangle, |\downarrow\rangle\}$ corresponding to pointing along $+z$ and $-z$ direction respectively. The initial state of the magnet is given by spins that are all aligned in $-z$ direction, $|\psi_0\rangle = |\downarrow\downarrow\downarrow\downarrow\rangle$. In experiments, scientists are able to measure the spatially averaged magnetization vector, $\vec{m}(t)$ as a function of time t, which is defined as the following observable,

$$\vec{m}(t) \equiv (m_x(t), m_y(t), m_z(t)), \tag{2}$$

$$\text{where,}\quad m_\alpha(t) = \langle\psi(t)|\frac{1}{4}\sum_{i=1}^{4}\hat{\sigma}_i^\alpha(t)\,|\psi(t)\rangle\ ,\ \alpha\in\{x,y,z\} \tag{3}$$

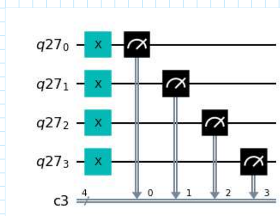
$$\text{and,}\quad |\psi(t)\rangle = \hat{U}(t)\,|\psi_0\rangle = e^{-i\hat{H}t}\,|\psi_0\rangle \tag{4}$$

where, $\hat{H}$ and $|\psi_0\rangle$ have been defined above.

In this problem, the goal is to utilize the Product-Formula based Hamiltonian Simulation Algorithm discussed in class to directly compile the time-evolution unitary for the Heisenberg model, $\hat{U(t)} = e^{-i\hat{H}t}$ by mapping the state of the spin to a qubit, $|\uparrow\rangle \to |0\rangle$, and $|\downarrow\rangle \to |1\rangle$.

The code in the accompanying jupyter notebook should be implemented in way that it can be used for arbitrary (J, B_x, B_y, B_z). For the purposes of testing, you can check on these values of the parameters: $J = 1, B_x = 1, B_y = 1, B_z = 1$. We provide a function `classically_simulate_Heisenberg_chain()` that classically simulates this model, and outputs $m_\alpha(t)$ which can be used to verify and benchmark the results of your quantum algorithm.

(a) What is the state-preparation circuit to prepare the initial state $|\psi_0\rangle$? Implement this circuit in the accompanying jupyter notebook in the function `create_initial_state()`.



-starting with |0000⟩, so applying X to all qubits gives us |1111⟩

(b) In the following, we obtain a circuit representation for the two-qubit gate for implementing the Heisenberg interaction. Let us define the Heisenberg interaction gate as,

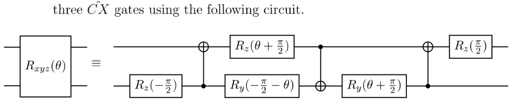
$$R_{xyz}(\theta) = e^{-i\frac{\theta}{2}\left(\hat{\sigma}_1^x\hat{\sigma}_2^x + \hat{\sigma}_1^y\hat{\sigma}_2^y + \hat{\sigma}_1^z\hat{\sigma}_2^z\right)} \tag{5}$$

(i) Using the properties of Pauli Matrices, show that the terms in the exponent in Eq. 5 commute with each other, i.e

$$[\hat{\sigma}_1^x\hat{\sigma}_2^x, \hat{\sigma}_1^y\hat{\sigma}_2^y] = [\hat{\sigma}_1^y\hat{\sigma}_2^y, \hat{\sigma}_1^z\hat{\sigma}_2^z] = [\hat{\sigma}_1^z\hat{\sigma}_2^z, \hat{\sigma}_1^x\hat{\sigma}_2^x] = 0 \tag{6}$$

(ii) What is the 4×4 matrix representation of $R_{xyz}(\theta)$ in the computational basis? (Hint: Use the result from (i) to split the exponential in Eq. 5 using the BCH formula, and then expand each of the exponentials $R_{\alpha\alpha}(\theta) = e^{-i\frac{\theta}{2}\hat{\sigma}_1^\alpha\hat{\sigma}_2^\alpha}$, $\alpha = x, y, z$ using Pauli Matrix properties.)

(iii) Show that the $R_{xyz}(\theta)$ gate can be decomposed (upto an overall phase) using three $\hat{CX}$ gates using the following circuit.



