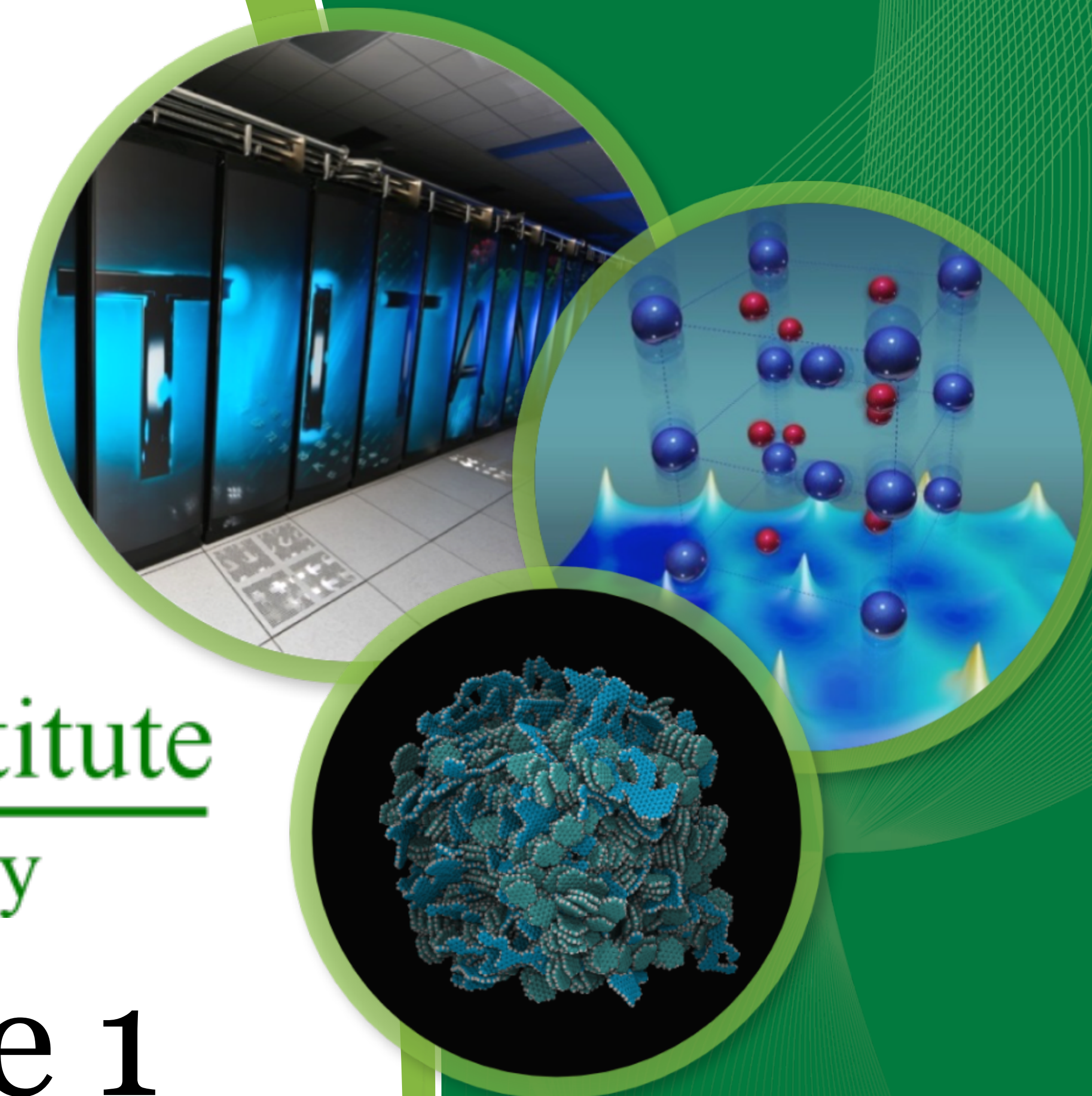
Cloud Quantum Computing of an Atomic Nucleus

Quantum Computing Institute
Oak Ridge National Laboratory



CIPANP18 - June 1

Eugene Dumitrescu

Collaborators: A. McCaskey, T. Papenbrock, G. Hagen, D. Dean, T. Morris, M. Savage, N. Klco, M.S. Chen, R. Pooser, P. Lougovski

- This talk: PRL 120.210501
- Kclo, et. al. arXiv:1803.03326
- XACC: arXiv:1710.01794
- github.com/eclipse/xacc

ORNL is managed by UT-Battelle for the US Department of Energy

Work supported by the DOE Quantum Testbed Pathfinder and ORNL LDRD projects.

OAK RIDGE National Laboratory

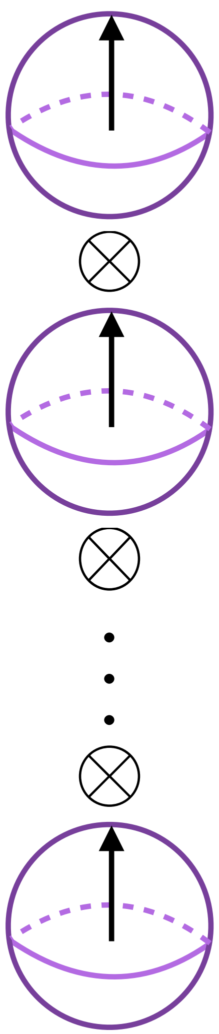
Outline

1. Quantum computer as a many-body simulator
2. Near term computational models
3. A nuclear quantum computation (i.e. reality)
4. XACC programming framework
5. Prospects for scalable QC

