

SU(2) Lattice Gauge Pair Hopping Constructs Suitable for Implementation on Quantum Computers

Patrick Dreher

Department of Computer Science / Department of Physics

North Carolina State University

Outline

Can Quantum Computers (QC) Play a Role in Lattice QCD Simulations?

1. Motivation

- Increases in the number of qubits in a QC and improvements in quantum volume are reaching a threshold of becoming sufficiently robust for LQCD “development sandboxes”

2. Goal

- 20 to 50 qubit QC machines that are coming online within the next year
- Opportunity to begin exploration if/how QC may play role in Lattice QCD (LQCD)

3. Why consider quantum computing

- Types of LQCD problems where quantum computers can make progress
- Identify basic LQCD building blocks that may be modified for quantum computation

4. Illustrate an example how re-formulating $SU(2)$ links on a lattice may provide a potential algorithm for implementation on a quantum computer

5. Comments - toward a role for quantum computers in LQCD research

Why Use Quantum Computing

- There are important problems in LQCD that cannot be addressed using simulations with actions on Euclidean metrics with conventional computers
 - QCD at large baryon density (understanding the state of matter inside neutron stars)
 - Real-time dynamics of strongly interacting matter (heavy-ion collisions)
 - The critical endpoint of the chiral transition
- Similar problems also arise in strongly correlated systems in condensed matter physics
 - Non-abelian SU(2) variant of quantum spin liquids
 - Majorana edge states for system of spinless fermions on two wires coupled by pair hopping
 - Topological phases
 - Topological superconductivity in 1-d condensed matter systems
 - Topological quantum computing and quantum memory

Implementing a Basic LQCD Process on a Quantum Computer

- In LQCD complex matrices describe link variables -- the parallel transporters of the gauge fields between adjacent lattice points
- Gauge fields are complex matrices that take values in the gauge group
- Because the group space is a continuous manifold -- the local Hilbert space is infinite dimensional in the standard Wilson formulations
- One of the challenge is to construct a quantum computing formulation so that the links can be represented with a handful of discrete states suitable for mapping to qubits on a universal quantum computer

How does this impact the design of an LQCD quantum computing algorithm?

A Basic Characteristic of a Conventional Computer

- Conventional Computer
 - Uses a binary data representation for floating point and integer quantities
 - Hardware is designed and constructed on this base 2 formalism
 - Binary representations reflect the lowest level structure for system and application software



A Basic Quantum Computer (QC) Characteristic

- Quantum Computer

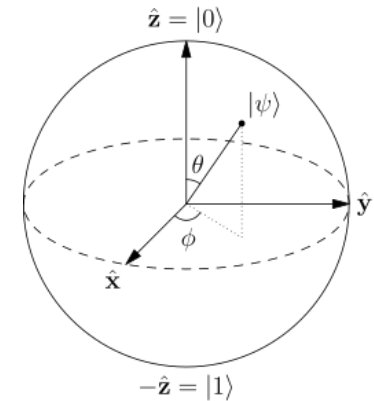
- Uses a data representation known as a qubit with the property that it can represent many different values simultaneously

$$|a\rangle = \alpha|0\rangle + \beta|1\rangle \quad |\alpha|^2 + |\beta|^2 = 1$$

$$|a\rangle = e^{i\gamma} \left[\cos\left(\frac{\theta}{2}\right) |0\rangle + e^{i\phi} \sin\left(\frac{\theta}{2}\right) |1\rangle \right]$$

