Simulating lattice gauge theories on a quantum computer

Tim Byrnes¹ and Yoshihisa Yamamoto¹,²

¹) Quantum Information Science Group, National Institute of Informatics, Tokyo, Japan
²) E. L. Ginzton Laboratory, Stanford University

Outline
1) Introduction to quantum computing and quantum simulation
2) Algorithms for quantum simulation
3) Simulating U(1) LGT
4) Simulating SU(2) LGT
5) Brief comment on SU(3) LGT
6) Conclusions
Quantum computing – a brief introduction

Original idea: Feynman conjectured that a quantum computer may solve quantum problems faster than a classical computer (1981).

• A quantum computer is composed of a number of qubits
The overall state at some time then

\[ |\psi\rangle = \sum_{n_1, n_2 \ldots, n_N} \psi_{n_1, n_2 \ldots, n_N} |n_1, n_2 \ldots, n_N\rangle \]

• We assume we can perform any unitary transformation on the qubits

\[ |\psi\rangle' = U |\psi\rangle \]

• Finally some measurement is made at the end, in some basis

\[ M = \{ |n_1, n_2 \ldots, n_N\rangle, \langle n_1, n_2 \ldots, n_N | \}, \quad n_1, n_2 \ldots, n_N = 0, 1 \]
Quantum computing today

Algorithms:

<table>
<thead>
<tr>
<th>Problem</th>
<th>Time taken on classical computer</th>
<th>Time taken on quantum computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shor’s Factorization</td>
<td>(\sim O(\exp[c(\log N)^{1/3} (\log(\log N))^{2/3}]))</td>
<td>(\sim O((\log N)^2 \log(\log N)))</td>
</tr>
<tr>
<td>Grover’s Database search</td>
<td>(\sim O(N))</td>
<td>(\sim O(\sqrt{N}))</td>
</tr>
</tbody>
</table>

Experiments:

12 qubit NMR

8 qubit ion trap

Still a long way to go before useful quantum computation!!

Negrevergne et al., PRL 96, 170501 (2006)


Much interest towards “Quantum simulation” in quantum computing
Why is quantum simulation interesting?

- Quantum simulation also known to have exponential speedup.

- Although Shor’s algorithm is significant in terms of RSA, do we really need to factorize large numbers?

- Big unsolved problems in physics, e.g. high-Tc superconductors

- Possibility of obtaining useful information with a smaller number of qubits:

<table>
<thead>
<tr>
<th>Shor’s algorithm</th>
<th>Quantum simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sim 10^6$ qubits?</td>
<td>$\sim 100$ qubits?</td>
</tr>
</tbody>
</table>
Two kinds of Quantum Simulation

1) “Analogue” quantum simulator
   e.g. optical lattice experiment

   Idea: artificially construct quantum system of interest (e.g. Bose-Hubbard model)

2) “Digital” quantum simulator
   Use a quantum computer to simulate problems

   Idea: Apply quantum gates to evolve qubits according to Hamiltonian of interest.

   Use quantum algorithms to obtain observables of Hamiltonian.

Simulating Hamiltonians on a quantum computer – “phase estimation”

- Let $|\psi\rangle$ be an eigenstate of some operator $A$. Then

$$\exp[iAt]|\psi\rangle = \exp[i2\pi\phi]|\psi\rangle$$

What is the eigenvalue $\phi$?

![Diagram showing quantum circuit for phase estimation](image)

$U = \exp[iAt]$

- $H \equiv \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$

- $U \equiv \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$

i.e. $H|0\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$

$CU|0\rangle|\psi\rangle = |0\rangle|\psi\rangle$

$CU|1\rangle|\psi\rangle = |1\rangle|U|\psi\rangle$
Phase estimation (continued)

• We now perform a inverse quantum Fourier transform.

\[
\phi = \exp[iAtU]\n\]

\[
\phi = \phi_\pi \phi_\pi \phi_\pi \phi_\pi \rightarrow \sum_{k} \prod_{i=1}^{n} [0\rangle + \exp[2\pi i (2^i \phi) \langle 1\rangle] = \sum_{k} \exp[i2\pi \phi k] k\rangle^{QFT} \rightarrow \phi\]

1) Initialize qubits in a state of high overlap with states of interest. 
\[
|\Psi_{init}\rangle = \sum_{n} \Lambda_n |\epsilon_n\rangle
\]

2) Perform phase estimation of \(U(t) = \exp[-iHt]\)
3) Obtain eigenvalue with probability 
\[
|\Lambda_n|^2
\]
How to perform $U = \exp[-iHt]$?