Elements of a Quantum Computation

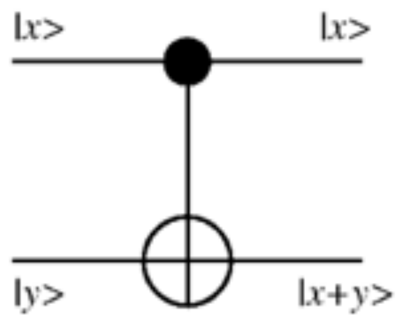
Elements of a Quantum Computation

Initialization

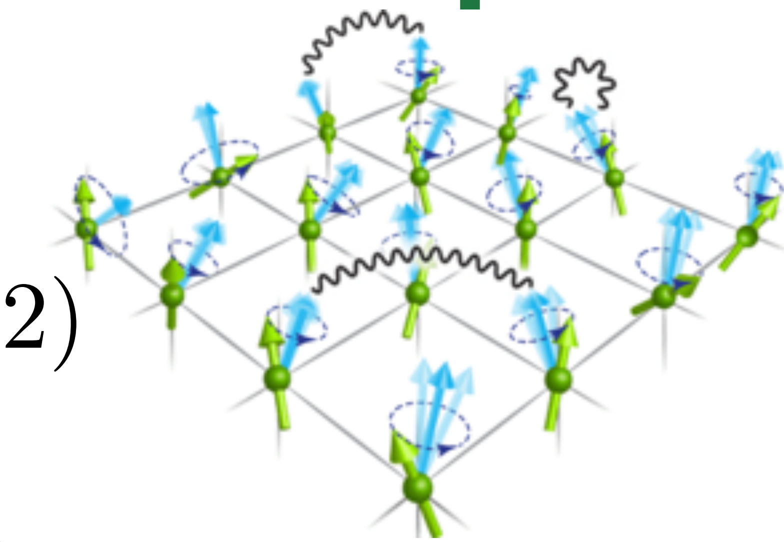


Elements of a Quantum Computation

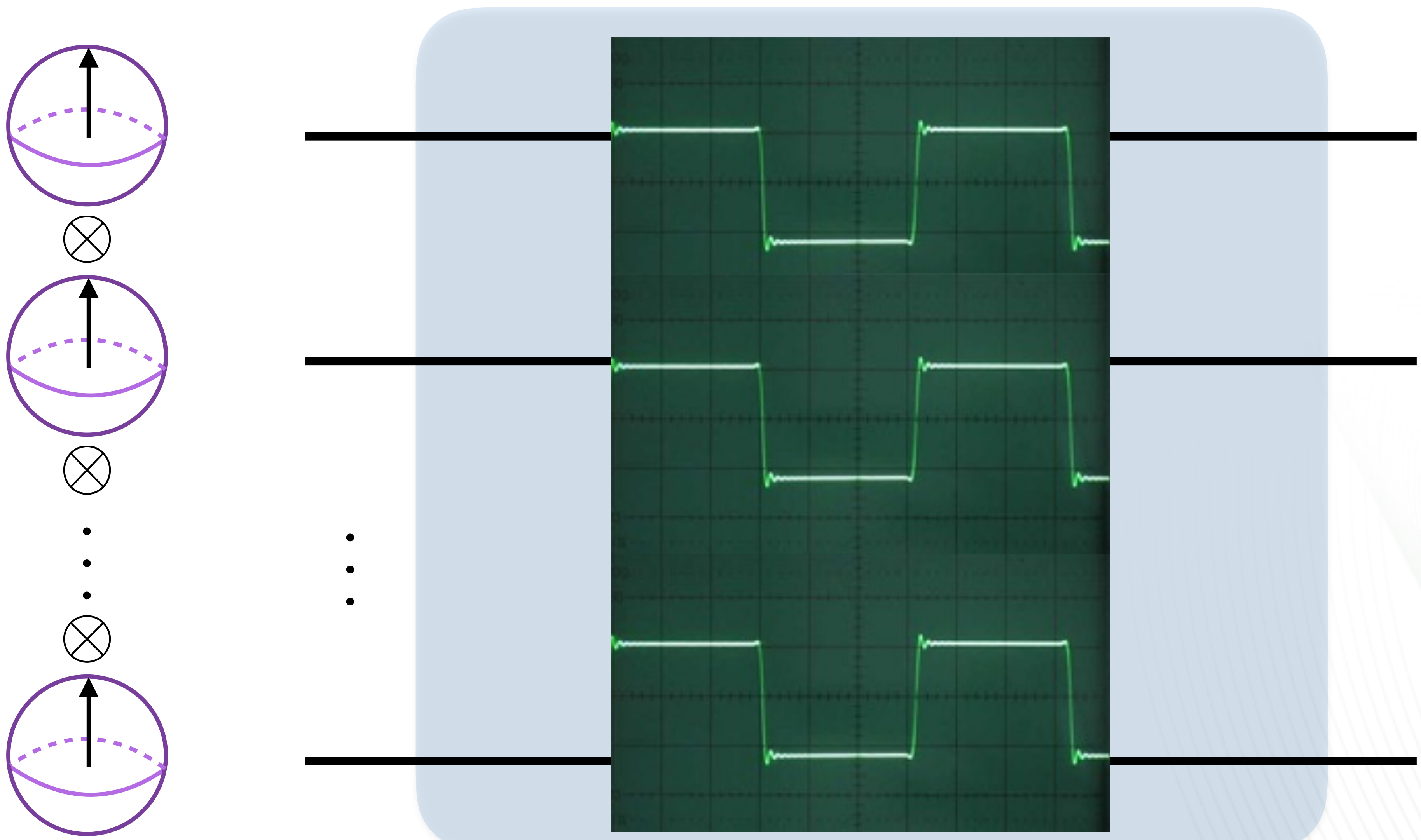
Initialization



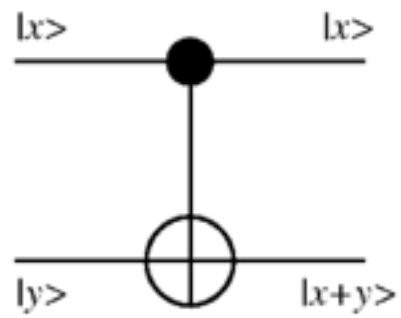
$+\ SU(2)$



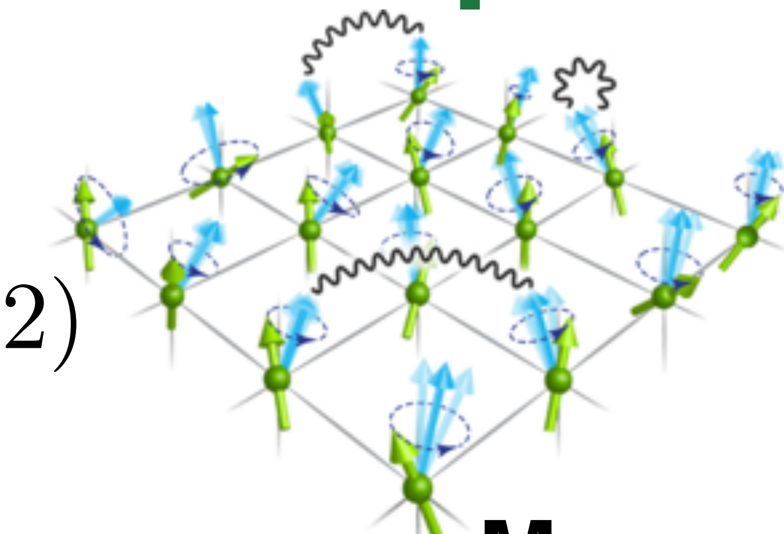
Evolution



Elements of a Quantum Computation



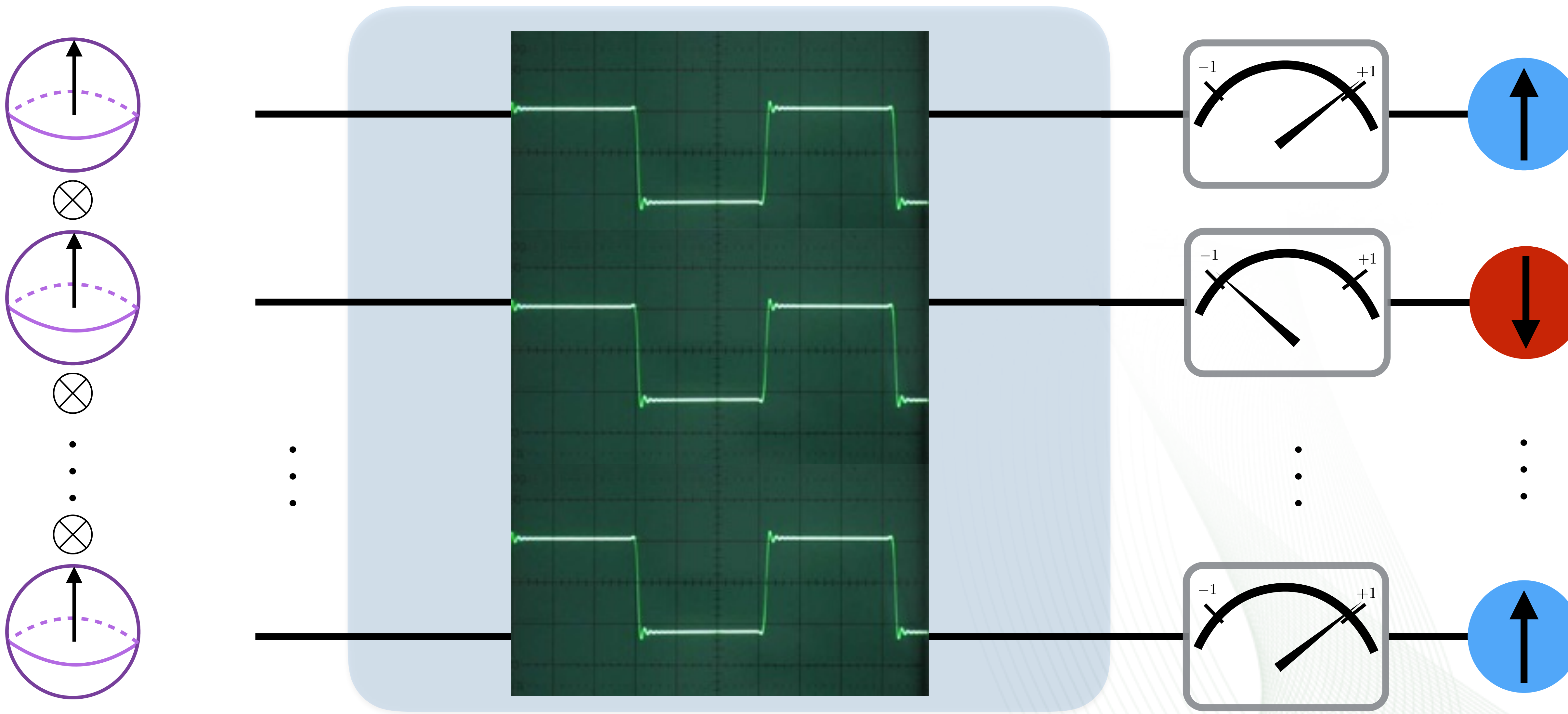
$+ \ SU(2)$



Initialization

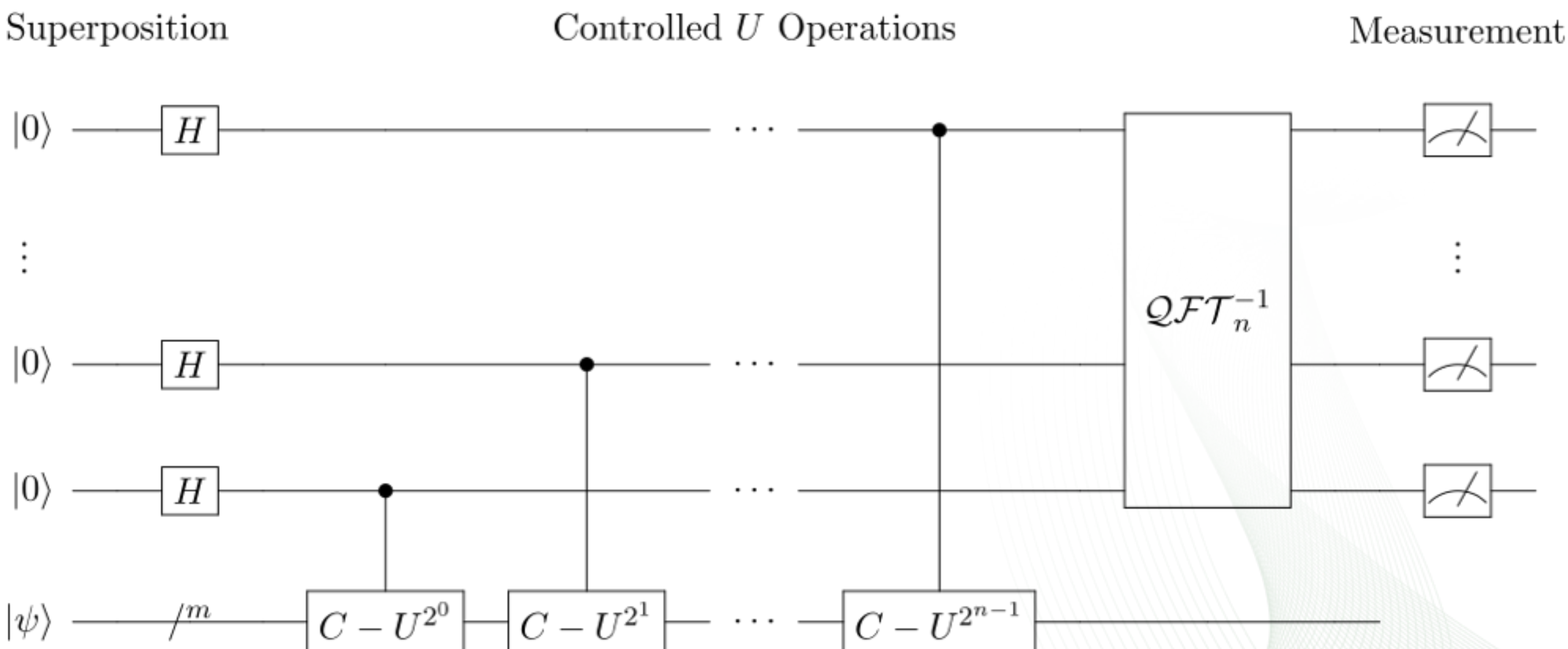
Evolution

Measurement

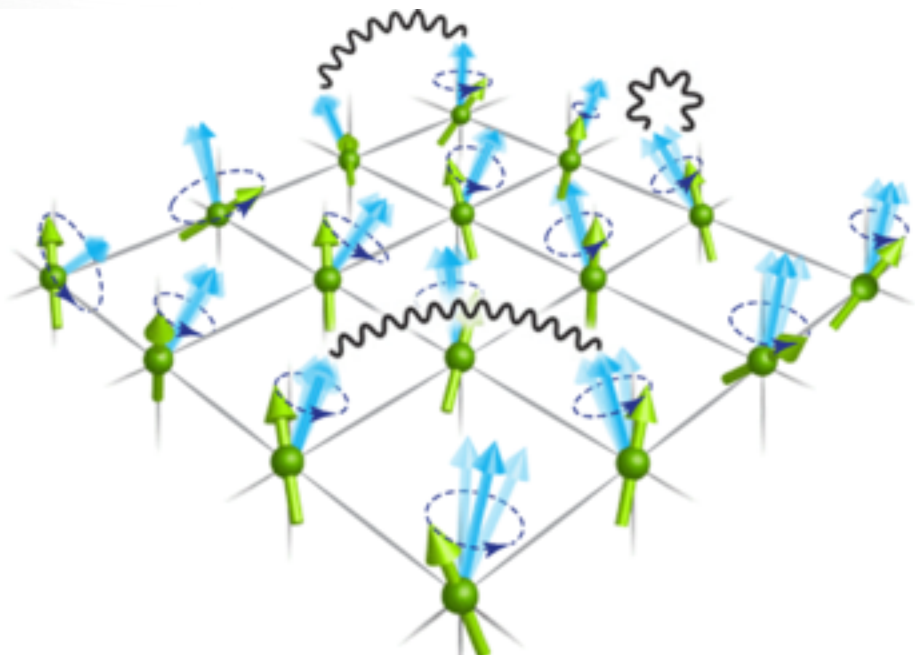


Textbook Quantum Simulation

- Use (universal) quantum simulator capable of applying time evolution on simulator system.
- Lloyd (96,97) Decompose (Trotterize) local time evolution and resolve eigenspectrum using quantum phase estimation algorithm

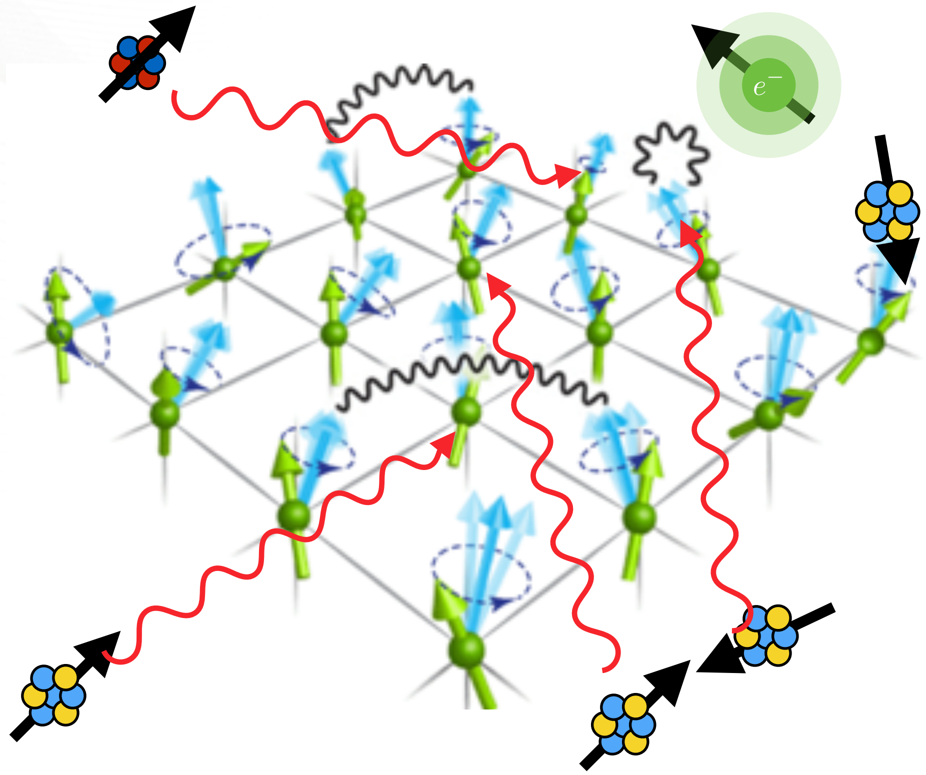


Environmental problems



$$i\hbar\partial_t\psi(t) = \hat{H}\psi(t)$$

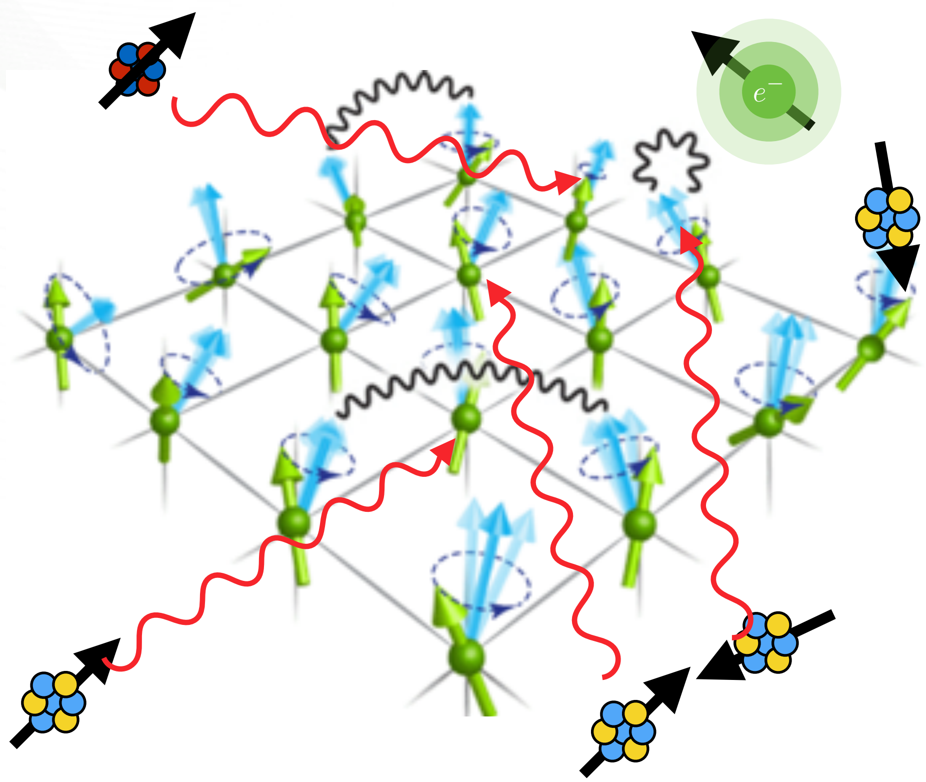
Environmental problems



$$i\hbar\partial_t\psi(t) = \hat{H}\psi(t)$$

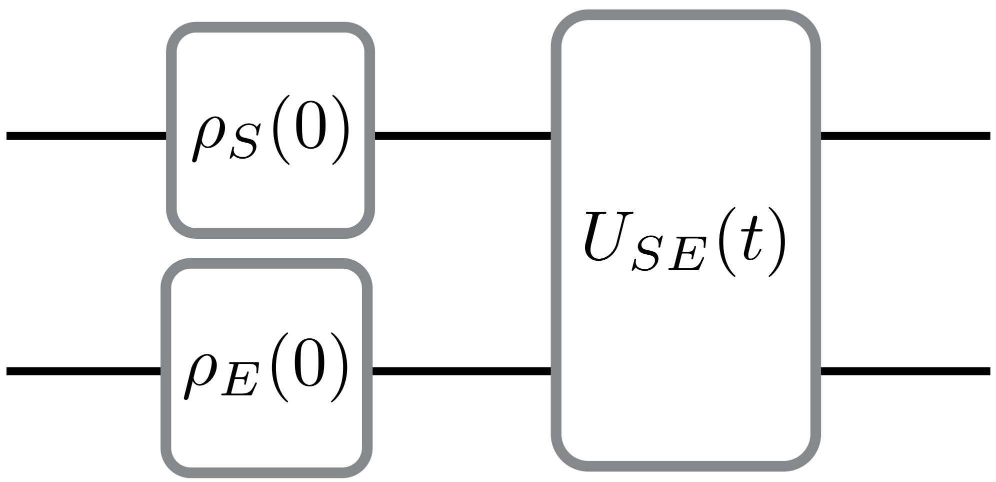
$$\hat{H} = \hat{H}_{sys} + \hat{H}_{env} + \hat{H}_{sys-env} + \hat{H}_{cont} + \hat{H}_{sys-cont} + \hat{H}_{cont-env}$$