[Hint: One way to do this is to show that the decomposed circuit gives the same output state as the matrix in (ii) when applied to the basis states $|00\rangle, |01\rangle, |10\rangle, |11\rangle$.]

b)

$A\otimes B\cdot C\otimes d = AC\otimes Bd$ & given Pauli matrices $AB = -BA$

i) $[\sigma_1^x\sigma_2^x, \sigma_1^y\sigma_2^y] = \sigma_1^x\sigma_2^x\sigma_1^y\sigma_2^y - \sigma_1^y\sigma_2^y\sigma_1^x\sigma_2^x$

$= \sigma_1^x\sigma_1^y\otimes\sigma_2^x\sigma_2^y - \sigma_1^y\sigma_1^x\otimes\sigma_2^y\sigma_2^x = \sigma_1^x\sigma_1^y\otimes\sigma_2^x\sigma_2^y - (-\sigma_1^x\sigma_1^y\otimes -\sigma_2^x\sigma_2^y)$

$= \sigma_1^x\sigma_1^y\otimes\sigma_2^x\sigma_2^y - (-)(-)(\sigma_1^x\sigma_1^y\otimes\sigma_2^x\sigma_2^y)$

$= 0$

ii) $e^{-i\frac{\theta}{2}(\sigma_1^x\sigma_2^x + \sigma_1^y\sigma_2^y + \sigma_1^z\sigma_2^z)} \Longrightarrow \prod_{\alpha}^{x,y,z} e^{-\frac{\theta}{2}\sigma_1^\alpha\sigma_2^\alpha}$

using Comutation Relation that $[\sigma_1^\alpha\sigma_2^\alpha, \sigma_1^\beta\sigma_2^\beta] = 0$ and BCH theory

Since $\rightarrow R_{xyz} = R_{xx}(\theta)\cdot R_{yy}(\theta)\cdot R_{zz}(\theta)$

$$= \begin{bmatrix} \cos(\theta/2) & 0 & 0 & i\sin(\theta/2) \\ 0 & \cos(\theta/2) & -i\sin(\theta/2) & 0 \\ 0 & -i\sin(\theta/2) & \cos(\theta/2) & 0 \\ -i\sin(\theta/2) & 0 & 0 & \cos(\theta/2) \end{bmatrix} \cdot \begin{bmatrix} \cos(\theta/2) & 0 & 0 & i\sin(\theta/2) \\ 0 & \cos(\theta/2) & -i\sin(\theta/2) & 0 \\ 0 & -i\sin(\theta/2) & \cos(\theta/2) & 0 \\ -i\sin(\theta/2) & 0 & 0 & \cos(\theta/2) \end{bmatrix} \cdot \begin{bmatrix} e^{-i\theta/2} & 0 & 0 & 0 \\ 0 & e^{i\theta/2} & 0 & 0 \\ 0 & 0 & e^{i\theta/2} & 0 \\ 0 & 0 & 0 & e^{-i\theta/2} \end{bmatrix}$$

$$\doteq \begin{pmatrix} e^{-(i\,a)/2}\left(\sin^2(\frac{a}{2}) + \cos^2(\frac{a}{2})\right) & 0 & 0 & 0 \\ 0 & e^{(i\,a)/2}\left(\cos^2(\frac{a}{2}) - \sin^2(\frac{a}{2})\right) & -2\,i\,e^{(i\,a)/2}\sin(\frac{a}{2})\cos(\frac{a}{2}) & 0 \\ 0 & -2\,i\,e^{(i\,a)/2}\sin(\frac{a}{2})\cos(\frac{a}{2}) & e^{(i\,a)/2}\left(\cos^2(\frac{a}{2}) - \sin^2(\frac{a}{2})\right) & 0 \\ 0 & 0 & 0 & e^{-(i\,a)/2}\left(\sin^2(\frac{a}{2}) + \cos^2(\frac{a}{2})\right) \end{pmatrix}$$

$$= \begin{bmatrix} e^{-(i\theta/2)} & 0 & 0 & 0 \\ 0 & e^{i\theta/2}[\cos^2(\frac{\theta}{2}) - \sin^2(\frac{\theta}{2})] & -2ie^{i\theta/2}\sin(\frac{\theta}{2})\cos(\frac{\theta}{2}) & 0 \\ 0 & -2ie^{i\theta/2}\sin(\frac{\theta}{2})\cos(\frac{\theta}{2}) & e^{i\theta/2}[\cos^2(\frac{\theta}{2}) - \sin^2(\frac{\theta}{2})] & 0 \\ 0 & 0 & 0 & e^{-i\theta/2} \end{bmatrix}$$

iii) $R_{xyz}(\theta) = \doteq I\otimes R_z(-\frac{\pi}{2})\cdot CX_0^1\cdot R_z(\theta+\frac{\pi}{2})\otimes R_y(\frac{-\pi}{2}-\theta)\cdot CX_1^0\cdot I\otimes R_y(\theta+\frac{\pi}{2})\cdot CX_0^1\cdot R_x(\frac{\pi}{2})\otimes I$

Done In Qiskit using Hamiltonian gate function to show that the unitary an circuit give the same result

(iv) In the accompanying Jupyter Notebook, implement the circuit provided above in the function `Rxyz(theta)`.

