The Future of Computation: Unleashing the Power of Quantum Computers

Peter P. Orth (Iowa State University & Ames Laboratory)
Collaboration with Josh Aftergood, Noah Berthusen, I-Chi Chen, Joao Getelina, Niladri Gomes, Kai-Ming Ho, Thomas Iadecola, Anirban Mukherjee, Cai-Zhuang Wang, Yongxin Yao, Feng Zhang (Iowa State University & Ames Laboratory)

Physics Colloquium, MSU Mankato, April 11, 2022

References:
• I.-C. Chen et al., arXiv:2203.08291 (2022)
• N. Gomes et al., Adv. Qu. Tech. 2100114 (2021)
• Y. Yao et al., PRX Quantum 2, 030307 (2021)
What is a quantum computer?

A quantum computer is a programmable computing device that works according to the fundamental physical laws of quantum mechanics.

Properties of a digital quantum computer

- Contains **qubits** = quantum mechanical 2-level systems = Spin-1/2
- Sounds similar to a classical bit {0, 1}, but is a totally different beast
- Can be in a **superposition** of two basis states \( |0\rangle \) and \( |1\rangle \)

Hamiltonian of a qubit

\[
H = -\omega_0 \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}
\]

Pauli-Z operator

**General state of a qubit**

\[
|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle
\]
Quantum gate operations

Properties of a digital quantum computer

> Contains qubits = quantum mechanical 2-level systems
> **Quantum gate operations** act on qubits and change their states
  > Sounds similar to classical gates {NOT, OR, ...}, but must be reversible
  > Single-qubit gates is unitary 2x2 matrix [SU(2)] = Rotations on Bloch sphere

\[
X = \begin{pmatrix}
0 & 1 \\
1 & 0 \\
\end{pmatrix} \quad \Rightarrow \quad X|0\rangle = |1\rangle, \quad X|1\rangle = |0\rangle
\]

\[
H = \frac{1}{\sqrt{2}} \begin{pmatrix}
1 & 1 \\
1 & -1 \\
\end{pmatrix} \quad \Rightarrow \quad H|0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle), \quad H|1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)
\]

> X gate flips qubit. Acts like a NOT gate.
> H: generates superposition (Hadamard gate).
Multi-qubit gates and entanglement

Properties of a digital quantum computer

- Contains qubits = quantum mechanical 2-level systems
- **Quantum gate operations** act on qubits and change their states
  - Sounds similar to classical gates {NOT, OR, ...}, but must be reversible
  - Single-qubit gates is **unitary** 2x2 matrix \([SU(2)] = \text{Rotations on Bloch sphere}\)
  - Multi-qubit gates are unitary rotations in \(SU(2^N)\)

**Controlled-NOT (CNOT)**

- CNOT = \[
\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0
\end{pmatrix}
\]
- CNOT creates entanglement

**Quantum circuit (Bell pair)**

- \(|0\rangle_{q0} \rightarrow \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)|0\rangle \rightarrow \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)\)
- Non-local quantum correlations
- Cannot be written as \(|\psi_1\rangle \otimes |\psi_2\rangle\)
Measurement of qubits

Properties of a digital quantum computer

> Contains qubits = quantum mechanical 2-level systems
> Quantum gate operations act on qubits and change their states
> **Quantum state is transformed to classical information by measurements**
  > Choose a basis in which to measure (usually Pauli-Z)
  > Measurement outcomes are operator eigenvalues: +1,-1 for Pauli-Z
  > Measurement outcome is **probabilistic** (Born rule)

\[
|\psi\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)
\]

**Measurement probabilities**

\[
|\langle 00|\psi\rangle|^2 = |\langle 11|\psi\rangle|^2 = \frac{1}{2}
\]
\[
|\langle 10|\psi\rangle|^2 = |\langle 01|\psi\rangle|^2 = 0
\]

> One of four bitstrings is measured each time
> Probability given by quantum wavefunction
> Infer that by repeated measurements (build histogram of #(observed bitstrings)
Interference in quantum circuits