Environmental problems

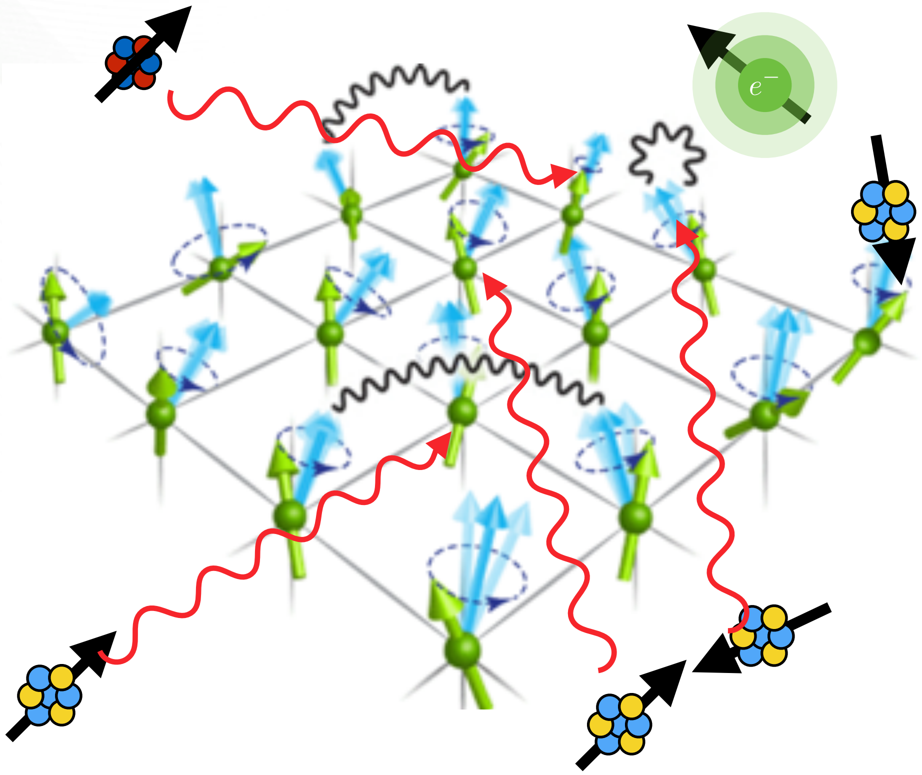


$$i\hbar\partial_t\psi(t) = \hat{H}\psi(t)$$

$$\hat{H} = \hat{H}_{sys}$$
$$+\hat{H}_{env} + \hat{H}_{sys-env}$$
$$+\hat{H}_{cont} + \hat{H}_{sys-cont}$$
$$+\hat{H}_{cont-env}$$

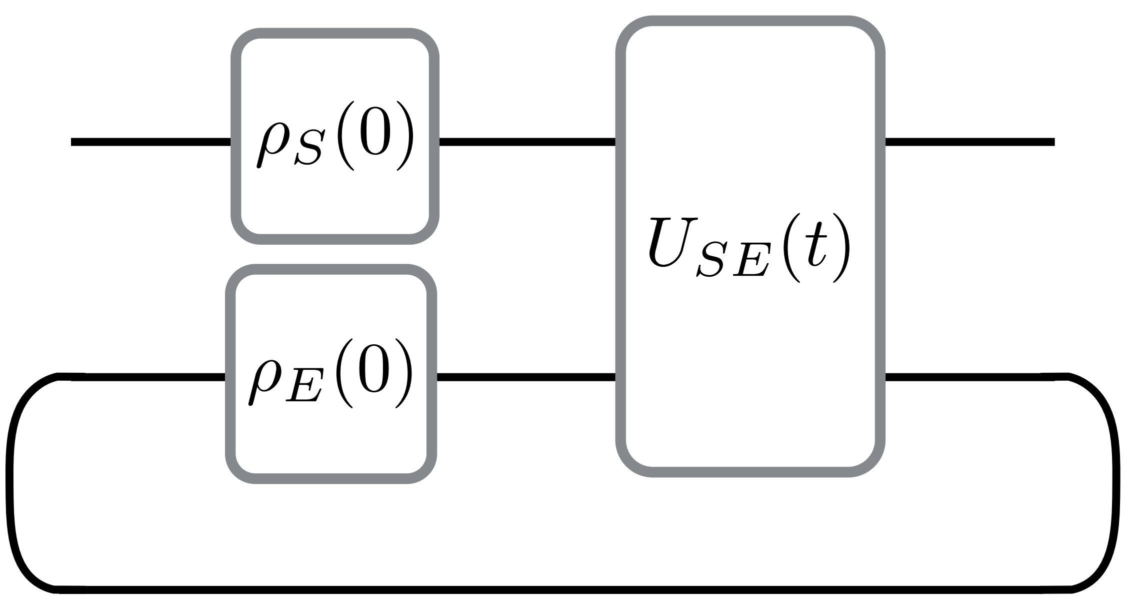


Environmental problems



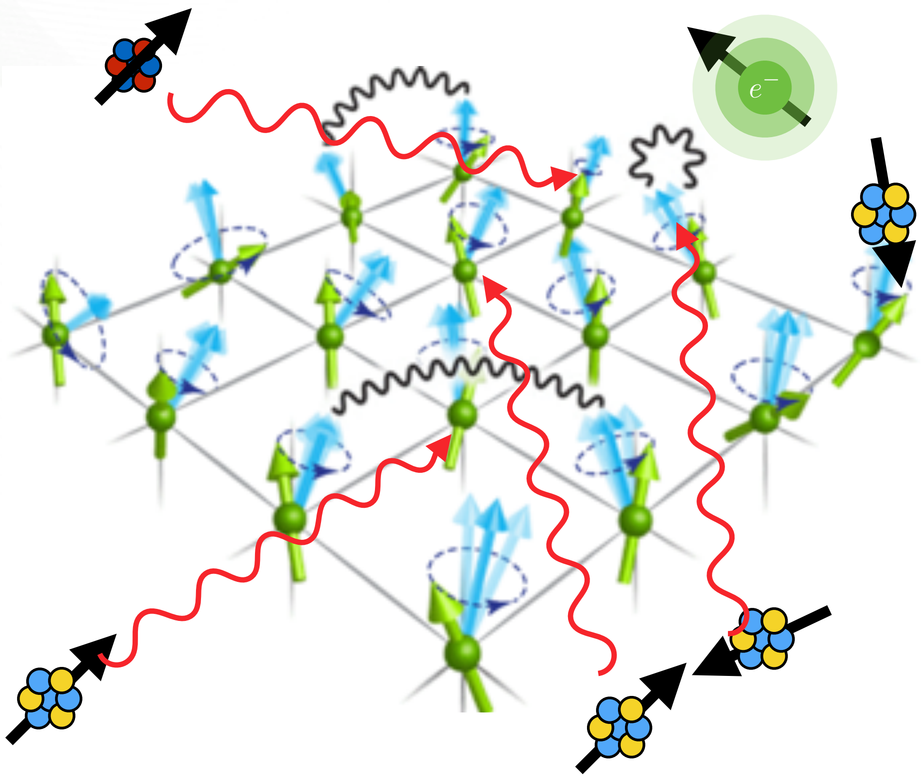
$$i\hbar\partial_t\psi(t) = \hat{H}\psi(t)$$

$$\begin{aligned}\hat{H} =& \hat{H}_{sys} \\ &+\hat{H}_{env} + \hat{H}_{sys-env} \\ &+\hat{H}_{cont} + \hat{H}_{sys-cont} \\ &+\hat{H}_{cont-env}\end{aligned}$$



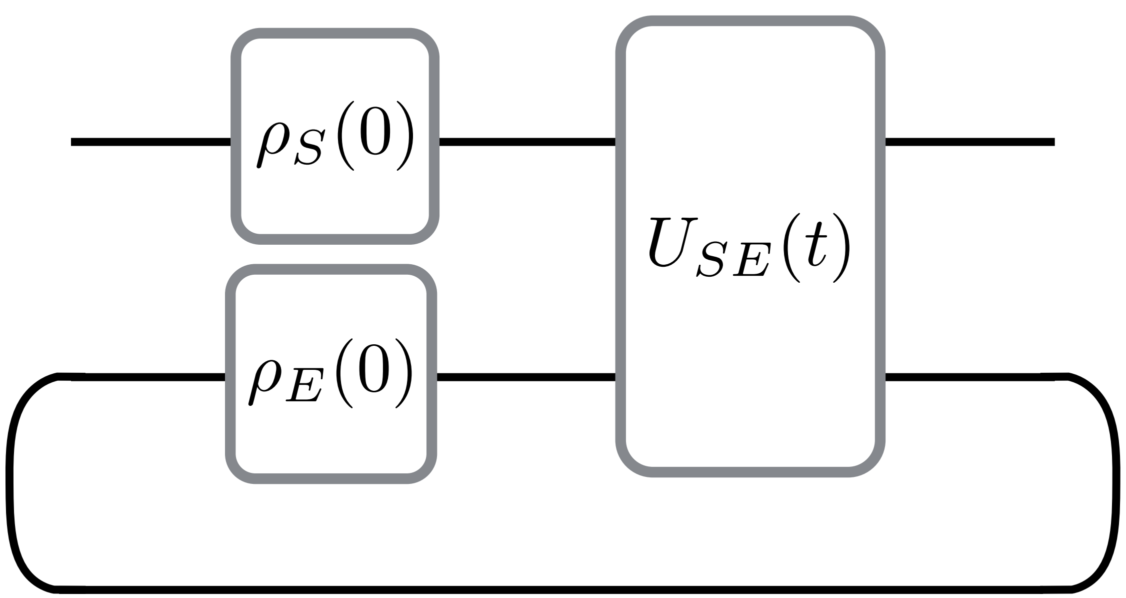
$$\rho_S(t) = \mathrm{Tr}_E\{\rho_{SE}(t)\}$$

Environmental problems



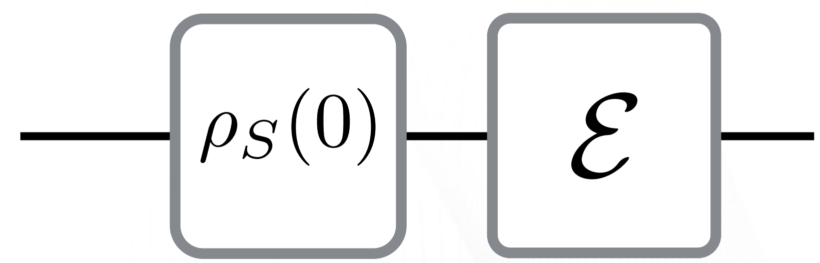
$$i\hbar\partial_t\psi(t) = \hat{H}\psi(t)$$

$$\hat{H} = \hat{H}_{sys} + \hat{H}_{env} + \hat{H}_{sys-env} + \hat{H}_{cont} + \hat{H}_{sys-cont} + \hat{H}_{cont-env}$$



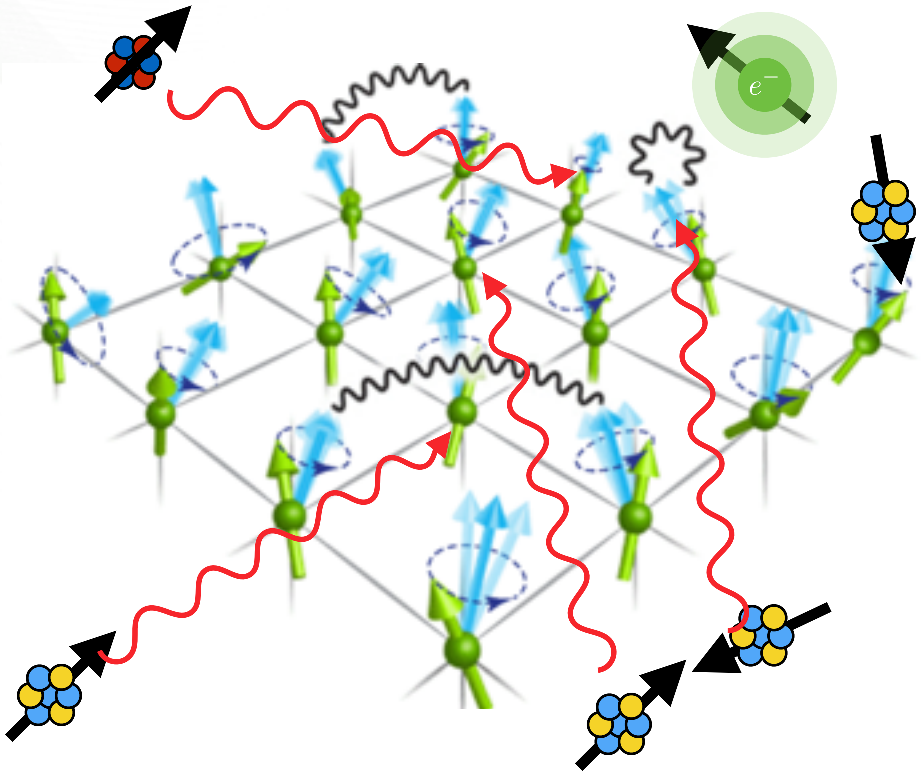
=

$$\mathcal{E} : \rho_i \mapsto \rho_f$$



$$\rho_S(t) = \mathrm{Tr}_E\{\rho_{SE}(t)\}$$

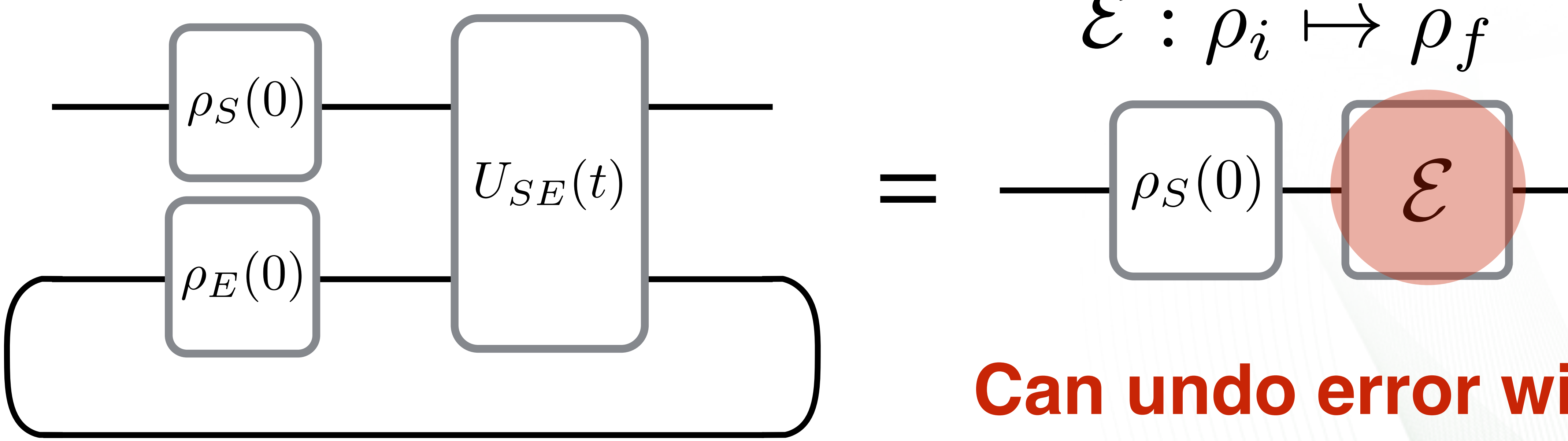
Environmental problems



$$i\hbar\partial_t\psi(t) = \hat{H}\psi(t)$$

$$\hat{H} = \hat{H}_{sys}$$
$$+\hat{H}_{env} + \hat{H}_{sys-env}$$
$$+\hat{H}_{cont} + \hat{H}_{sys-cont}$$
$$+\hat{H}_{cont-env}$$

$$\mathcal{E} : \rho_i \mapsto \rho_f$$



$$\rho_S(t) = \mathrm{Tr}_E\{\rho_{SE}(t)\}$$

Can undo error with quantum error correction

Fault Tolerant Quantum Computation

- Large scale quantum error correction (Shor, CSS, Kitaev, etc.) needed with scale depending on logical error rate
- Fault tolerant QC enabled by intrinsically topological qubits (i.e. cond-matt Majoranas)
- Small quantum error correcting codes (Blatt - steane encoding)
- Hardware dependent, i.e. channel specific, optimized variational error corrected channels (~ML approach)