- Consider simplest case: Simulating a spin Hamiltonian
  - E.g. Heisenberg model
    \[
    H = J \sum_{<ij>} \sigma_i^x \sigma_j^x + \sigma_i^y \sigma_j^y + \sigma_i^z \sigma_j^z
    \]
    Evolve system forwards in time using Trotter decomposition
    \[
    e^{-iHt} = (e^{-iH_1t/m} e^{-iH_2t/m} \cdots e^{-iH_Mt/m})^m
    \]

- Use appropriate combination of gates to do each term.
  Example:
  \[
  \exp[-I \sigma_1^x \sigma_2^z] = \exp[-I (\pi/4) \sigma_1^x] \exp[I \sigma_1^z \sigma_2^z] \exp[I (\pi/4) \sigma_1^x]
  \]

- Previous work shows that fermionic and bosonic systems can be simulated using a similar strategy:
  - **Fermions**: Ortiz et al., Phys. Rev. A 64, 022319 (2001)
  - **Bosons**: Somma et al., quant-ph 0304063

The rest of this talk: how to extend this to LGT?
Simplest case: U(1) Lattice Gauge Theory

The U(1) Hamiltonian:

\[ H_{U(1)} = \sum_{n \in \text{links}} E^2(n) - x \sum_{p \in \text{plaquettes}} \left( Z(p) + Z^+(p) \right) \]

\[ x = 1/ g^4 \]

\[ Z(p) = U_1 U_2 U_3^+ U_4^+ \quad U_i = \exp[i g A_i] \quad [E_i, A_j] = i \delta_{ij} \]

Us act on the links of the lattice. The plaquette operator acts on a square.
Hilbert space of a single link: $U(1)$

- On a single link: Since $[E, A] = i$ and $U = \exp[igA]$

$E$ eigenvalue

\[
\begin{array}{c|c}
2 & UU|0\rangle \\
1 & U|0\rangle \\
0 & |0\rangle \\
-1 & U^+|0\rangle \\
-2 & U^+U^+|0\rangle \\
\vdots \\
\end{array}
\]

$[E, U] = U$  $[E, U^+] = -U^+$

$E|0\rangle = 0$

The $U, U^+$ operators are like translation operators.
Qubit implementation

Register keeps track of E eigenstate.

\[
\tilde{H}_{U(1)} = \sum_i (\tilde{E}_i)^2 - x \sum_p (\tilde{Z}_p + \tilde{Z}^+_p)
\]

\[
Z_p \leftrightarrow \tilde{Z}_p = \tilde{U}_1^+ \tilde{U}_2^+ \tilde{U}_3 \tilde{U}_4
\]

\[
\tilde{U} = \sum_{n=-e_{\text{max}}}^{e_{\text{max}}} \sigma_n^{-} \sigma_{n+1}^{+}
\]

\[
\tilde{E}^2 = \sum_{n=-e_{\text{max}}}^{e_{\text{max}}} n^2 (\sigma_i^z + 1) / 2
\]

Examples:

\[
\tilde{U}|\downarrow\uparrow\downarrow\downarrow\rangle = |\downarrow\downarrow\uparrow\downarrow\rangle
\]

\[
\tilde{E}^2 |\downarrow\uparrow\downarrow\downarrow\rangle = n^2 |\downarrow\uparrow\downarrow\downarrow\rangle
\]
SU(2) Lattice Gauge Theory

The gauge field part of the Hamiltonian is:

\[ H = \sum_{n \in \text{links}} E^2(n) - x \sum_{\text{plaquettes}} \text{Tr} \left( U(n_1) U(n_2) U^+(n_3) U^+(n_4) + H.c. \right) \]

\[ U(n) = \exp \left[ ig \mathbf{\sigma} \cdot \mathbf{A}(n) \right] \]

\( \mathbf{\sigma} \): Generators of SU(N)
e.g. SU(2): Pauli matrices \( \sigma^x, \sigma^y, \sigma^z \)

\( \mathbf{A}(n) \): 3-component field operator

\[ [E_i, A_j] = i \delta_{ij} \]
Hilbert space of a single link: SU(2)

U here is a 2x2 matrix. What happens when we multiply many U’s together?

- From group theory we know Bra-kets hold Clebsch Gordan coefficients, and U’s have the same parameters of the transformation.

\[
U^{j_1}_{m_1 n_1} U^{j_2}_{m_2 n_2} = \sum_{J = |j_1 - j_2|}^{j_1 + j_2} \langle JM | j_1 m_1 ; j_2 m_2 \rangle \langle JN | j_1 n_1 ; j_2 n_2 \rangle U^{J}_{MN}
\]

Bra-kets hold Clebsch Gordan coefficients, and U’s have the same parameters of the transformation.