(c) In this sub-problem, we will set up the circuit to time-evolve the initial state.

(i) Consider a time step, Δt. Use the first order product formula to obtain the time evolution operator for that particular step, $\hat{U}(\Delta t) = e^{-i\hat{H}\Delta t}$. For this part, represent the circuit in terms of the $R_{xyz}(\theta)$ gate and single qubit rotations.

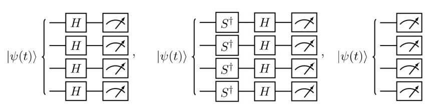
$$\not{}^{4}\,\boxed{U(\Delta t)}\,\not{} \;=\; ?, \qquad \not{}^{4}\,\boxed{U(t)}\,\not{} \;=\; ?$$

(ii) In the accompanying jupyter notebook, implement a function `first_order_PF_step()` that generates the circuit obtained in part (c)-(i) to produce the unitary to evolve the state for a single time step `delta_t`. Using this function, define a new function `first_order_PF()` that generates the circuit to evolve a state up to `m` steps, corresponding to a time $t = m\Delta t$.
[Hint: Recall that $U(t = m\Delta t) = (U(\Delta t))^m$.]

(iii) Combine the circuit generated in part (a) and (d) to generate the full circuit that evolves the state, $|\psi_0\rangle$ for time t under the time-evolution operator $U(t)$.

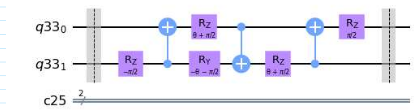
(d) The final piece to simulating the experiment, is to connect the measurement outcomes from the quantum circuit to the observables of interest in Eqs. 2-4.

(i) Consider the following circuits that implement measurement outcomes along x, y and z basis respectively. Now, given a set of measurements results (outcome



probabilities) from these circuits, $\{p_{\alpha,0000}, p_{\alpha,0001}\cdots p_{\alpha,1111}\}$ with $\alpha = x, y, z$ corresponding to measurements in the x, y and z basis respectively. What is the expression for m_x and m_y and m_z in terms of these measured probabilities?

iv)



c) $H = \frac{J}{2}[\sigma_1^x\sigma_2^x + \sigma_1^y\sigma_2^y + \sigma_1^z\sigma_2^z] + \frac{J}{2}[\sigma_2^x\sigma_3^x + \sigma_2^y\sigma_3^y + \sigma_2^z\sigma_3^z] + \frac{J}{2}[\sigma_3^x\sigma_4^x + \sigma_3^y\sigma_4^y + \sigma_3^z\sigma_4^z] + \sum_i^{x,y,z}(\sigma_1^i + \sigma_2^i + \sigma_3^i + \sigma_4^i)\cdot\frac{B^i}{2}$

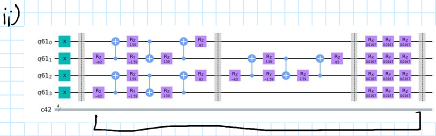
$R_{xyz_{12}}(J/2)$ $\quad R_{xyz_{23}}(J/2)$ $\quad R_{xyz_{34}}(J/2)$ $\quad \sum_i^{x,y,z} R_{x_i}(\frac{B^x}{2})R_{y_i}(\frac{B^y}{2})R_{z_i}(\frac{B^z}{2})$

$\alpha = \frac{B^x\Delta t}{2}$ $\quad \beta = \frac{B^y\Delta t}{2}$ $\quad \gamma = \frac{B^z\Delta t}{2}$

where $U(t) = e^{im\Delta t H}$
$= (e^{i\Delta t H})^m$

Repeated m times

ii)



t/Δt times

d) i) $m_\alpha = \langle\psi(t)|\frac{1}{4}\sum_{i=1}^{4}\sigma_i^\alpha(t)|\psi(t)\rangle$

$= \frac{1}{4}\sum_{i=1}^{4}\langle\psi|\sigma^\alpha|\psi\rangle$

$= \frac{1}{4}(\langle\sigma_1^\alpha\rangle + \langle\sigma_2^\alpha\rangle + \langle\sigma_3^\alpha\rangle + \langle\sigma_4^\alpha\rangle)$