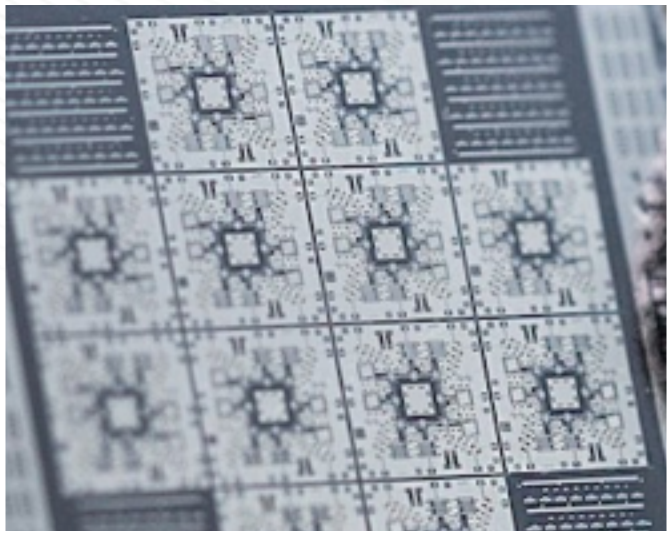
Babbush, et. al. 2018

problem		physical qubits		execution time (hours	
System	Spin-Orbitals (N)	$p = 10^{-3}$	$p = 10^{-4}$	$p = 10^{-3}$	$p = 10^{-4}$
Hubbard model	72	1.4×10^6	4.4×10^5	4.6	2.6
Hubbard model	128	2.1×10^6	6.6×10^5	15	8.4
Hubbard model	200	3.2×10^6	8.9×10^5	40	21
Hubbard model	800	1.4×10^7	3.6×10^6	6.7×10^2	3.7×10^2
Electronic structure	54	1.4×10^6	3.9×10^5	0.82	0.43
Electronic structure	128	2.4×10^6	8.1×10^5	9.9	5.6
Electronic structure	250	4.4×10^6	1.2×10^6	58	30
Electronic structure	1024	2.0×10^7	4.8×10^6	2.7×10^3	1.4×10^3