Quantum circuit (interference):

\[ |0\rangle_{q_0} \rightarrow \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)|1\rangle \rightarrow \frac{1}{\sqrt{2}} (|0\rangle + e^{i\varphi}|1\rangle)|1\rangle \rightarrow \frac{1}{\sqrt{2}} (\cos \frac{\varphi}{2} |0\rangle - i \sin \frac{\varphi}{2} |1\rangle)|1\rangle \]

Interference of different circuit paths

- Outcome depends on phase difference \( \varphi \) along two paths
  - Qubit q0 in state \( |0\rangle \) for \( \varphi=0 \),
  - Qubit q0 in state \( |1\rangle \) for \( \varphi=\pi \)

Controlled-Z phase gate

\[ CU_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\varphi} \end{pmatrix} \]

Corresponding light interferometer (Mach-Zehnder)

- Hadamard gate H acts as semi-transparent mirror
- Qubit q0 stores information about phase evolution of qubit q1
- Qubit q1 acts as state dependent phase delay
Quantum versus classical computer

Most important differences between classical and quantum computer

> QC can be in a superposition of bit states
> QC exhibits interference of different circuit paths, analogous to waves or light
> QC exhibits entanglement and thus non-local effects
> QC intrinsically probabilistic
> QC more powerful for certain tasks:
  factoring, searching, quantum simulation,...

www.scottaaronson.com/blog,
smbc-comics.com

Figure by C. Addams (NYT)
How does a quantum computer look like?

Various implementation platforms are being built. Too early to tell which ones succeed.

DiVincenzo criteria for scalable quantum computer

> Well-characterized qubits, scalability to large systems
> Ability to initialize state & perform “universal” set of gate operations
> Long lifetime of quantum state >> gate operation
> Measurement capability with high fidelity

Superconducting qubits (Yale, UCSB, ETH, IBM, Google, Rigetti, Intel…)

Trapped ions (NIST, Innsbruck, IonQ, Honeywell, …)

Photonic QC (Xanadu, PsiQ, QuiX, …)

500 nm Silicon spin qubits (Princeton, New South Wales, SQC, …)

& others (neutral atoms, bosonic processors, …)
Implement quantum circuits and run using quantum cloud services

Different quantum programming frameworks are available

- IBM Qiskit, Circ (Google), PyQuil (Rigetti), Q# (Microsoft): syntax similar to Python
- Quantum Programming Studio (QPS): easy drag&drop circuits
- Many open quantum software projects: Unitary Fund, Qiskit, OpenFermion, Circ, Quest, Yao, ...
- Run quantum circuits using quantum simulators (incl. noise models) and/or on IBM hardware

From Quantum Programming Studio

From IBM Quantum Experience

Export circuits to Circ, Qiskit, PyQuil, ...
What can you do with a quantum computer?

Quantum computers promise **dramatic speedups** over classical computers for certain tasks. New computing paradigm!

Longer term applications

- Factoring integers using Shor’s algorithm: break public-key RSA encryption
- Speed-up searches of unstructured databases using Grover’s algorithm
- Simulate quantum dynamics: protein folding, molecular dynamics, chemical reactions, ...

Deutsch-Josza algorithm (example of exponential speedup):

**Task:** A black box $U_f$ performs transformation $|x\rangle|y\rangle \rightarrow |x\rangle|y \oplus f(x)\rangle$ with $x \in \{0,1\}^n$ and $f(x) \in \{0,1\}$. It is promised that $f(x)$ is either constant or balanced (= 1 for half of all $x$ and zero otherwise). Is $f(x)$ constant or balanced?
Deutsch-Josza algorithm

**Task:** A black box $U_f$ performs transformation $|x \rangle |y \rangle \rightarrow |x \rangle |y \oplus f(x) \rangle$ with $x \in \{0, 1\}^n$ and $f(x) \in \{0, 1\}$.

It is promised that $f(x)$ is either constant or balanced (= 1 for half of all $x$ and zero otherwise).

