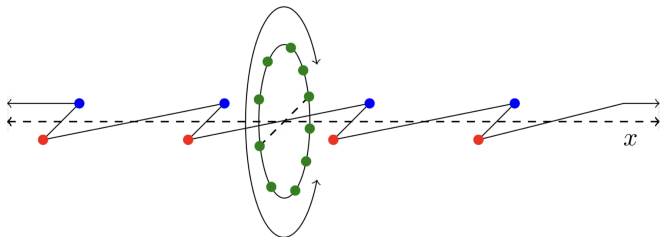


Lattice July 2021

## Quantum Algorithms for Simulating the Lattice Schwinger Model



Alexander F. Shaw

University of Maryland - College Park

Based on work of same title in Quantum 4, 306 (2020)  
by AF Shaw, P Lougovski, JR Stryker and N Wiebe

# Why **Quantum** Simulation?

\*[Yamazaki et. al (2010) PhysRevD.81.111504]

- | Using fundamental models to calculate physics at longer length scales is hard.

# Why Quantum Simulation?

\*[Yamazaki et. al (2010) PhysRevD.81.111504]

- | Using fundamental models to calculate physics at longer length scales is hard.
- | Access different physics (real-time dynamics)

# Why Quantum Simulation?

\* [Yamazaki et. al (2010) PhysRevD.81.111504]

- | Using fundamental models to calculate physics at longer length scales is hard.
- | Access different physics (real-time dynamics)
- | Classical methods have been very successful, but hit roadblocks. (exponential resource scalings, sign problems)

# Goal: Estimate Observables from QCD

# Goal: Estimate Observables from QCD

How to, using real-time dynamics:

# Goal: Estimate Observables from QCD

How to, using real-time dynamics:

1. Map states of your theory to states of the simulator

# Goal: Estimate Observables from QCD

How to, using real-time dynamics:

1. Map states of your theory to states of the simulator
2. Prepare an initial state  $|\phi_0\rangle$



# Goal: Estimate Observables from QCD

How to, using real-time dynamics:

1. Map states of your theory to states of the simulator
2. Prepare an initial state  $|j\rangle_0$
3. Compute the time-evolved state  $e^{-iHt}|j\rangle_0$ .

# Goal: Estimate Observables from QCD

How to, using real-time dynamics:

1. Map states of your theory to states of the simulator
2. Prepare an initial state  $|j_0\rangle$
3. Compute the time-evolved state  $e^{-iHt}|j_0\rangle$ .
4. Measure in the basis of some observable

# Goal: Estimate Observables from QCD

How to, using real-time dynamics:

1. Map states of your theory to states of the simulator
2. Prepare an initial state  $|j\rangle_0$
3. Compute the time-evolved state  $e^{-iHt}|j\rangle_0$ .
4. Measure in the basis of some observable  $O$ .
5. Repeat to estimate  $\langle O \rangle$ .

# Goal: Estimate Observables from QCD

How to, using real-time dynamics:

1. Map states of your theory to states of the simulator
2. Prepare an initial state  $|j_0\rangle$
3. Compute the time-evolved state  $e^{-iHt}|j_0\rangle$ .
4. Measure in the basis of some observable  $O$ .
5. Repeat to estimate  $\langle O \rangle$ .

How hard is this to do on a quantum computer? Depends on the computational model.

# Goal: Estimate Observables from QCD

How to, using real-time dynamics:

1. Map states of your theory to states of the simulator
2. Prepare an initial state  $|0\rangle$
3. Compute the time-evolved state  $e^{-iHt}|0\rangle$
4. Measure in the basis of some observable  $O$
5. Repeat to estimate  $\langle O \rangle$ .

How hard is this to do on a quantum computer? Depends on the computational model.

	Cheap	Expensive
Near-term (NISQ)	Single-qubit rotations	Entangling gates (CNOT)
Far-term (Fault-Tolerant)	Clifford + CNOT gates	T-gates

QCD is too hard to start with.

QCD is too hard to start with.

The Lattice Schwinger Model\*

odd sites - electrons

even sites - positrons

link sites - electric field

)=  
continuum limit

QED in (1+1)D

# QCD is too hard to start with.

## The Lattice Schwinger Model\*

odd sites - electrons

even sites - positrons

link sites - electric field

)=  
continuum limit

QED in (1+1)D

Simplest model that replicates features of QCD (confinement, spontaneous breaking of chiral symmetry).