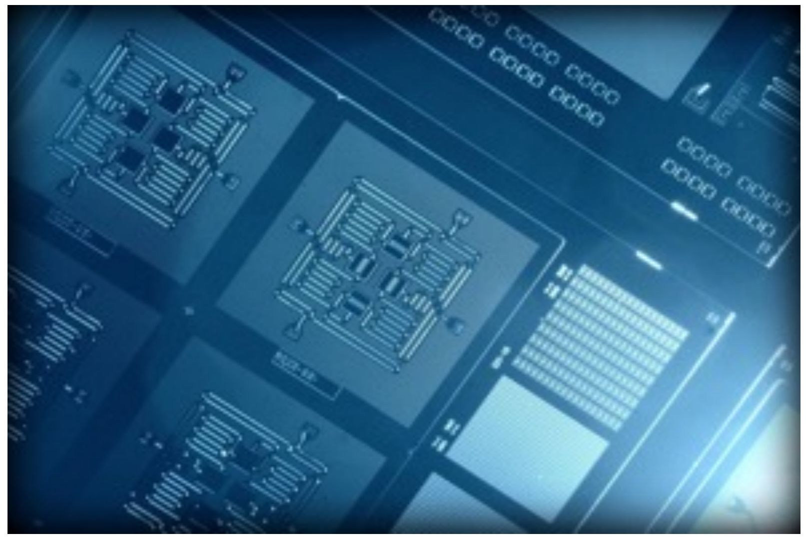
Near Term Prospects

Near Term Prospects

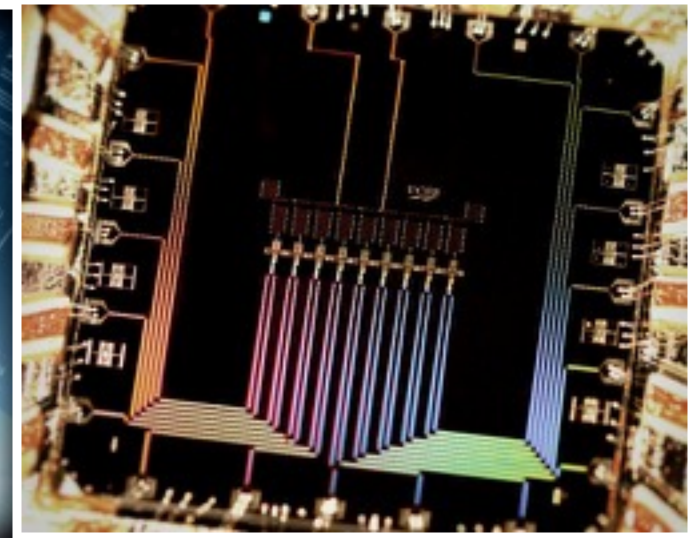
Rigetti



IBM

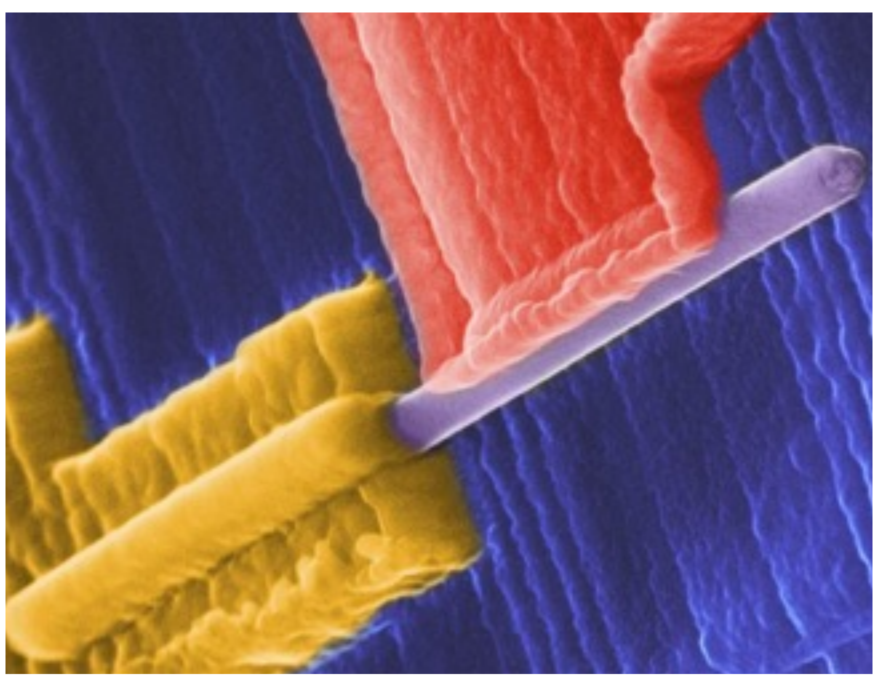


Google

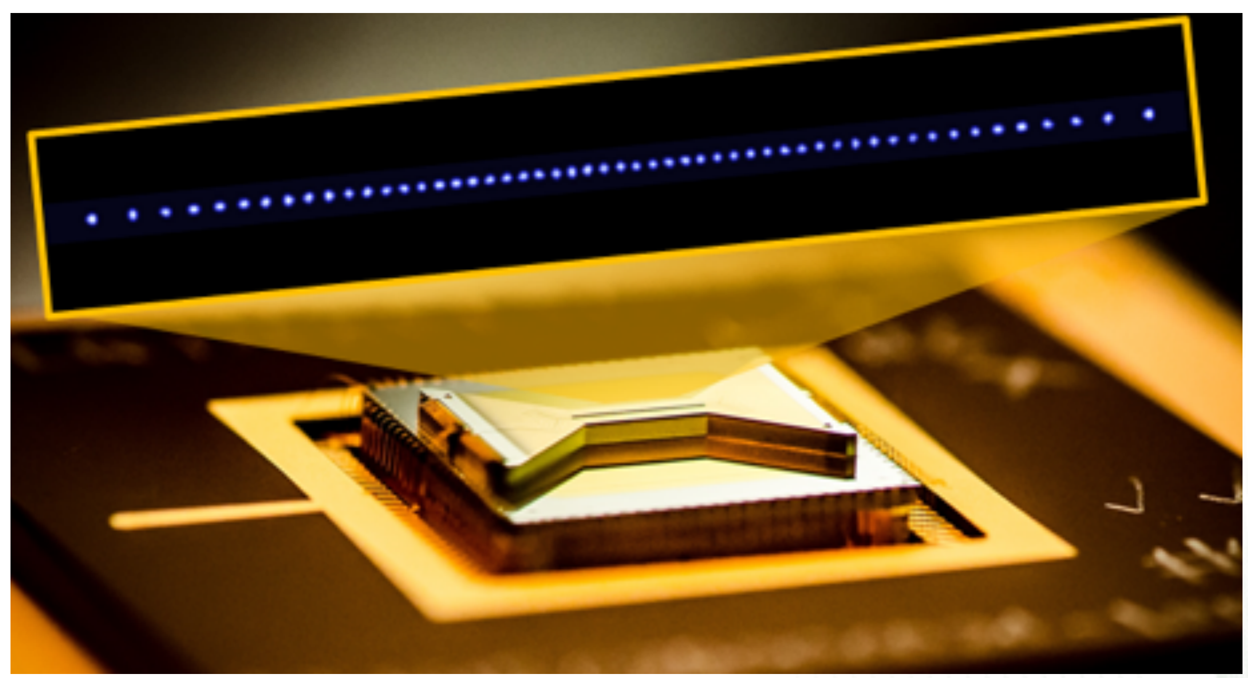


Today:
~ 50 qubits
~ 50 gates

Delft, NL

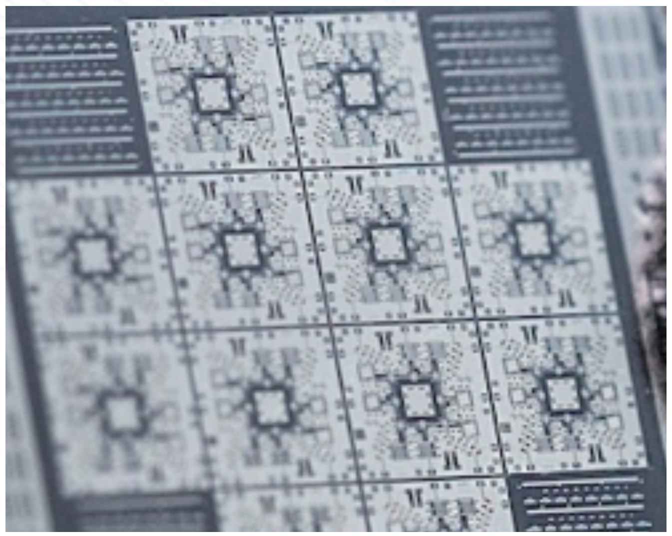


Ions: Monroe, Blatt, etc

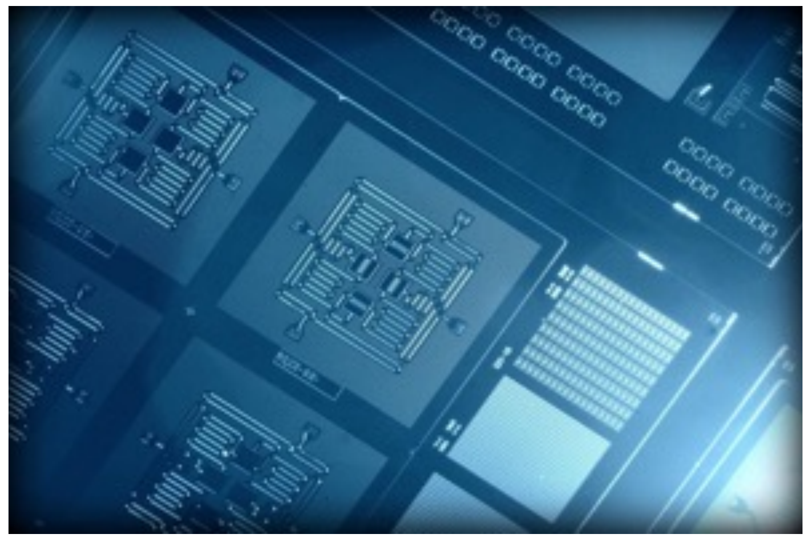


Near Term Prospects

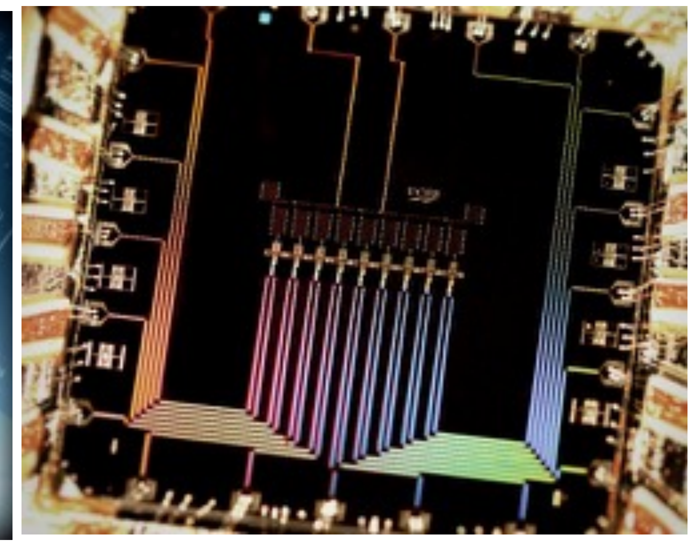
Rigetti



IBM

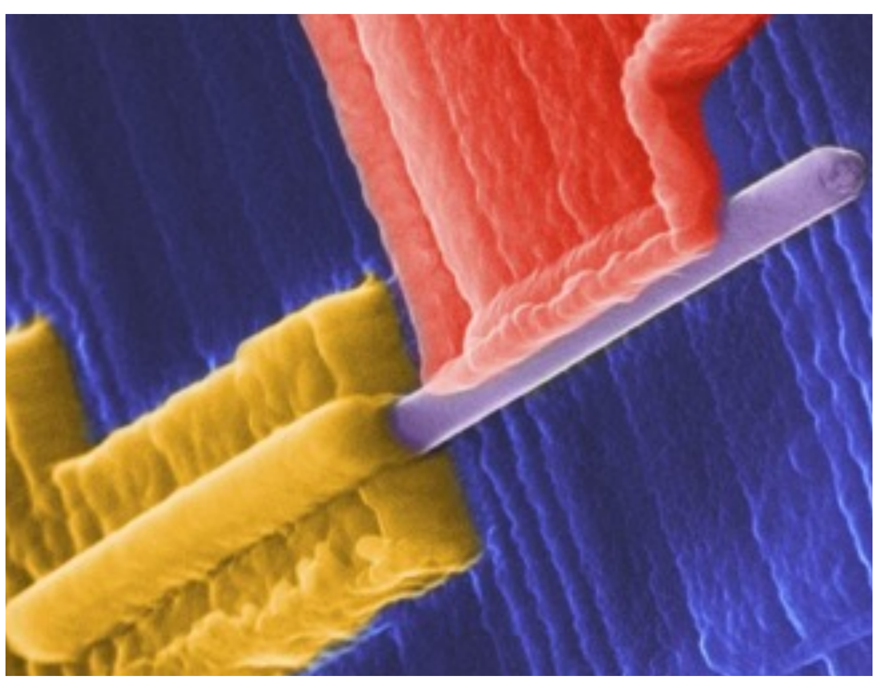


Google

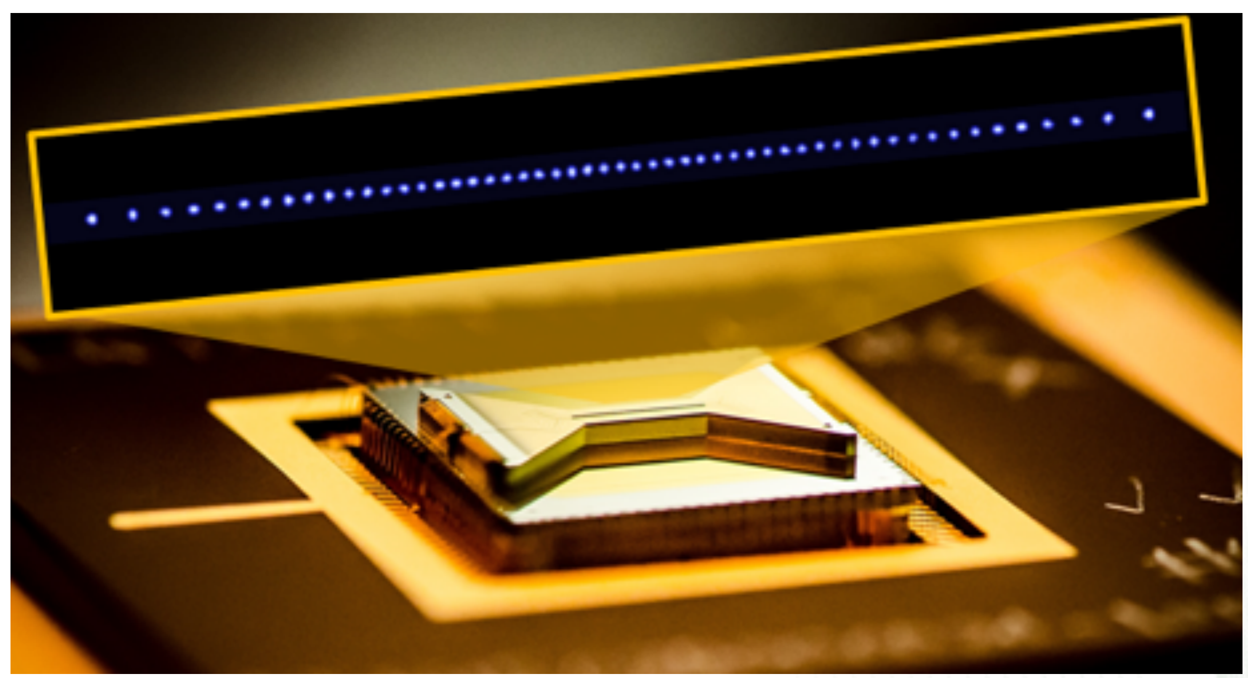


Today:
~ 50 qubits
~ 50 gates

Delft, NL



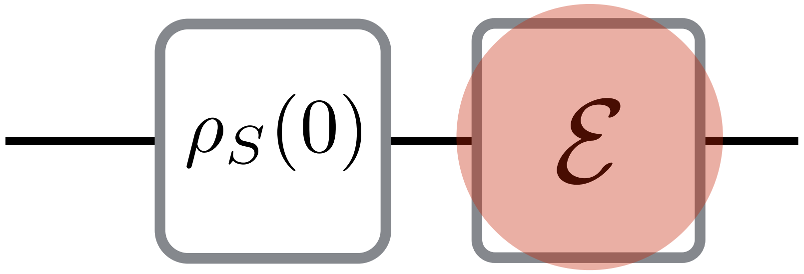
Ions: Monroe, Blatt, etc



Future:
~ 500 qubits
~ 500 gates

Noisy intermediate-scale quantum (NISQ) era

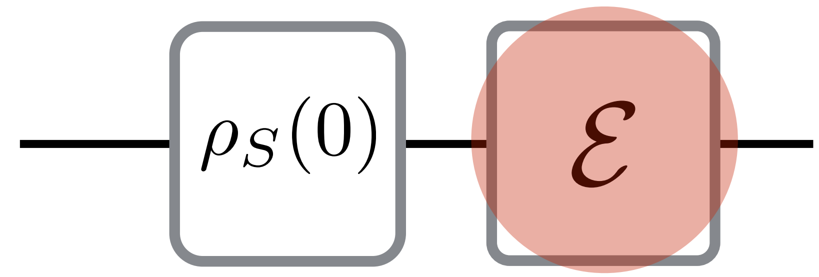
Low-depth computational approaches



$$T_c \ll T_1, T_2$$

Low-depth computational approaches

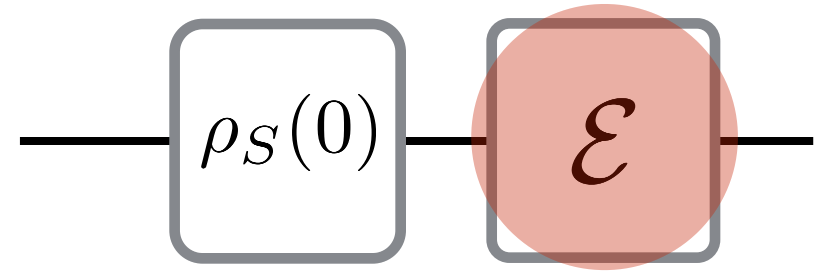
- Adiabatic (Analog) — D-WAVE, quantum simulators, etc.



$$T_c \ll T_1, T_2$$

Low-depth computational approaches

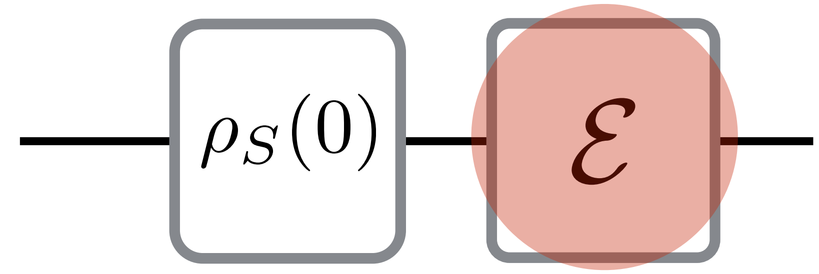
- Adiabatic (Analog) — D-WAVE, quantum simulators, etc.
- Variational Quantum Eigensolver (VQE) *Peruzzo, McClean, Nat Comm. 2014*:



$$T_c \ll T_1, T_2$$

Low-depth computational approaches

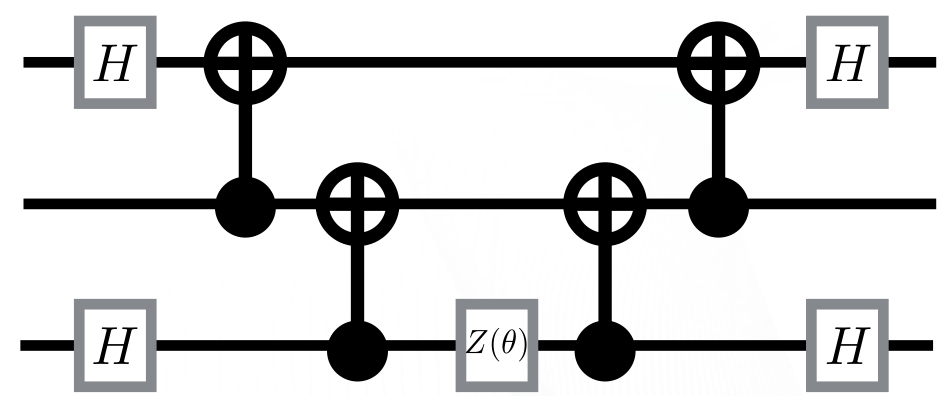
- Adiabatic (Analog) — D-WAVE, quantum simulators, etc.
- Variational Quantum Eigensolver (VQE) *Peruzzo, McClean, Nat Comm. 2014*:
  - Explore state space with parameterized unitary coupled cluster anstaz (A.A.G.)



$$T_c \ll T_1, T_2$$

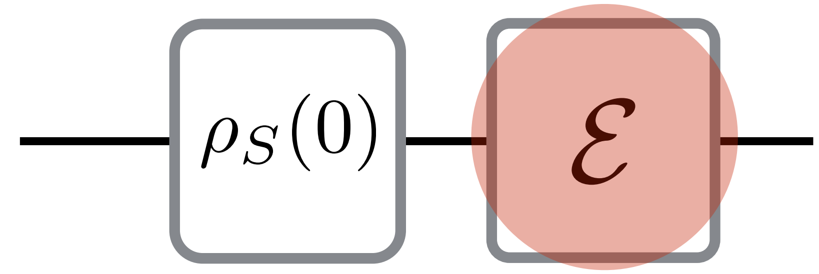
$$U(\boldsymbol{\theta}) \equiv e^{T(\boldsymbol{\theta}) - T^\dagger(\boldsymbol{\theta})}$$

$$|\Psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\Psi_{HF}\rangle$$



Low-depth computational approaches

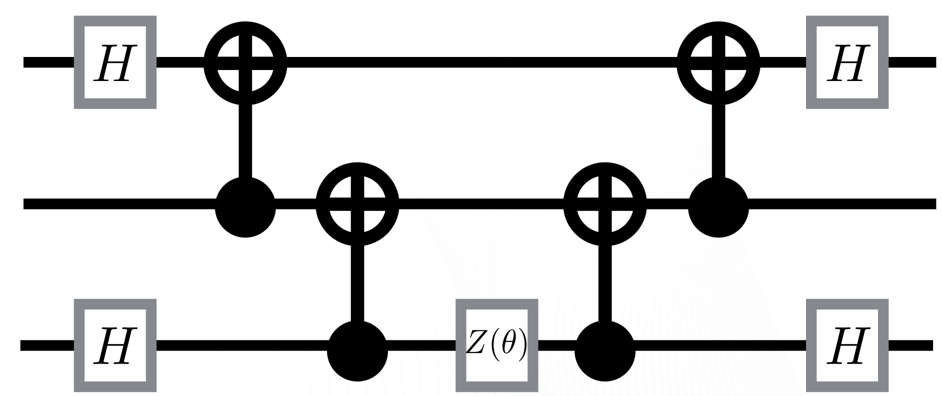
- Adiabatic (Analog) — D-WAVE, quantum simulators, etc.
- Variational Quantum Eigensolver (VQE) *Peruzzo, McClean, Nat Comm. 2014*:
  - Explore state space with parameterized unitary coupled cluster anstaz (A.A.G.)
  - Sample in polynomial number of computational basis to evaluate Hamiltonian cost function



$$T_c \ll T_1, T_2$$

$$U(\boldsymbol{\theta}) \equiv e^{T(\boldsymbol{\theta}) - T^\dagger(\boldsymbol{\theta})}$$

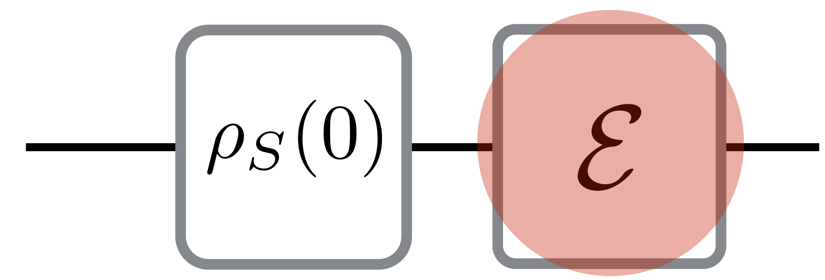
$$|\Psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\Psi_{HF}\rangle$$



$$\hat{H} = \sum_{pq} h_{pq} \hat{c}_p^\dagger \hat{c}_q + \frac{1}{2} \sum_{pqrs} h_{pqrs} \hat{c}_p^\dagger \hat{c}_q^\dagger \hat{c}_r \hat{c}_s$$