E.g. \( U^{1/2 \uparrow \downarrow} U^{1/2 \downarrow \uparrow} = -\frac{1}{2} U^{J=0}_{00} + \frac{1}{2} U^{J=1}_{00} \)

\( U^{J} = \exp[i\boldsymbol{\theta} \cdot \boldsymbol{\sigma}^{J}] \) \( \boldsymbol{\sigma}^{J} \) is the J-representation of SU(2)
• If we define

\[ |JMN\rangle \equiv U_{MN}^J |0\rangle \]

• Then we can regard the previous equation as an operator equation

\[
U_{mn} |JMN\rangle = \sum_{J'=|J-1/2|}^{J+1/2} \langle J'M';1/2;m,JM|J'N';1/2;n,JN|J'M'N'\rangle
\]

• Each time the Hamiltonian acts on a particular link it produces two states, with coefficients determined by Clebsch-Gordan coefficients. (Similar to Tensor operators in Wigner-Eckart theorem)
Implementing the Tensor operator

- A similar type of problem was examined by Bacon, Chuang, Harrow: quant-ph/0407082. A quantum circuit to do a Clebsch Gordan transformation

\[ X : C_x |s\rangle |m\rangle = |s\rangle |m + s\rangle \]

\[ R_y(\theta_{J,m'}) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \]

\[ \cos \theta = \sqrt{\frac{J + m' + 1/2}{2J + 1}} \]

\[ |\frac{1}{2}, s\rangle |J, m\rangle \rightarrow \sum_{J' = |J - 1/2|}^{J + 1/2} \langle J', M' = m + s |J, m; \frac{1}{2}, s\rangle |J', M'\rangle \]
Qubit implementation SU(2)

- For each link, assign 3 sets of registers:
  c.f. For U(1) LGT: \( |L\rangle \) (1 register)

- Using a modification of the circuit, we obtain the transformation between gauge operators and spin operators

\[
U_{mn} \leftrightarrow V^{\pm\pm} = M^{\pm} N^{\pm} \left[ J^{+} Z_{1\pm} + J^{-} Z_{2\pm} \right]
\]

\[
J^{\pm} | J, M, N \rangle = | J \pm 1/2, M, N \rangle
\]
\[
M^{\pm} | J, M, N \rangle = | J, M \pm 1/2, N \rangle
\]
\[
N^{\pm} | J, M, N \rangle = | J, M, N \pm 1/2 \rangle
\]

\[
Z_{1\pm} = \sqrt{\frac{(J \pm M + 1)(J \pm N + 1)}{(2J + 1)(2J + 2)}}
\]
\[
Z_{2\pm} = \sqrt{\frac{(J \mp M)(J \mp N)}{2J(2J + 1)}}
\]

\[
U_{mn} | J M N \rangle = \sum_{J' = |J - 1/2|}^{J + 1/2} \langle J' M' | \frac{1}{2} - m ; J M \rangle \langle J' N' | \frac{1}{2} n ; J N \rangle | J' M' N' \rangle
\]
These expressions are simply plugged into the Hamiltonian:

\[
H = \sum_{n \in \text{links}} J^2(n) - x \sum_{\text{plaquettes}} \text{Tr} \left( V(n_1) V(n_2) V^+(n_3) V^+(n_4) \right) + H.c.
\]

Total number of qubits =
(no links)x(no qubits in register for \( |J,M,N\rangle \))
Efficiency Issues

- The greatest overhead in the simulation is performing the time evolution operator $e^{-iHt}$

- Dependence with the total number of links $N$: (worst case)
  - Number of qubits:
    $\propto \text{(no. qubits per link)} \times \text{(no. links)} = pN$
  
  - Number of operations: (at best)
    $\propto \text{(no terms in Hamiltonian)} \times \text{(Trotter)} = mN$
    
    \[ e^{-iHt} = \left( e^{-iH_1 t/m} e^{-iH_2 t/m} \ldots e^{-iH_m t/m} \right)^m \]

    - At worst each term requires long swap operations across entire set of qubits in quantum computer memory.

\[ \propto mN \leq \text{Total number of operations} \leq \propto m^pN^2 \]
What happens for SU(3)?

- Each link requires 8 registers:

\[ U^R_{\alpha\beta} \ket{0} = \left| \begin{array}{c} p, q, T_1, T_1^z, Y_1, T_2, T_2^z, Y_2 \\ R \alpha \beta \end{array} \right| \]

- Formulas for adding 3 and 3* representations of Clebsch-Gordan coefficients exist: do similar transformation of U’s.
Summary and Conclusions

- Reformulated the lattice gauge theory Hamiltonian in terms of qubit operations. Transformations may be performed for SU(N) gauge theories.
- The number of qubit operations a low-degree polynomial in the number of lattice sites to implement the time evolution operator $e^{-iHt}$.
- A simulation on a quantum computer is virtually an exact calculation of such field theories, which are notoriously difficult.
- Including fermions to SU(3) will give full QCD. This only adds one qubit, compared to the large number for each gauge link.
- For a spin-based Q.C., this involves doing a Jordan-Wigner transformation, which involves long-range products of spins. (Verstraete et al. cond-mat 0508353)?

For further details, see: Byrnes and Yamamoto, Phys. Rev. A 73, 022328 (2006)