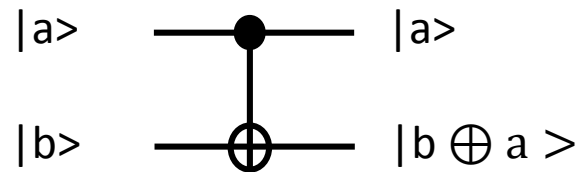
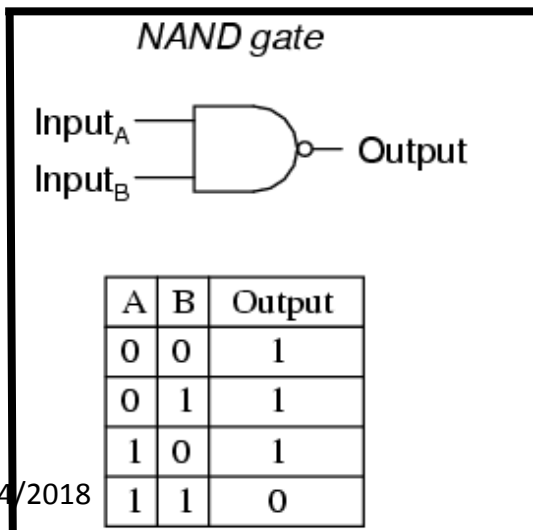
- Design an algorithm based on the properties of qubits and quantum mechanics
- Need multiple qubits to implement an algorithm
- Output is an expectation value measured through many system samplings

- The conventional and QC architectures approach a computational problem from very different perspectives → have very different hardware architectures and software environments



Differences in Basic Computer Logic Structure

- Quantum computer
 - Qubits obey the postulates of quantum mechanics - properties of reversibility and unitarity
 - Manipulation of the qubits also accomplished through gates
 - Differs from a conventional computer because the design must enforce the postulates of quantum mechanics
- Conventional computer
 - Uses standard logic gates (NAND, etc.)



$$U_{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$U_{CNOT}^\dagger U_{CNOT} = I$$

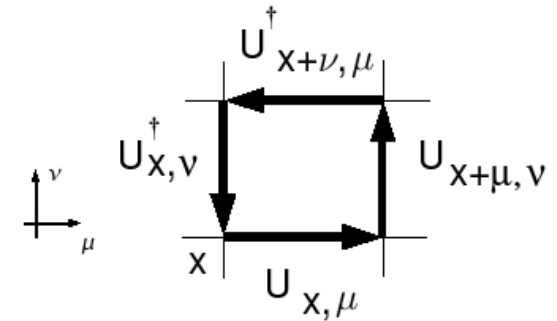
$$|a\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |b\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \begin{array}{ll} |aa\rangle \rightarrow |aa\rangle & |ba\rangle \rightarrow |bb\rangle \\ |ab\rangle \rightarrow |ab\rangle & |bb\rangle \rightarrow |ba\rangle \end{array}$$

Patrick Dreher - Lattice 2018

Assembling a Basic Lattice QCD Construct Using a Conventional Computer

- Construct a lattice using a connected grid of points and links
- Construct a plaquette by making closed loop of points and links

$$P_{\mu\nu} = \sum_{x, \mu > \nu} u_{\mu}(x) u_{\nu}(x + \mu) u_{\mu}^{\dagger}(x + \nu) u_{\nu}^{\dagger}(x)$$



- Prepare a conventional computation by loading
 - Complex vectors representing fermions on the lattice sites
 - Complex matrices representing the gauge transformations on the links between the sites
- Computation starts and proceeds iteratively and completes when certain conditions are met

Assembling a Basic Lattice QCD Construct Using a Quantum Computer

- Start by constructing a point and link formalism a similar manner as the conventional approach to form a simple Action
- Rather than Wilson approach -- use the Quantum Link Model (QLM) formulation -- originally intended for cluster algorithms*
- Compared to Wilson's LGT formulation, the real and imaginary parts of the links $U_{x,y}^{i,j}$ in a non-abelian QLM are now represented by non-commuting Hermitian operators that obey gauge covariance

$$U_{x,\mu} = U_{x,\mu}^0 + i\vec{U}_{x,\mu} \cdot \vec{\sigma}$$

$$U_{x,\mu}^\dagger = U_{x,\mu}^0 - i\vec{U}_{x,\mu} \cdot \vec{\sigma}$$

$$U'_{x,\mu} = \prod_y e^{(-i\vec{\alpha}_y \cdot \vec{G}_y)} U_{x,\mu} \prod_z e^{(i\vec{\alpha}_z \cdot \vec{G}_z)} = e^{(i\vec{\alpha}_x \cdot \vec{\sigma})} U_{x,\mu} e^{(-i\vec{\alpha}_{x+\mu} \cdot \vec{\sigma})}$$

* Several papers and authors Wiese, Chandrasekharan, Schlittgen, Brower, et. al.