**Determine whether $f(x)$ is constant or balanced?**

We used that:

$$U_f |x \rangle (|0 \rangle - |1 \rangle) = (-1)^{f(x)} |x \rangle (|0 \rangle - |1 \rangle)$$

**Example for $n = 1$:**

$$|\psi_0 \rangle = |01 \rangle \quad \Rightarrow \quad |\psi_1 \rangle = \frac{1}{2} \left[ |0 \rangle + |1 \rangle \right] \left[ |0 \rangle - |1 \rangle \right]$$

$$|\psi_2 \rangle = \frac{1}{2} \left( (-1)^{f(0)} |0 \rangle + (-1)^{f(1)} |1 \rangle \right) \left( |0 \rangle - |1 \rangle \right) =$$

$$= \begin{cases} 
\pm \frac{1}{2} \left( |0 \rangle + |1 \rangle \right) \left( |0 \rangle - |1 \rangle \right) & \text{if } f(0) = f(1) \\
\pm \frac{1}{2} \left( |0 \rangle - |1 \rangle \right) \left( |0 \rangle - |1 \rangle \right) & \text{if } f(0) \neq f(1)
\end{cases}$$

One function call instead of two. For $n$ qubits:

- Classically need $2^{n-1} + 1$ calls
- Quantum: only one function call
- Exponential speedup
Simulating nature using quantum computers

- R. Feynman: “Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical.”

- Hilbert space dimension grows exponentially with the number of particles: $N = 2^n$

- Example: $n = 1000 \rightarrow N = 2^n = 10^{300} \gg$ number of baryons in the universe $10^{80}$

- Cannot even store wavefunction, but QC can create it!

**Idea:** Prepare wavefunction on QC using gates and measure its properties

- Find ground state energy of an interacting Hamiltonian $H$

- Algorithm: Prepare non-interacting initial state and slowly turn on interactions

$$H(t) = H_0 + tH_{\text{int}}, \quad 0 \leq t \leq 1$$

Richard Feynman
Noisy intermediate-scale quantum computing (NISQ) era

Important caveat: Current quantum computers are too noisy to allow for quantum error correction. Intermediate = 10 – 100s of qubits.

- Without error correction, errors accumulate over time and the maximal gate depth is limited

Near-term applications:
- Generate truly random numbers by sampling from a random wavefunction
- Hybrid quantum-classical algorithms using parametrized quantum circuits
  - Optimization problems
  - Optimize cost function in variational state
  - Very general!

Kandala et al (IBM) (2017)
Berthusen, PPO et al. (2022)
Quantum advantage

Quantum Advantage: perform tasks (of practical relevance) with controlled quantum systems going beyond what can be currently achieved with classical digital computers.

- Google announced quantum advantage (or supremacy) in 2019: Performed calculation on “Sycamore” chip in 200 sec that Google’s estimated would take 10’000 years on classical hardware.
- Led to classical algorithmic development that showed it can be done (potentially) much faster.
- Take-away: Google’s calculation was important proof-of-principle (similar to Wright flyer).

Most likely need quantum error correction for full quantum advantage!!!

But let’s do the research!
Quantum gold rush (before quantum winter?)

Progress in quantum technology has spurred large investments & a lot of industry activity.

- Many job opportunities
- Both hardware & software development
- Both large companies and startups
Quantum dynamics simulations

- I-Chi Chen, Benjamin Burdick, Yongxin Yao, PPO, Thomas Iadecola
  *Error-Mitigated Simulation of Quantum Many-Body Scars on Quantum Computers with Pulse-Level Control*
- Noah F. Berthusen, Thaís V. Trevisan, Thomas Iadecola, PPO
  *Quantum dynamics simulations beyond the coherence time on NISQ hardware by variational Trotter compression*
- Yong-Xin Yao, Niladri Gomes, Feng Zhang, Thomas Iadecola, Cai-Zhuang Wang, Kai-Ming Ho, PPO
  *Adaptive Variational Quantum Dynamics Simulations*
Quantum dynamics simulations

Initial state

\[ |\Psi(0)\rangle = \sum_n c_n |n\rangle \]

Energy eigenstate of many-body \( H \)

\[ |\Psi(t)\rangle = \sum_n c_n e^{-iE_n t} |n\rangle \]

Dynamics

Dynamics of an observable \( \mathcal{O} \)

\[ \langle \mathcal{O}(t) \rangle = \sum_{n,m} c_n c_m^* e^{i(E_m - E_n)t} \langle m | \mathcal{O} | n \rangle \]