$= \frac{1}{4}([2P_{1\,0} - 1] + [2P_{2\,0} - 1] + [2P_{3\,0} - 1] + [2P_{4\,0} - 1])$

$= \frac{1}{4}(2P_{1\,0} + 2P_{2\,0} + 2P_{3\,0} + 2P_{4\,0} - 4)$

$= \frac{1}{2}(P_1 + P_2 + P_3 + P_4) - 1$

$P_1 = \frac{1}{16}(P_{0000} + P_{0001} + P_{0011} \cdots P_{0111})$
$P_2 = \frac{1}{16}(P_{0000} + P_{0011} + P_{1000} \cdots P_{1011})$
$P_3 = \frac{1}{16}(P_{0000} + P_{0001} + P_{0101} \cdots P_{1101})$
$P_4 = \frac{1}{16}(P_{0000} \quad P_{1110})$

P_{0000} has 4 occurrences due to 4 zeros

so

$= \frac{1}{2}\left(\sum_{abcd}\frac{P_{abcd}}{2}((-1)^a + (-1)^b + (-1)^c + (-1)^d)\right)$ w/

(ii) Now, write a function, `measurement_circuits()` that takes as input a circuit, and generates a list of circuits that measures all the qubits in the x y and z basis respectively.

(iii) Produce a batch of circuits, that take the circuit generated from (iii) to produce a complete circuit involve preparation of the initial state, time-evolution with $U(t)$ and measurement.

(f) In the accompanying jupyter notebook, execute the quantum circuit, collect the measurement outcomes and compute the average magnetization along x, y and z directions. For this part of the problem set the number of shots to be at least 1000000, and $J = B_x = B_y = B_z = 1$.

(i) Set $\Delta t = 0.05$ and Plot the average magentizations as a function of time (show for times between $t = 0$ and $t = 2.5$).

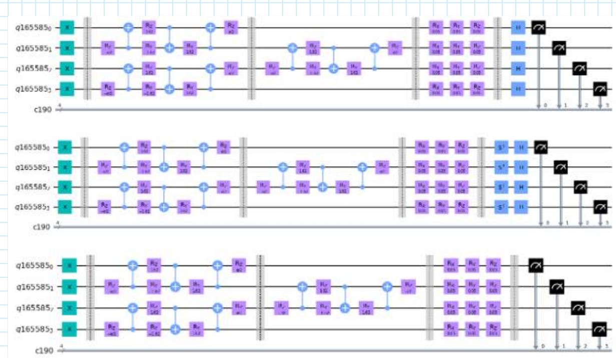
(ii) Examine the error as a function of the number of time-steps. Fix a time, $t = 1.0$, and vary the number of steps between [10,20,30,40,50,60,70,80,90,100]. Setting $\Delta t = t/m$ and plot the error in the computed magnetization as a function of m. For the exact result, use the classical simulation provided in the notebook.

(iii) What is the total number of single and two qubit gates needed in part (ii) to have a relative error in the computed magnetization of at most 2%? Identify m from part (ii) that meets the error threshold, and account for gates needed to change basis for measurement as well as number of shots. Calculate this analytically by analyzing the gate counts in each trotter step and use the `count_ops` function in `qiskit` to verify your result.

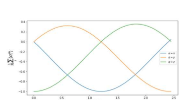
ii)

Done In Qiskit

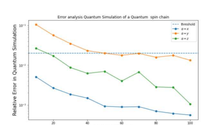
iii)



f) i)



ii)



iii) Qiskit:

OrderedDict([('rz', 650), ('ry', 500), ('cx', 450), ('rx', 200), ('barrier', 151), ('x', 4), ('h', 4), ('measure', 4)])
OrderedDict([('rz', 650), ('ry', 500), ('cx', 450), ('rx', 200), ('barrier', 151), ('x', 4), ('sdg', 4), ('h', 4), ('measure', 4)])
OrderedDict([('rz', 650), ('ry', 500), ('cx', 450), ('rx', 200), ('barrier', 151), ('x', 4), ('measure', 4)])

X 1000000 Shots

analysis: $m \sim \left(\frac{t^2}{\epsilon}\|[H_1, H_2]\|\right)$

$t^2 \quad - \quad 1$

$\frac{}{\epsilon} \to \frac{}{0.02} = 50$

In m we have 27 single qubit & 9 2-qubit gates

$\times 50 \times 3 \times 1000000$ $\times 50 \times 3 \times 1000000$

Iterations / circuits / shots

$= 4050M$ $= 1350M$

+ $(4\times3)2$ statePrep & measure   2 qubit gates

$= 4050M$: 1 qubit gates