Construct A Formalism For Building Sites And Links on a Quantum Computer

- Using the commutation relations between \vec{G}_x and link matrices $U_{x,y}^{i,j}$

$$\left[\vec{G}_x, U_{y,\mu} \right] = \delta_{x,y+\hat{\mu}} U_{y,\mu} \vec{\sigma} - \delta_{x,y} \vec{\sigma} U_{y,\mu}$$

$$\left[\vec{G}_x, U_{y,\mu}^\dagger \right] = \delta_{x,y} U_{y,\mu}^\dagger \vec{\sigma} - \delta_{x,y+\hat{\mu}} \vec{\sigma} U_{y,\mu}^\dagger$$

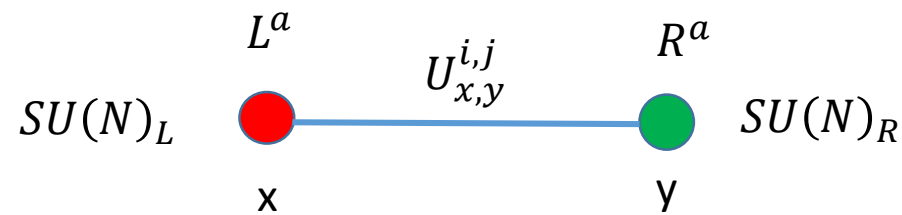
- A representation for \vec{G}_x that satisfies these commutation relations is

$$\vec{G}_x = \sum_{\mu} (\vec{R}_{x-\hat{\mu},\mu} + \vec{L}_{x,\mu})$$

where $\vec{R}_{x,\mu}$ and $\vec{L}_{x,\mu}$ represent gauge transformations on the right and left ends of the link

Links from Wilson and QLM Perspectives

- Observe that the quantum link models allow a factorization of the quantum link operators on the lattice into so-called rishon constituents (in contrast to Wilson's lattice gauge theory)
- Express $\vec{R}_{x,\mu}$ and $\vec{L}_{x,\mu}$ as “rishons” that factorize the quantum link operators into fermions at the ends of the link



The $\vec{R}_{x,\mu}$ and $\vec{L}_{x,\mu}$ generate an $SU(N)_R \otimes SU(N)_L$ algebra on each link

Rishon Properties and the Quantum Link Model

- $\vec{R}_{x,\mu}$ and $\vec{L}_{x,\mu}$ act as anti-commuting operators $c_{x,\pm k}^i$ and $c_{x,\pm k}^{i\dagger}$ at site x with a link direction $\pm k$

- The rishons obey a set of anti-commutation relations

$$\{c_{x,\pm k}^i, c_{y,\pm l}^{j\dagger}\} = \delta_{xy}\delta_{\pm k,\pm l}\delta_{ij} \quad \{c_{x,\pm k}^i, c_{y,\pm l}^j\} = \{c_{x,\pm k}^{i\dagger}, c_{y,\pm l}^{j\dagger}\} = 0$$

- In terms of the rishons $\vec{R}_{x,\mu}$ $\vec{L}_{x,\mu}$ and $U_{x,y}^{i,j}$ can be expressed as

$$L_{x,y}^a = c_{x,+}^{i\dagger} \lambda_{i,j}^a c_{x,+}^j \quad U_{x,y}^{i,j} = c_{x,+}^i c_{y,-}^{j\dagger} \quad R_{x,y}^a = c_{y,-}^{i\dagger} \lambda_{i,j}^a c_{y,-}^j$$