Low-depth computational approaches

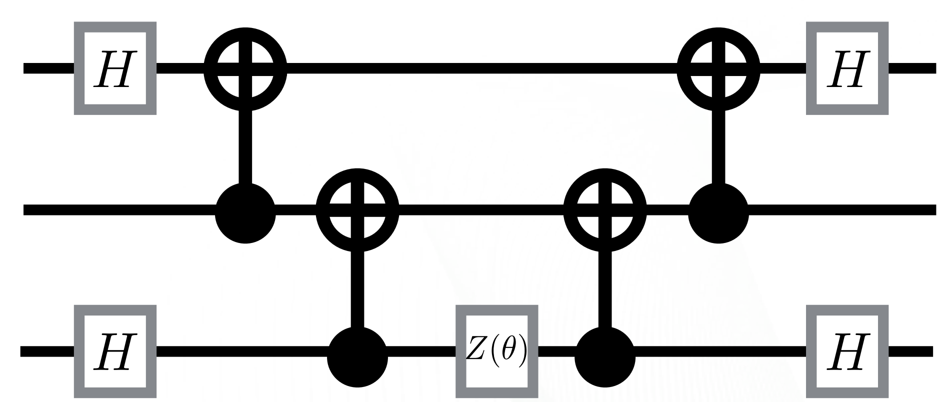
- Adiabatic (Analog) — D-WAVE, quantum simulators, etc.
- Variational Quantum Eigensolver (VQE) *Peruzzo, McClean, Nat Comm. 2014*:
  - Explore state space with parameterized unitary coupled cluster anstaz (A.A.G.)
  - Sample in polynomial number of computational basis to evaluate Hamiltonian cost function
  - Minimization run on classical computer.



$$T_c \ll T_1, T_2$$

$$U(\boldsymbol{\theta}) \equiv e^{T(\boldsymbol{\theta}) - T^\dagger(\boldsymbol{\theta})}$$

$$|\Psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\Psi_{HF}\rangle$$

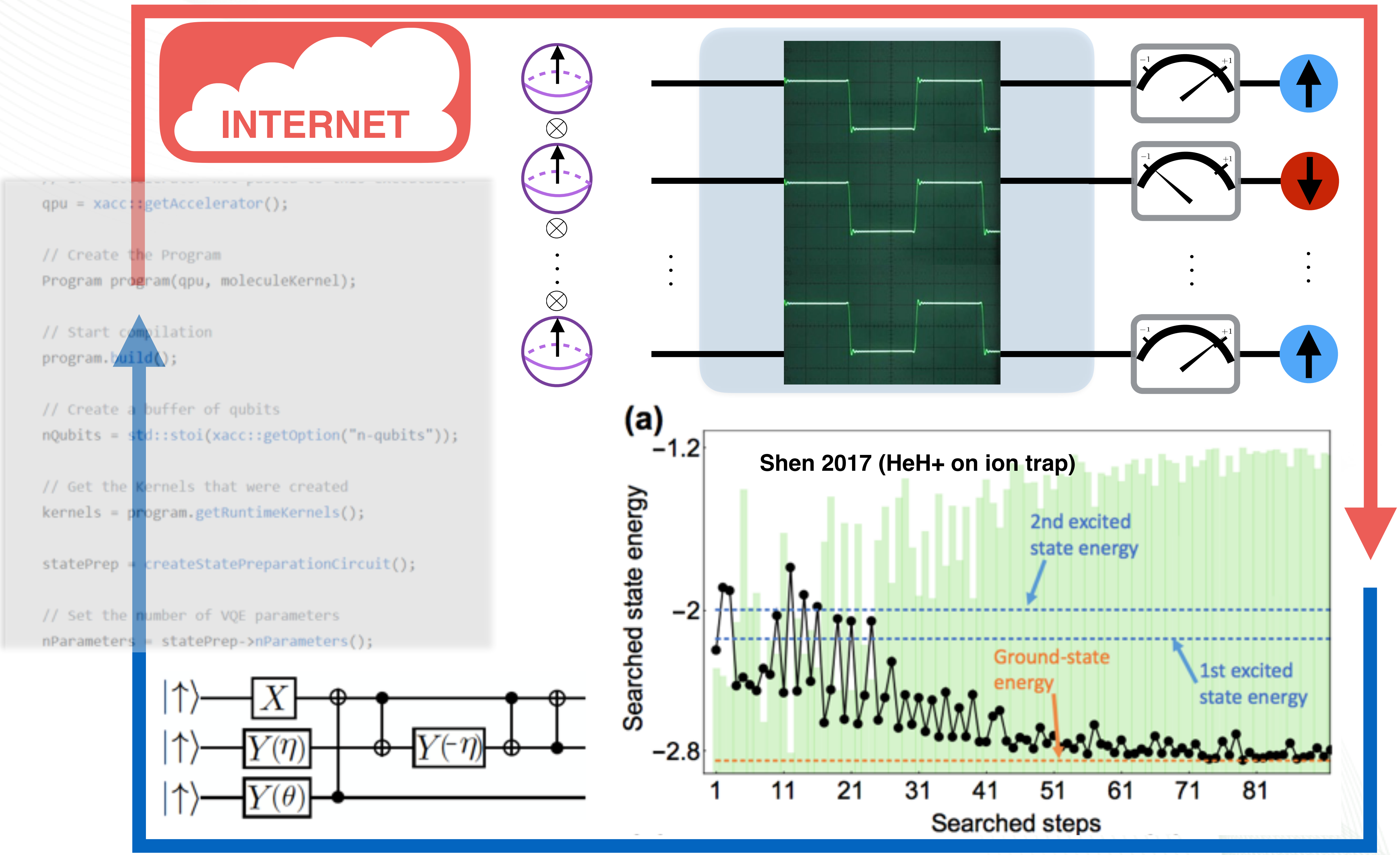


$$\hat{H} = \sum_{pq} h_{pq} \hat{c}_p^\dagger \hat{c}_q + \frac{1}{2} \sum_{pqrs} h_{pqrs} \hat{c}_p^\dagger \hat{c}_q^\dagger \hat{c}_r \hat{c}_s$$

$$\sum_i \langle \psi(\boldsymbol{\theta}) | h_i | \psi(\boldsymbol{\theta}) \rangle \geq E_0$$

Hybrid quantum-classical computing

quantum objective function



submit updated quantum program

Game plan ("simplest deuteron")

1. Hamiltonian from pionless EFT at leading order; fit to deuteron binding energy; constructed in harmonic-oscillator basis of ${}^3S_1$ partial wave [à la Binder et al. (2016); Bansal et al. (2017)]; cutoff at about 150 MeV.

$$H_N = \sum_{n,n'=0}^{N-1} \langle n'|(T+V)|n\rangle a_{n'}^\dagger a_n$$

$$\langle n'|V|n\rangle = V_0 \delta_n^0 \delta_n^{n'}$$

$$V_0 = -5.68658111 \text{ MeV}$$

2. Map single-particle states $|n\rangle$ onto qubits using $|0\rangle = |\uparrow\rangle$ and $|1\rangle = |\downarrow\rangle$. This is an analog of the Jordan-Wigner transform.

$$a_p^\dagger \leftrightarrow \sigma_-^{(p)} \equiv \frac{1}{2}(X_p - iY_p) \qquad a_p \leftrightarrow \sigma_+^{(p)} \equiv \frac{1}{2}(X_p + iY_p)$$

3. Solve H_1, H_2 (and H_3) and extrapolate to infinite space using harmonic oscillator variant of Lüscher's formula [More, Furnstahl, Papenbrock (2013)]

$$E_N = -\frac{\hbar^2 k^2}{2m}\left(1 - 2\frac{\gamma^2}{k}e^{-2kL} - 4\frac{\gamma^4 L}{k}e^{-4kL}\right) + \frac{\hbar^2 k \gamma^2}{m}\left(1 - \frac{\gamma^2}{k} - \frac{\gamma^4}{4k^2} + 2w_2 k\gamma^4\right)e^{-4kL}$$

Variational Wavefunction

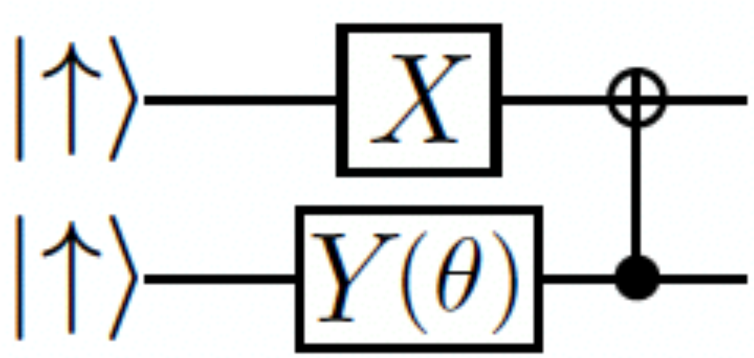
Wave functions on two qubits

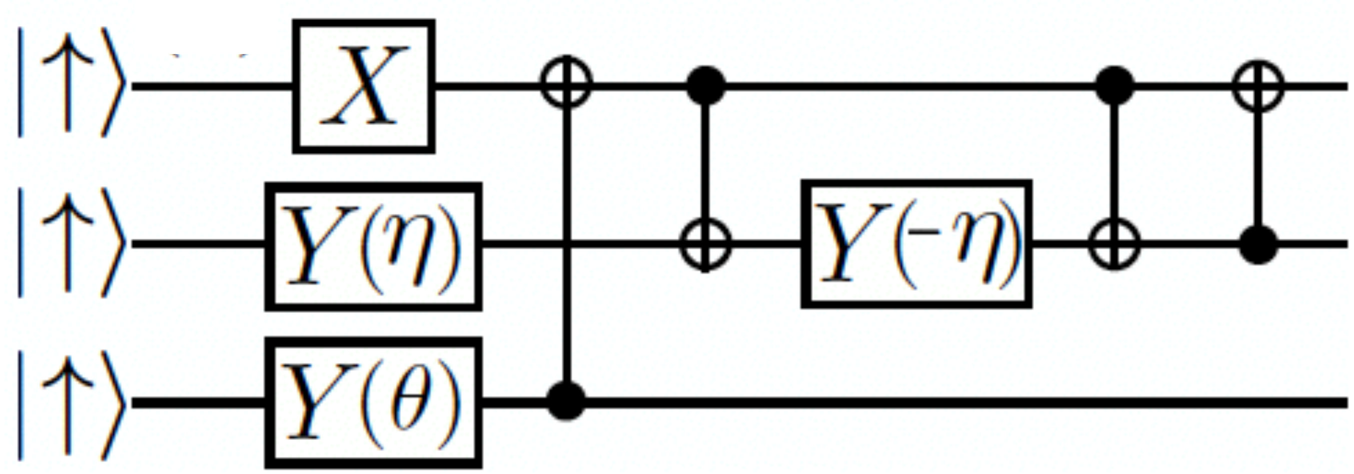
$$U(\theta)|\downarrow\uparrow\rangle \qquad U(\theta) \equiv e^{\theta\left(a_0^\dagger a_1 - a_1^\dagger a_0\right)} = e^{i\frac{\theta}{2}(X_0 Y_1 - X_1 Y_0)}$$

Wave functions on three qubits

$$U(\eta,\theta)|\downarrow\uparrow\uparrow\rangle \qquad U(\eta,\theta) \equiv e^{\eta\left(a_0^\dagger a_1 - a_1^\dagger a_0\right) + \theta\left(a_0^\dagger a_2 - a_2^\dagger a_0\right)}$$

Minimize number of two-qubit CNOT operations to mitigate low two-qubit fidelities (construct a “low-depth circuit”)





Variational Wavefunction

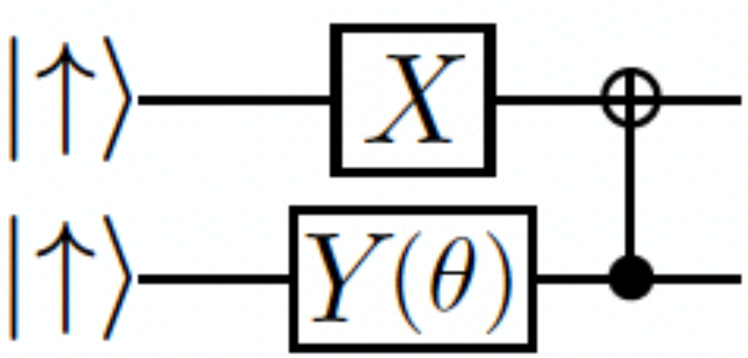
Wave functions on two qubits

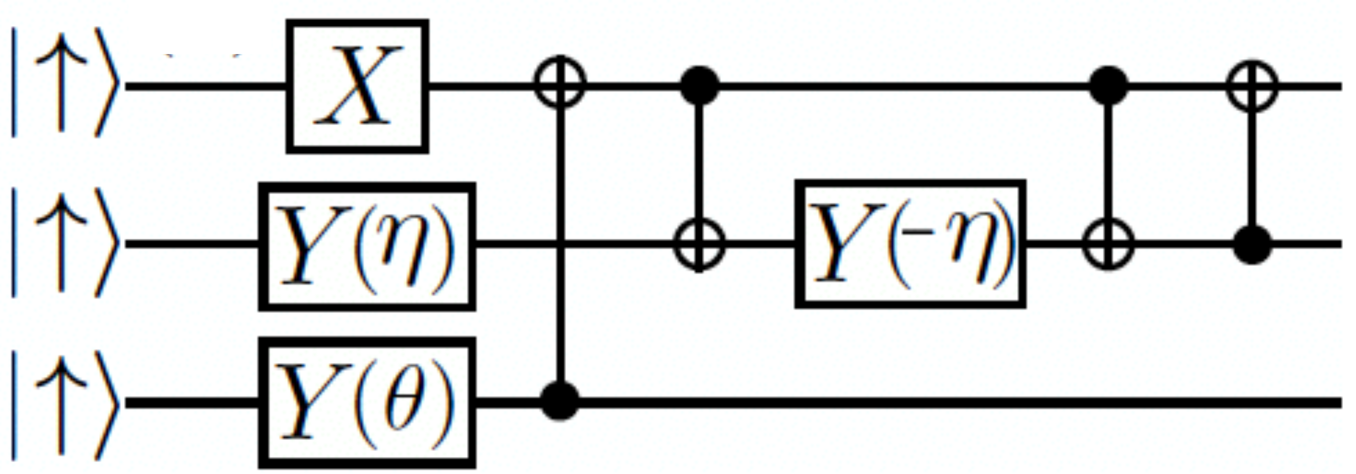
$$U(\theta)|\downarrow\uparrow\rangle \qquad U(\theta) \equiv e^{\theta\left(a_0^\dagger a_1 - a_1^\dagger a_0\right)} = e^{i\frac{\theta}{2}(X_0 Y_1 - X_1 Y_0)}$$