Classically hard due to rapid growth of entanglement in nonequilibrium for generic \( H \)

Reason: contains highly excited states ➞ Volume-law entanglement entropy

Entanglement = complexity of classical calculation

Exponential growth of classical resources like the bond dimension in tensor networks

Opportunity for quantum computing
Overview of quantum algorithms for dynamics simulations

- Lie-Suzuki-Trotter Product formulas (PF)
  - Simple yet limited to early times for current hardware noise
  - Trotter circuit depth scales as $O(t^{1+1/k})$ fixed $t_{\text{max}}$
- Algorithms with best asymptotic scaling have significant overhead
  - Linear combination of unitaries (TS) [1], quantum walk methods [2], quantum signal processing (QSP) [3]
- Hybrid quantum-classical variational methods [5, 6]
  - Work with fixed gate depth ideally tailored for NISQ hardware
  - Trading gate depth for doing many QPU measurements

[5] Li, Benjamin, Endo, Yuan (2019); Y. Yao, PPO, T. Iadecola et al. (2021).
Overview of quantum algorithms for dynamics simulations

- Lie-Suzuki-Trotter Product formulas (PF)
  - Simple yet limited to early times for current hardware noise
  - Trotter circuit depth scales as $\mathcal{O}(t^{1+1/k})$ for fixed $t_{\text{max}}$
- Algorithms with best asymptotic scaling have significant overhead
  - Linear combination of unitaries (TS) [1], quantum walk methods [2], quantum signal processing (QSP) [3]
- Hybrid quantum-classical variational methods [5]
  - Work with fixed gate depth ideally tailored for NISQ hardware
  - Trading gate depth for doing many QPU measurements

In this talk:
- Trotter dynamics
- Variational MacLachlan approach
- Combine simplicity of Trotter product with a variational approach to simulate for long times.

Demonstrate full algorithm on IBM hardware [6].

Trotter product formula simulations of quantum dynamics

> Decompose Hamiltonian into sum of terms that include commuting operators $H = H_{\text{even}} + H_{\text{odd}}$

$$H_{\text{even}} = \frac{J}{4} \sum_{i \text{ even}} (X_i X_{i+1} + Y_i Y_{i+1} + Z_i Z_{i+1}) \quad \text{and} \quad H_{\text{odd}} = \frac{J}{4} \sum_{i \text{ odd}} (X_i X_{i+1} + Y_i Y_{i+1} + Z_i Z_{i+1})$$

> 1st order Trotter product formula

$$\left[ e^{-i(H_{\text{even}} + H_{\text{odd}}) \frac{t}{N}} \right]^N = \prod_{\alpha=1}^N \left[ e^{-iH_{\text{even}} \frac{t}{N}} e^{-iH_{\text{odd}} \frac{t}{N}} + \mathcal{O}(t^2/N^2) \right]$$

Trotter step size

$$\tau = t/N$$

Must be chosen small $\gg$ deep circuits

Can be easily implemented as product of two-qubit unitaries

While product formulas are straightforward to implement, they result in deep circuits for long and precise simulations

Lloyd (1996)
NISQ Trotter simulations of mixed field Ising model

> Benchmark Trotter simulations on current NISQ hardware

Mixed-field Ising model:

\[
H = \frac{V}{4} \sum_{i=1}^{L-1} Z_i Z_{i+1} + \frac{V}{2} \sum_{i=2}^{L-1} Z_i + \frac{V}{4} (Z_1 + Z_L) + \Omega \sum_{i=1}^{L} X_i
\]

> Naïve Trotter simulation limited to \( t \approx 1/J \) due to finite coherence time on device

Bernien, Lukin (2017)

Displays many-body coherent dynamics for \( V \gg \Omega \)

\( V = 2, \, \Omega = 0.48 \)

One step of Trotter circuit in \( L=5 \) system, starting from Neel state.