\* [Kogut, Susskind (1975) 10.1103/PhysRevD.11.395]



# Computing time evolution

## The Lattice Schwinger Model

Computational basis = occupation basis.

# Computing time evolution

## The Lattice Schwinger Model

Computational basis = occupation basis EX: for a fermion on an odd site (after Jordan-Wigner):

computational basis	occupation basis
$j0i$	presence of electron
$j1i$	vacuum

# Computing time evolution

## The Lattice Schwinger Model

Computational basis = occupation basis EX: for a fermion on an odd site (after Jordan-Wigner):

computational basis	occupation basis
j0i	presence of electron
j1i	vacuum

The value of the electric field is represented as a binary integer in a qubit register, one for each link.

# Computing time evolution

Decompose the Schwinger Model Hamiltonian

$$H_{\text{Schwinger}} = \sum_r \left[ \frac{1}{2} E_r^2 + \frac{1}{2} \left( \sum_r \hat{y}_r \hat{y}_r \right) + \sum_r \left( \hat{y}_r \hat{y}_{r+1} \right) \right]$$

$$= \underbrace{H_{\text{E-eld}} + H_{\text{mass}}}_{\text{diagonal in computational basis}} + \underbrace{H_{\text{interaction}}}_{\text{linear combination of tensor product of pauli matrices}}$$

$$= \prod_{j=1}^n H_j \quad \text{Simulatable!}$$

# Computing time evolution

Decompose the Schwinger Model Hamiltonian

$$\begin{aligned}
 H_{\text{Schwinger}} &= \sum_r \left[ \frac{1}{2} E_r^2 + \frac{1}{2} \left( \sigma_r^x \sigma_{r+1}^x + \sigma_r^y \sigma_{r+1}^y \right) \right] + \sum_r \left[ \frac{1}{2} \left( \sigma_r^z \sigma_{r+1}^z + \sigma_r^x \sigma_{r+1}^x + \sigma_r^y \sigma_{r+1}^y \right) \right] \\
 &= \underbrace{\left[ \sum_r \left( \frac{1}{2} E_r^2 + \frac{1}{2} \sigma_r^z \sigma_{r+1}^z \right) \right]}_{\text{diagonal in computational basis}} + \underbrace{\sum_r \left[ \frac{1}{2} \left( \sigma_r^x \sigma_{r+1}^x + \sigma_r^y \sigma_{r+1}^y \right) \right]}_{\text{linear combination of tensor product of pauli matrices}} \\
 &= \prod_{j=1}^n H_j \quad \text{Simulatable!}
 \end{aligned}$$

$$\begin{aligned}
 V_{\text{trotter step}}(t) &:= \prod_{j=1}^n e^{-iH_j t/2} \prod_{j=m}^1 e^{-iH_j t/2} \\
 e^{-iH_{\text{Schwinger}} t} &= \prod_{x,y>x} \frac{1}{12} \sum_{x; y>x} k[[H_x; H_y]; H_z] k t^3 \\
 &+ \prod_{x; y>x} \frac{1}{24} \sum_{x; y>x} k[[H_x; H_y]; H_x] k t^3 \equiv \text{(lattice parameter} \times \text{evolution time)}
 \end{aligned}$$

## What's in the Paper:

I Decomposition of  $H_{\text{Schwinger}} = \sum_j P_j H_j$

## What's in the Paper:

- | Decomposition of  $H_{\text{Schwinger}} = \sum_j H_j$
- | Circuits to implement each  $e^{iH_j t}$  in both the near-term (NISQ) and far-term settings.

## What's in the Paper:

- | Decomposition of  $H_{\text{Schwinger}} = \sum_j^P H_j$
- | Circuits to implement each  $e^{iH_j t}$  in both the near-term (NISQ) and far-term settings.
- | Calculate  
(lattice parameters)  $e^{iH_{\text{Schwinger}} t} V_{\text{trotter step}}(t)$



## What's in the Paper:

- | Decomposition of  $H_{\text{Schwinger}} = \sum_j^P H_j$
- | Circuits to implement each  $e^{iH_j t}$  in both the near-term (NISQ) and far-term settings.
- | Calculate  
(lattice parameters)  $e^{iH_{\text{Schwinger}} t} V_{\text{trotter step}}(t)$
- | Upper bound on sufficient quantum-computational resources required for an arbitrary simulation:

## What's in the Paper:

- | Decomposition of  $H_{\text{Schwinger}} = \sum_j P_j H_j$
- | Circuits to implement each  $e^{iH_j t}$  in both the near-term (NISQ) and far-term settings.
- | Calculate (lattice parameters)  $e^{iH_{\text{Schwinger}} t} V_{\text{trotter step}}(t)$
- | Upper bound on sufficient quantum-computational resources required for an arbitrary simulation:

COST(total evolution time, error in final state, lattice parameters)

$$2 \Theta \frac{N^{3-2T} x^{1-2}}{1-2}$$

where  $N$  = lattice size,  $T$  = evolution time,  $\epsilon$  = E-eld cut-off, and  $x = 1 - (ag)^2$  (a lattice spacing,  $g$  coupling constant)

## Data and Conclusion (Room for Improvement)

We compile these results further with estimation of mean positron/electron density :

Near-term (NISQ) Simulation (no ancilla)

$x = (ag)^2$  with a lattice spacing and coupling constant

$g$  = error in CNOT channel

$\tilde{x}^2$  = worst case mean square error in mean positron density

# Data and Conclusion (Room for Improvement)

Far-term Simulation ( $N$  is size of lattice,  $\beta$  is electric cutoff )

# Questions on the Road to Quantum Simulating QCD:

# Questions on the Road to Quantum Simulating QCD:

- | Can our algorithms generalize to higher dimensions and  $SU(N)$ ?

# Questions on the Road to Quantum Simulating QCD:

- | Can our algorithms generalize to higher dimensions and  $SU(N)$ ?
- | How do other algorithms (LCU, Qubitization) compare?

# Questions on the Road to Quantum Simulating QCD:

- | Can our algorithms generalize to higher dimensions and  $SU(N)$ ?
- | How do other algorithms (LCU, Qubitization) compare?
- | Could other formulations of the model be beneficial? ( $SU(2)$  - Quantum Link Model, Loop-String-Hadron)



# Questions on the Road to Quantum Simulating QCD:

- | Can our algorithms generalize to higher dimensions and  $SU(N)$ ?
- | How do other algorithms (LCU, Qubitization) compare?
- | Could other formulations of the model be beneficial? ( $SU(2)$  - Quantum Link Model, Loop-String-Hadron)
- | How can we best realize gauge invariance during simulation?

Our work is a benchmark to aid in answering these questions.

LCU - [Childs, Wiebe (2012) 10.26421/QIC12.11-12]

Qubitization - [Low, Chuang (2019) 10.22331/q-2019-07-12-163]

Quantum Link Model - [Chandrasekharan, Wiese (1996) 10.1016/S0550-3213(97)80041-7]

Loop-String-Hadron - [Raychowdhury, Stryker (2019) 10.1103/PhysRevD.101.114502]

## Preliminary Results - SU(2):

Using abelian gauge invariant hopping decomposition\* - Discussed in J. Stryker's talk **this evening, 10:45pm EST**:

$T \frac{1}{a^2 g^2}; \Lambda; \frac{2m}{g^2 a}; N; T = \text{Trotter Steps to reach error } 10^{-3}$ :

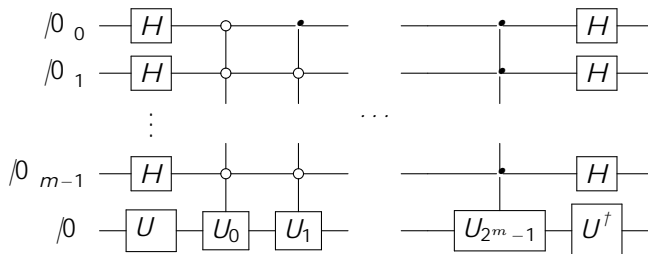
## Preliminary Results - SU(2):

Using abelian gauge invariant hopping decomposition\* - Discussed in J. Stryker's talk **this evening, 10:45pm EST**:

$T \frac{1}{a^2 g^2}; \Lambda; \frac{2m}{g^2 a}; N; T = \text{Trotter Steps to reach error } 10^{-3}$ :

\*[J. Stryker (2021) arXiv:2105.11548]

# Using amplitude estimation to estimate $\langle \hat{O} \rangle$



If  $\hat{O} = \sum_{a=0}^{2^m-1} \hat{U}_a$ ,  $\hat{U}_a$  unitary, then above circuit ends up in:

$\text{Prob}(\text{measure } |j00\dots 0i\rangle) = C \langle \hat{O} \rangle$ ,  $C$  known constant.