Wave functions on three qubits

$$U(\eta,\theta)|\downarrow\uparrow\uparrow\rangle \qquad U(\eta,\theta) \equiv e^{\eta\left(a_0^\dagger a_1 - a_1^\dagger a_0\right)+\theta\left(a_0^\dagger a_2 - a_2^\dagger a_0\right)}$$

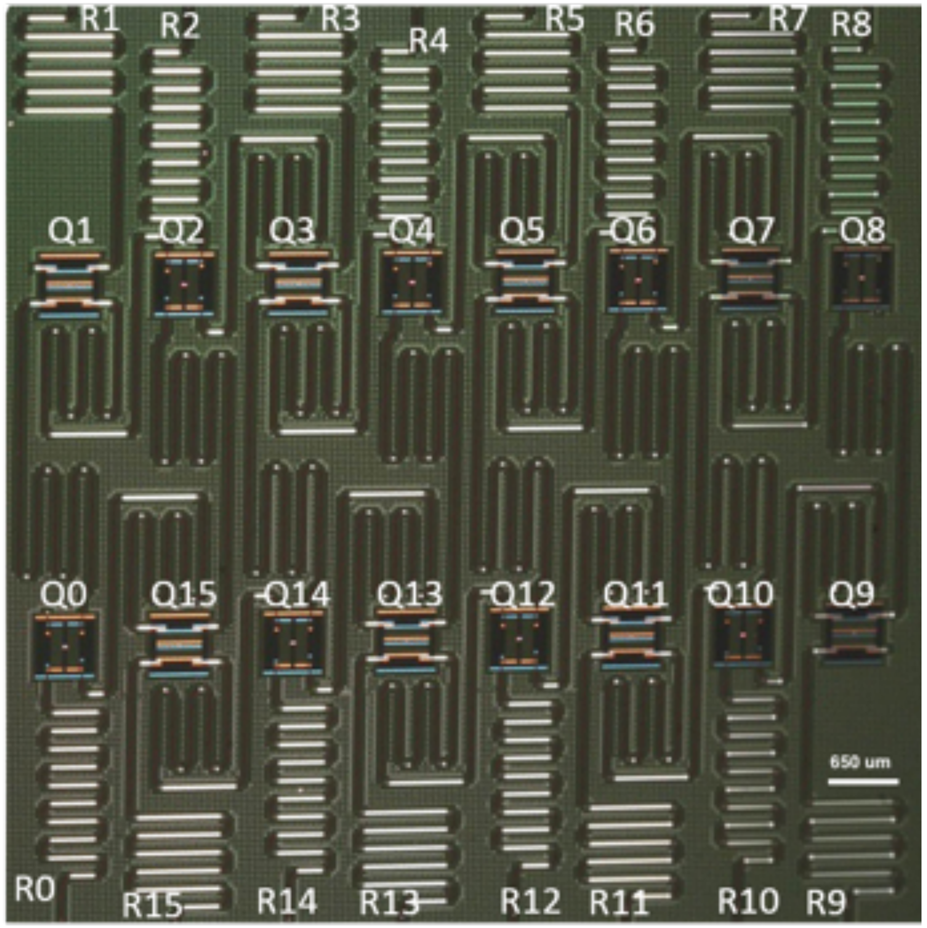
Minimize number of two-qubit CNOT operations to mitigate low two-qubit fidelities (construct a “low-depth circuit”)

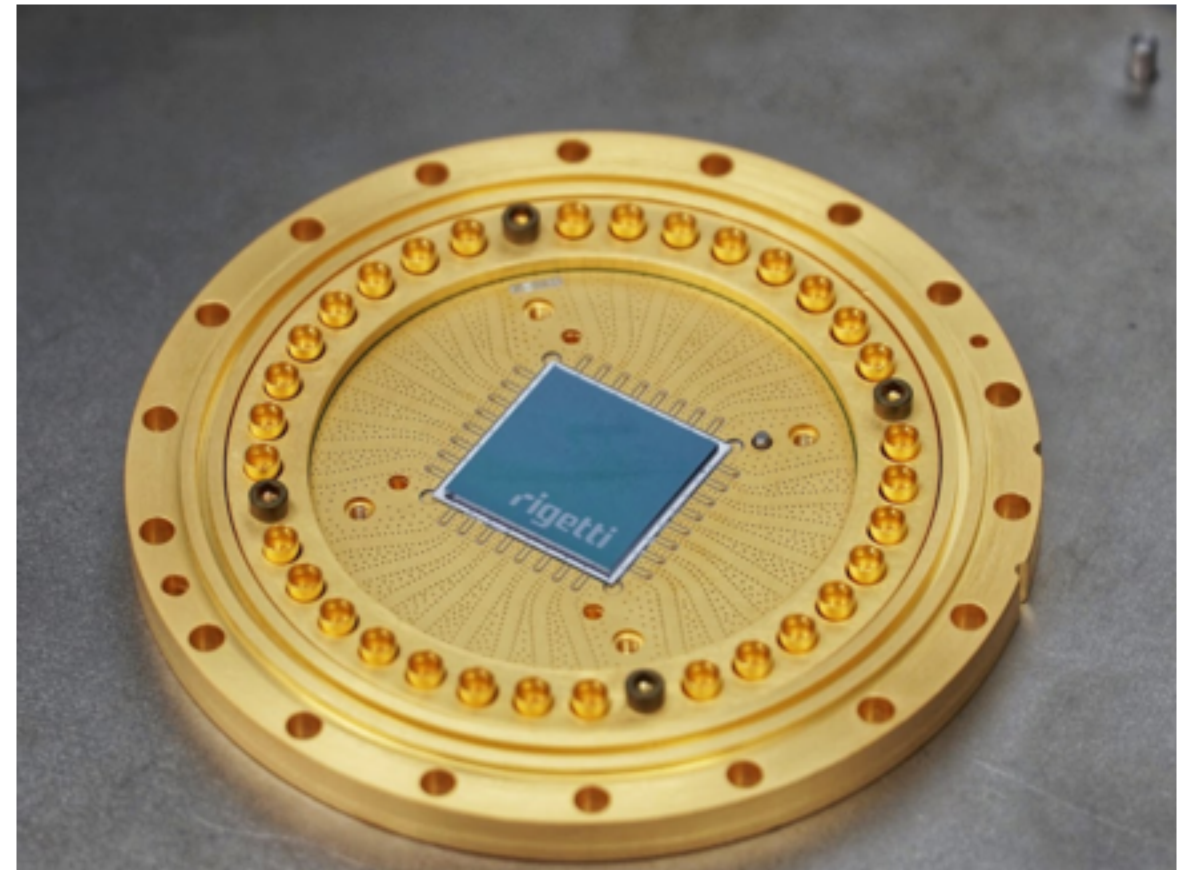


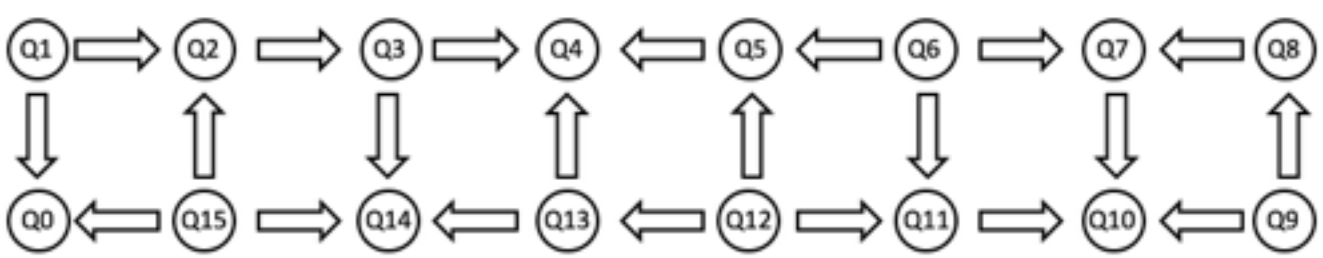


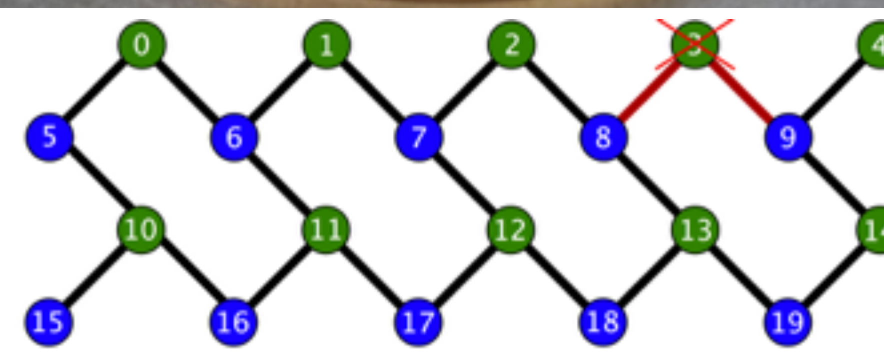
$$|0,0\rangle \xrightarrow{X\otimes e^{i\theta Y}} \cos(\theta)|1,0\rangle - \sin(\theta)|1,1\rangle \xrightarrow{\text{CNOT}(2,1)} \cos(\theta)|1,0\rangle - \sin(\theta)|0,1\rangle$$

Hardware Focus





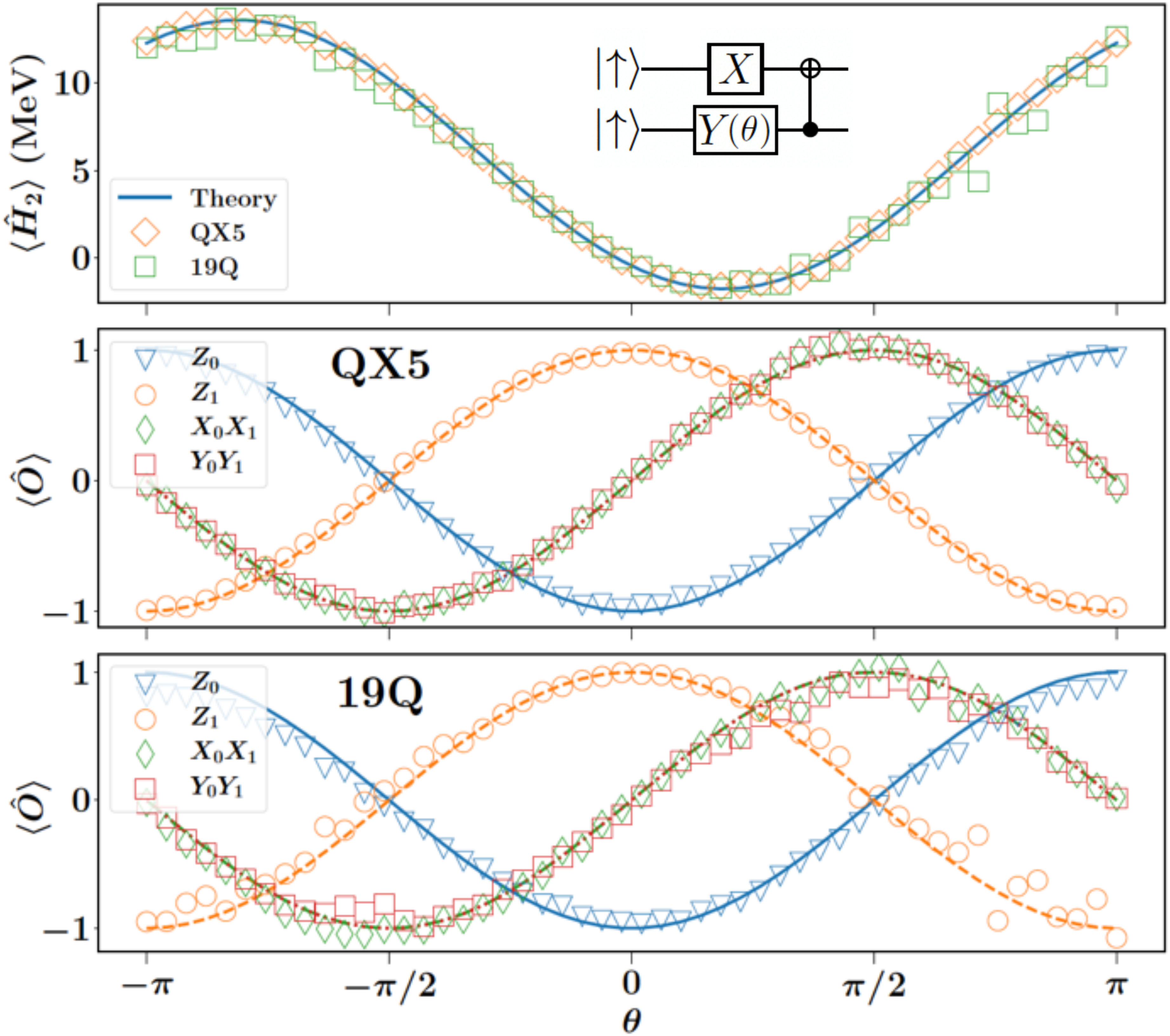




	1-Qubit Gate Fidelity			2-Qubit Gate Fidelity			Read Out Fidelity		
Computer	Min	Max	Ave	Min	Max	Ave	Min	Max	Ave
IBM QX5	99.59%	99.87%	99.77%	91.98%	97.29%	95.70%	88.53%	96.66%	93.32%
Rigetti 19Q	94.96%	99.42%	98.63%	79.00%	93.60%	87.50%	84.00%	97.00%	93.30%