- The quantum link operator U_{xy} shuffles a rishon from one end of the link to the other, keeping the total number of rishons per link fixed

$$N_{xy} = c_{xy,+}^{i\dagger} c_{xy,+}^i + c_{xy,-}^{i\dagger} c_{xy,-}^i$$

Advantages to Using QLM Formulation for Links on QC

- The number of distinct fermion states on the link is dependent on
 - the order of the gauge group
 - The embedding algebra within $SU(N)_L \otimes SU(N)_R$
- All other standard commutation relations with Wilson's lattice formulation including the Hamiltonian and gauge transformations are still valid in the Quantum Link Model
- The advantage to this alternative formulation of non-Abelian gauge theories is that
 - The link Hilbert space on the lattice is now finite dimensional
 - Physics on the link represented in a creation/annihilation operator formulation
 - This allows the LQCD link to be reconfigured from a conventional to a QC

A Physics Example

- Consider a simple Hamiltonian containing only non-abelian gauge fields
- Select the finite temperature non-linear sigma model
- SU(2) case models chiral phase transition with $SU(2)_L \otimes SU(2)_R$
- Physics can be expressed in the form of a Heisenberg model

$$Z = \int \prod_k [dU_k] \exp\left(\frac{1}{2} \beta \left[\sum_{\langle ij \rangle} \text{tr} U_i^\dagger U_j + \text{h.c.} \right]\right)$$

- The QLM expression for the link $U_{x,y}^{i,j} = c_{x,+}^i c_{y,-}^{j\dagger}$ corresponds to the Hamiltonian expression

$$\mathbf{H} = \text{Tr} U_x U_y^\dagger + \text{h.c.} = c_{x,+}^i c_{x,-}^{j\dagger} c_{y,-}^j c_{y,+}^{i\dagger} + \text{h.c.}$$

Rishons and the Heisenberg Model for SU(2)

- SU(2) quantum link matrix elements $U_{x,y}^{i,j}$ can be expressed in terms of the rishon anti-commuting operators [-/+ corresponds to x/y link ends]

$$U_{xy}^{11} = c_{xy,+}^{1\dagger} c_{xy,-}^1 + c_{xy,-}^2 c_{xy,+}^2$$

$$U_{xy}^{12} = c_{xy,+}^{2\dagger} c_{xy,-}^1 - c_{xy,-}^{2\dagger} c_{xy,+}^1$$

$$U_{xy}^{21} = c_{xy,+}^{1\dagger} c_{xy,-}^1 - c_{xy,-}^{1\dagger} c_{xy,+}^2$$

$$U_{xy}^{22} = c_{xy,+}^{2\dagger} c_{xy,-}^2 + c_{xy,-}^{1\dagger} c_{xy,+}^1$$

- For SU(2) one of the representations in this QLM allows $N_{xy} = 2$ rishons per link for a total of 6 fermion states on the link
- In this representation there are 4 fermion states where the rishons reside on opposite ends of the link
- Due to their fermionic nature it is also possible for the two rishons to form a color-singlet and sit on the same end of the link

Constraints for Mapping the Rishon Operator Hamiltonian Formulation to Basic Logic Gates of a Quantum Computer

- Two Constraints
- 1st : Must decompose every fermionic operator into Pauli strings
 - Transform Hamiltonian operator into a time evolution \mathbf{H} as $e^{(i\mathbf{H}t)}$
 - Approximate $e^{(i\mathbf{H}t)}$ by sequences of exponentiated Pauli strings $\exp(i\theta_\sigma s\sigma)$ with real weights where \mathbf{s} is a time slice of t (Trotterization) and \mathbf{H} is

$$\mathbf{H} = \sum_{\sigma \in \{X, Y, Z, I\}} \theta_\sigma \times \sigma$$

$\sigma \in \{X, Y, Z, I\}$ are the Pauli matrices X , Y , and Z acting on one qubit

- Numbers, signs and values of the time slices and ordering of the exponentiated strings impact accuracy of the simulation
- 2nd : Hamiltonian must be Hermitian $\rightarrow \theta_\sigma$ must be real