Use pulse level control and error mitigation strategies to extend simulation time

Trotter simulation on IBM Nairobi QPU
Pulse level control and error mitigation

- Pulse level control allows to make optimal use of finite coherence time on device
  - Direct implementation of $R_{zz}$ gate via cross-resonance pulse $\Rightarrow$ shortens program by about half

- Error mitigation is key to extend final time of simulation
  - Zero-noise extrapolation (Mitiq) + Pauli twirling: $G \mapsto GG^\dagger G$.
  - Readout error mitigation (tensor product assumption):
    $$C_{\text{ideal}} = M^{-1}C_{\text{noisy}}.$$  
    $$M = \begin{bmatrix} 1 - \epsilon_1 & \eta_1 \\ \epsilon_1 & 1 - \eta_1 \end{bmatrix} \otimes \cdots$$
  - Symmetry-based postselection (tailored to specific model)
  - Dynamical decoupling: apply $X(\pi)$ and $X(-\pi)$ during qubit idle time

Kraus form

\[ E_h = \sum_{a=0}^3 \sum_{b=0}^3 \alpha_{h,a,b} \sigma_c^a \sigma_t^b \]

\[ \tilde{N}_\Lambda = F_\Lambda[1] + \sum_{(a,b)\neq(0,0)} \epsilon_{a,b} [\sigma_c^a \sigma_t^b] \]

Pauli twirling converts noise to stochastic form $\Rightarrow$ justification for ZNE

Li, Benjamin (2017)
Pulse level control and error mitigation

- Simulation of 12 qubits on IBM Guadalupe
- Comparison of pulse gate versus standard CNOT realization of Rzz
- Full error mitigation techniques for both

Pulse and zero-noise extrapolation (ZNE) are effective strategies to reduce errors.

\[ V = 0.9, \Omega = 0.6 \]
ZNE used linear extrapolation and scale factors \{1, 1.5, 2\}.
Results for 12 qubits on IBM Guadelupe

- Simulation of 12 qubits on IBM Guadelupe
- Comparison of pulse gate versus standard CNOT realization of Rzz
- Full error mitigation techniques for both
- Qubits have different quality
  - Compare i=1,2 with i=6 for example
  - Gate noise
  - Decoherence times

Custom pulse gate for $R_{zz}$ shows advantage proportional to shortening of pulse sequence $\mathcal{O}$ Trotter simulations limited to early times

$V = 2, \Omega = 0.48$
Quantum dynamics simulations

- I-Chi Chen, Benjamin Burdick, Yongxin Yao, PPO, Thomas Iadecola
  *Error-Mitigated Simulation of Quantum Many-Body Scars on Quantum Computers with Pulse-Level Control*

- Noah F. Berthusen, Thaís V. Trevisan, Thomas Iadecola, PPO
  *Quantum dynamics simulations beyond the coherence time on NISQ hardware by variational Trotter compression*

- Yong-Xin Yao, Niladri Gomes, Feng Zhang, Thomas Iadecola, Cai-Zhuang Wang, Kai-Ming Ho, PPO
  *Adaptive Variational Quantum Dynamics Simulations*
Time-dependent variational quantum algorithms

Variational form of quantum state

\[ |\Psi[\theta]\rangle = \prod_{\mu=0}^{N_\theta-1} e^{-i\theta_\mu A_\mu} |\Psi_0\rangle. \]

Von Neumann equation

\[ \frac{d\rho}{dt} = \mathcal{L}(\rho) = -i[\mathcal{H}, \rho] \]

MacLachlan distance b/w exact and variational time evolution

\[ L^2 \equiv \left\| \sum_\mu \frac{\partial \rho[\theta]}{\partial \theta_\mu} \dot{\theta}_\mu - \mathcal{L}[\rho] \right\|^2 = \sum_{\mu\nu} M_{\mu\nu} \dot{\theta}_\mu \dot{\theta}_\nu - 2 \sum_\mu V_\mu \dot{\theta}_\mu + \text{Tr}[\mathcal{L}[\rho]^2]. \]

Minimize \( L^2 \)

EOM for variational parameters

\[ \sum_\nu M_{\mu\nu} \dot{\theta}_\nu = V_\mu. \]

Matrix \( M_{\mu\nu} \) and vector \( V_\mu \) measured on QPU

Only so good as the ansatz can follow the dynamics

☞ How to select an efficient yet flexible variational ansatz?

M measures state change under parameter change

V depends on Hamiltonian

[1] Li, Benjamin, Endo, Yuan (2019).
Adaptive ansatz construction in pseudo-Trotter form: flexible and avoids pitfalls of fixed ansatz

Add operator from predefined pool to ansatz if MacLachlan distance increases above set threshold

Operator pool we use contains all Pauli strings that appear in Hamiltonian

[1] Y. Yao et al., PRX Quantum 2, 030307 (2021)
Application I: continuous quench in integrable spin chain

- Linear quench of anisotropic XY chain in transverse magnetic field
  \[ \hat{H} = -J \sum_{i=0}^{N-2} \left( (1 + \gamma) \hat{X}_i \hat{X}_{i+1} + (1 - \gamma) \hat{Y}_i \hat{Y}_{i+1} \right) + h_z \sum_{i=0}^{N-1} \hat{Z}_i \text{ with } \gamma(t) = 1 - \frac{2t}{T} \]

- AVQDS follows exact solution during and after quench, shown for \( N = 8 \)

- Circuit depth saturates at 100 CNOTs << Trotter circuit depth \( 10^4 \) CNOTs

- Can simulate system with gate depth independent of time \( t \) \( \Rightarrow \) can simulate to arbitrary times!
Application II: sudden quench in nonintegrable spin chain

- Sudden quench in mixed-field Ising model
  
  \[ \hat{H} = -J \sum_{i=0}^{N-1} \hat{Z}_i \hat{Z}_{i+1} + \sum_{i=0}^{N-1} \left( h_x \hat{X}_i + h_z \hat{Z}_i \right) \]

  Loschmidt echo: \[ L(t) = \left| \langle \Psi_0 | e^{-i\hat{H}t} | \Psi_0 \rangle \right|^2 \]

  Initial state: \[ |\Psi_0\rangle = |\uparrow \cdots \uparrow\rangle \]

- Circuit depth two orders of magnitude smaller than Trotter circuit depth

- Saturated AVQDS circuit depth scales exponentially with system size \( N \)

- # measurements is main bottleneck of algorithm \( \propto N^2 \)
Quantum dynamics simulations

• I-Chi Chen, Benjamin Burdick, Yongxin Yao, PPO, Thomas Iadecola
  *Error-Mitigated Simulation of Quantum Many-Body Scars on Quantum Computers with Pulse-Level Control*

• Noah F. Berthusen, Thaís V. Trevisan, Thomas Iadecola, PPO
  *Quantum dynamics simulations beyond the coherence time on NISQ hardware by variational Trotter compression*

• Yong-Xin Yao, Niladri Gomes, Feng Zhang, Thomas Iadecola, Cai-Zhuang Wang, Kai-Ming Ho, PPO
  *Adaptive Variational Quantum Dynamics Simulations*
Variational Trotter Compression (VTC) algorithm

Key idea of VTC algorithm [1, 2]:

> First, propagate state using Trotter: \( |\psi(\theta_t)\rangle \xrightarrow{\text{U_{trot}(\tau)}} U_{trot}(\tau) |\psi(\theta_t)\rangle \)

> Then, update variational parameters \( \theta_t \rightarrow \theta_{t+\tau} \) by optimizing fidelity cost function

Fidelity cost function

\[
C = \left| \langle \psi_0 | U^\dagger(\theta_{t+\tau}) U_{trot}(\tau) U(\theta_t) |\psi_0\rangle \right|^2
\]

Our variational state:

\[
|\psi(\theta)\rangle = U(\theta) |\psi_0\rangle = \prod_{l=1}^{\ell} \prod_{i=1}^{N} e^{-i\theta_{t,i} A_i} |\psi_0\rangle
\]

\( \ell = \) number of layers
\( N = \) number of parameters per layer
\( A_i = \) Hermitian operator (e.g. Pauli matrix)

Return probability to initial state is maximal for optimal parameters \( \theta_{t+\tau} \)

Measure cost function on QPU [3]

Application to Heisenberg model: choice of ansatz

1D AF Heisenberg model

\[ H_0 = \frac{J}{4} \sum_{i=1}^{M} (X_i X_{i+1} + Y_i Y_{i+1} + Z_i Z_{i+1}) \]