Map the Rishon Operator Hamiltonian Formulation to the Basic Logic Gates of a Quantum Computer

- Need a process that maps Hamiltonians of fermionic systems onto Pauli spin matrices that can then be connected to rotation gates on an n qubit QC
- Use a Jordan-Wigner transformation* to map Hamiltonians of fermionic systems onto Pauli spin matrices that can be connected to rotation gates on a quantum computer for a qubit i [multiplications are tensor products]

$$c_j \triangleq Z^{\otimes j-1} \otimes \sigma^- \otimes I^{\otimes n-j}$$

$$c_j^\dagger \triangleq Z^{\otimes j-1} \otimes \sigma^+ \otimes I^{\otimes n-j}$$

where

$$\sigma^- = |0\rangle\langle 1| = \frac{1}{2}(X + iY)$$

$$\sigma^+ = |1\rangle\langle 0| = \frac{1}{2}(X - iY)$$

- This will produce a complex set of time sequence gates operations that can be executed on a quantum computer

*Alternatives include Verstraete-Cirac mapping and Bravyi-Kitaev transform to find a suitable representation

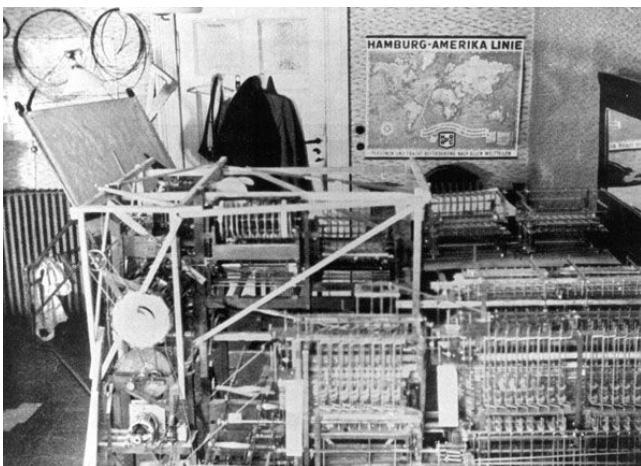
QC Gate Operations to Physics Expectation Values

- For the example illustrated the simplest idea may be to apply the transformation to each individual rishon operator
- Downside is that it can generate very complex set of gate operations
- For the $SU(2)$ example of two rishon hopping it may be advantageous to map pairs of operators together rather than individually to reflect the physics requirements that both rishons shift together on the link
- Will reduce the number of gate operations that must be programmed
- Apply algorithm (such as quantum phase estimation, adiabatic simulations or variational quantum eigensolvers) that
 - Inputs ansatz states
 - Prepares eigenstates
 - Outputs a scalar resultant (ex. expectation value of the energy)

Start Playing in the Sandbox

- This talk illustrates a toy example of a process that can transform a basic LQCD process into a format compatible for a quantum computer
- 20 to 50 qubit quantum computers are coming online within the next year
- These multi-qubit machines may be sufficiently robust to handle these levels of gate operations in a universal computing mode
- There is a potential opportunity to
 - Develop/test QC implementations of LQCD quantum computing building blocks
 - Develop/improve early programming models and utilities for translating algorithms into gate operations (ex. XACC at ORNL, IBM-Q Qiskit, etc.)

Looking Forward



- Early programmable Z1 computer
- Originally created by Germany's Konrad Zuse in his parents' living room in 1936 to 1938



- Early programmable quantum computer (IBM-Q)
- Several generations of quantum computers
 - 5, 16, 20 and testing 50 qubit machines



- It has been 80 years since the Z1 was introduced
- Testing has already begun with a 50 qubit machine while also improving the quantum volume
- Somewhere around 100 qubits a QC with good quantum volume control may cross the theoretical threshold of exceeding the computational power of the largest supercomputers
- It is unlikely that it will take another 80 years before QC will play a role in LQCD
- Begin preparing LQCD building blocks to be ready to utilize larger QC in the future