Start from classical Néel state and time-evolve with \( H_0 \):

\[ |\psi(t)\rangle = e^{-iH_0 t} |010101 \cdots \rangle \]

\[ |\psi(\mathcal{O}(\ell))\rangle = \prod_{l=1}^{\ell} U_{\text{even}}(\phi_l) U_{\text{odd}}(\theta_l) |\psi_0\rangle \]

\[ U_{\text{odd}}(\theta_l) = \prod_{j \text{ odd}} e^{-i \phi_{l,j} (X_j X_{j+1} + Y_j Y_{j+1} + Z_j Z_{j+1})} \]

\[ U_{\text{even}}(\phi_l) = \prod_{j \text{ even}} e^{-i \phi_{l,j} (X_j X_{j+1} + Y_j Y_{j+1} + Z_j Z_{j+1})} \]

Determine depth of layered ansatz \( \ell \equiv \ell^* \) to accurately describe \( |\psi(t)\rangle \)

Brickwall form of quantum circuit

Department of Physics and Astronomy
Required layer numbers versus time

- Start from classical Néel state and time-evolve with $H_0: |\psi(t)\rangle = e^{-iH_0 t} |010101 \cdots \rangle$
- Determine depth of layered ansatz $\ell$ to accurately describe $|\psi(t)\rangle$

Variational form

$$|\psi(\mathcal{G}(\ell))\rangle = \prod_{i=1}^{\ell} U_{\text{even}}(\phi_i) U_{\text{odd}}(\theta_i) |\psi_0\rangle$$

Overlap with exact state

$$1 - \mathcal{F}(t, \mathcal{G}(\ell)) = 1 - |\langle \psi(\mathcal{G}(\ell)) | \psi(t) \rangle|^2$$

Required layer number $\ell$ to achieve $1 - \mathcal{F} < 10^{-4}$ grows linearity with time and then saturates.
VTC benchmark on statevector simulator

\[ M = 11 \]
\[ l^* = 76 \]

\[ \mathcal{F}(t, \hat{\theta}_t) = \left| \langle \psi(\hat{\theta}_t) | \psi(t) \rangle \right|^2 \]

Best Compression \( \equiv \left| \langle \psi(t) | U_{\text{Trot}}(n = \ell) \psi(\hat{\theta}_{t-\tau}) \rangle \right|^2 \)

VTC overlap \( \equiv \left| \langle \psi(\hat{\theta}_t) | U_{\text{Trot}} \psi(\hat{\theta}_{t-\tau}) \rangle \right|^2 \)

- VTC allows simulating to arbitrarily long times with high fidelity.

Parameters: \( \ell = 76, n = 76 \)
\( \tau = 15.2/J, \Delta t = 0.2/J \)
VTC on ideal circuit simulators

- Double-time contour cost function circuit
- Non-gradient-based optimizer: CMA-ES
- Larger shot numbers increase fidelity
- Single compression step takes few hours

VTC is feasible for noisy cost function.

\[
F(t, \hat{\theta}_t) = |\langle \psi(\hat{\theta}_t) | \psi(t) \rangle |^2
\]

\(M = 6\)
Cost function evaluation on IBM hardware:
- `ibmq_santiago` & `ibmq_quito`
- Final fidelity = 0.96, where Trotter fidelity has decayed to < 0.4 already
- 15 compression steps
- Average fidelity $\langle F \rangle = 0.86$
- $\mathcal{M} = 5700$ measurement circuits in total
- Comparable number of measurements for MacLachlan simulations $\approx 10^4$
Summary

> Trotter dynamics simulations with pulse-level control
  > Straightforward to implement even for large systems
  > Error mitigation and pulse-level control boost performance
> Adaptive variational quantum dynamics simulation (AVQDS) framework
  > Orders of magnitude shallower circuits than Trotter simulations
> Explicit demonstration of dynamics simulation beyond QPU coherence time
  > Variational Trotter Compression on IBM hardware for $M = 3$ sites
  > Can simulate out to arbitrary long times with high fidelity

References:
• I.-C. Chen et al., arXiv:2203.08291 (2022)
• Y. Yao et al., PRX Quantum 2, 030307 (2021)