Expectation on two qubits

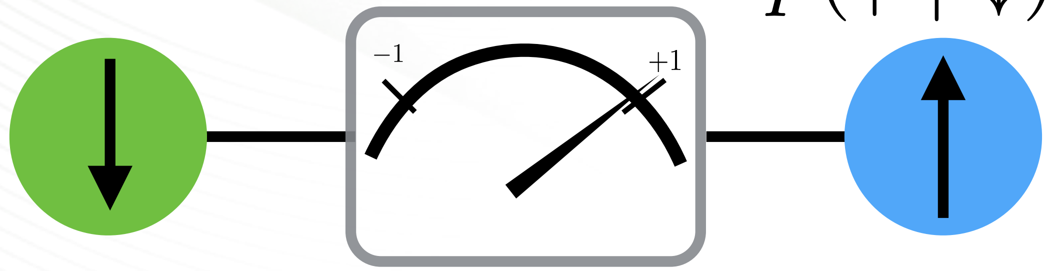
$$H_2 = 5.906709I + 0.218291Z_0 - 6.125Z_1 - 2.143304\left(X_0X_1 + Y_0Y_1\right)$$



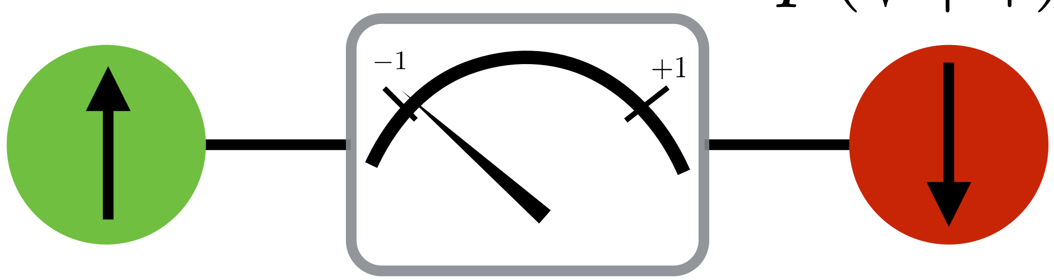
Readout post-processing

Readout post-processing

$p(\uparrow \mid \downarrow)$



$p(\downarrow \mid \uparrow)$



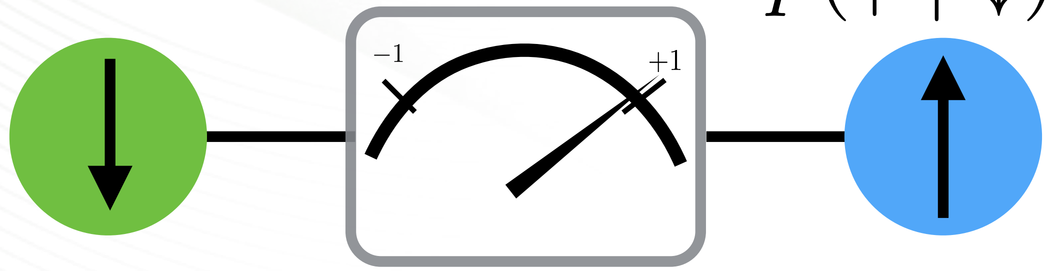
$$p_{\pm} = p(\uparrow \mid \downarrow) \pm p(\uparrow \mid \downarrow)$$

	Q0	Q1	Q2	Q3	Q4
Frequency (GHz)	4.84	4.48	4.88	5.03	5.01
T1 (µs)	71.94	78.41	62.65	57.68	47.50
T2 (µs)	18.27	15.63	22.23	29.33	45.38
Gate error (10^{-3})	17.06	30.68	7.97	12.43	3.52
Readout error (10^{-2})	6.45	27.35	10.40	21.75	8.75
	CX0_1	**CX1_0**	**CX2_1**	**CX3_4**	**CX4_3**
MultiQubit gate error (10^{-2})	4.04	4.04	5.82	3.45	3.45
	CX0_5	**CX1_2**	**CX2_7**	**CX3_9**	**CX4_8**
	13.93	5.82	5.86	6.28	3.79

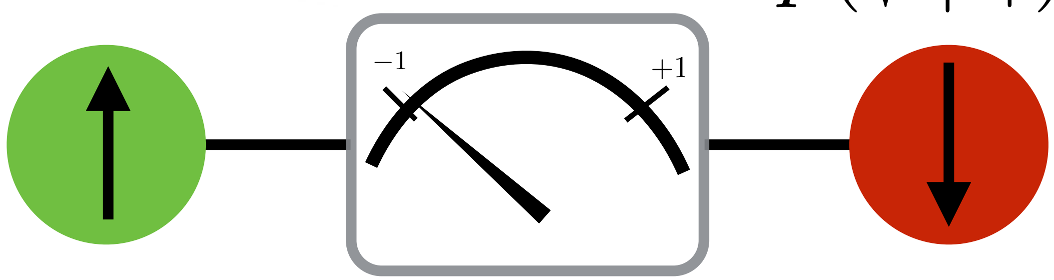
https://quantumexperience.ng.bluemix.net/qx/devices

Readout post-processing

$p(\uparrow \mid \downarrow)$



$p(\downarrow \mid \uparrow)$



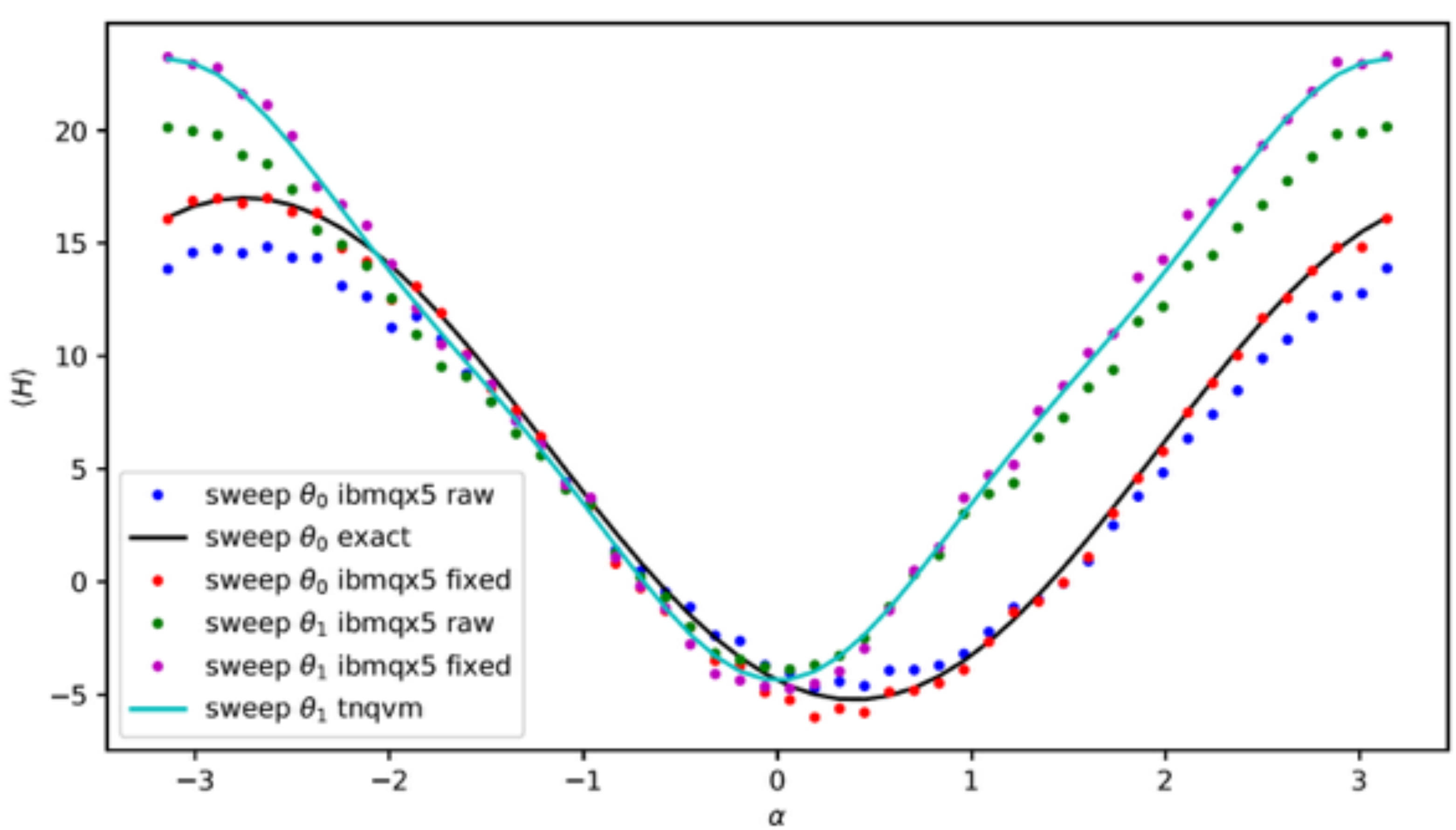
	Q0	Q1	Q2	Q3	Q4
Frequency (GHz)	4.84	4.48	4.88	5.03	5.01
T1 (μs)	71.94	78.41	62.65	57.68	47.50
T2 (μs)	18.27	15.63	22.23	29.33	45.38
Gate error (10^{-3})	17.06	30.68	7.97	12.43	3.52
Readout error (10^{-2})	6.45	27.35	10.40	21.75	8.75
	CX0_1	**CX1_0**	**CX2_1**	**CX3_4**	**CX4_3**
MultiQubit gate error (10^{-2})	4.04	4.04	5.82	3.45	3.45
	CX0_5	**CX1_2**	**CX2_7**	**CX3_9**	**CX4_8**
	13.93	5.82	5.86	6.28	3.79

https://quantumexperience.ng.bluemix.net/qx/devices

$$p_{\pm} = p(\uparrow \mid \downarrow) \pm p(\uparrow \mid \downarrow)$$

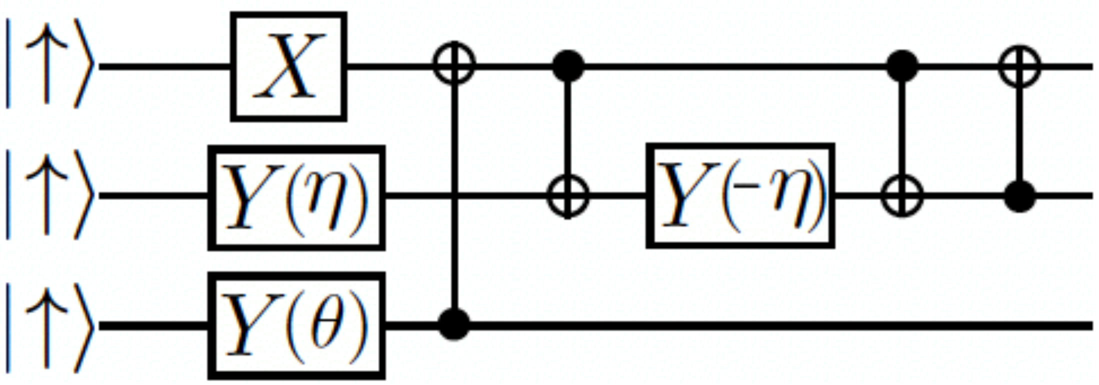
$$\langle \prod_{i \in s} Z_i \rangle \approx$$

$$\langle \prod_{i \in s} \frac{\tilde{Z}_i - p_-^i}{1 - p_+^i} \rangle$$



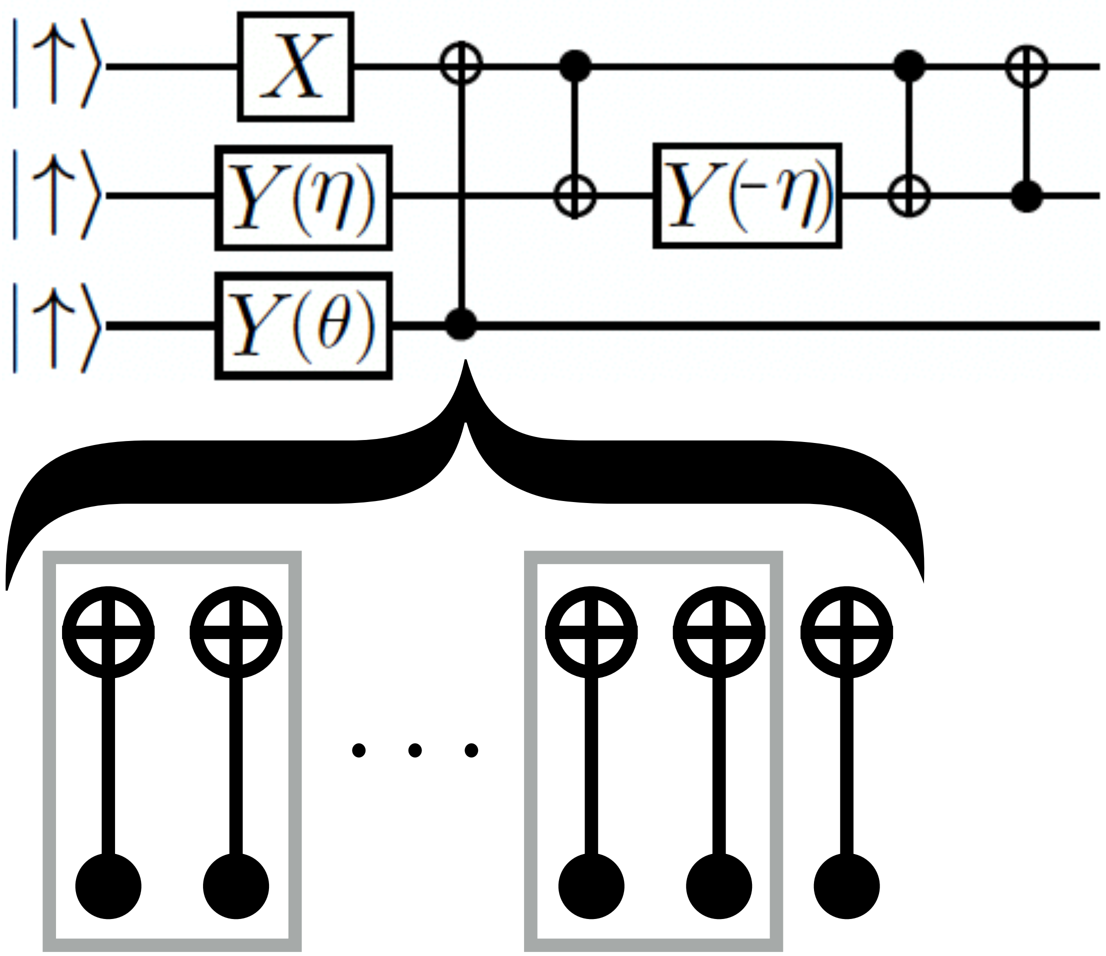
Three Qubits

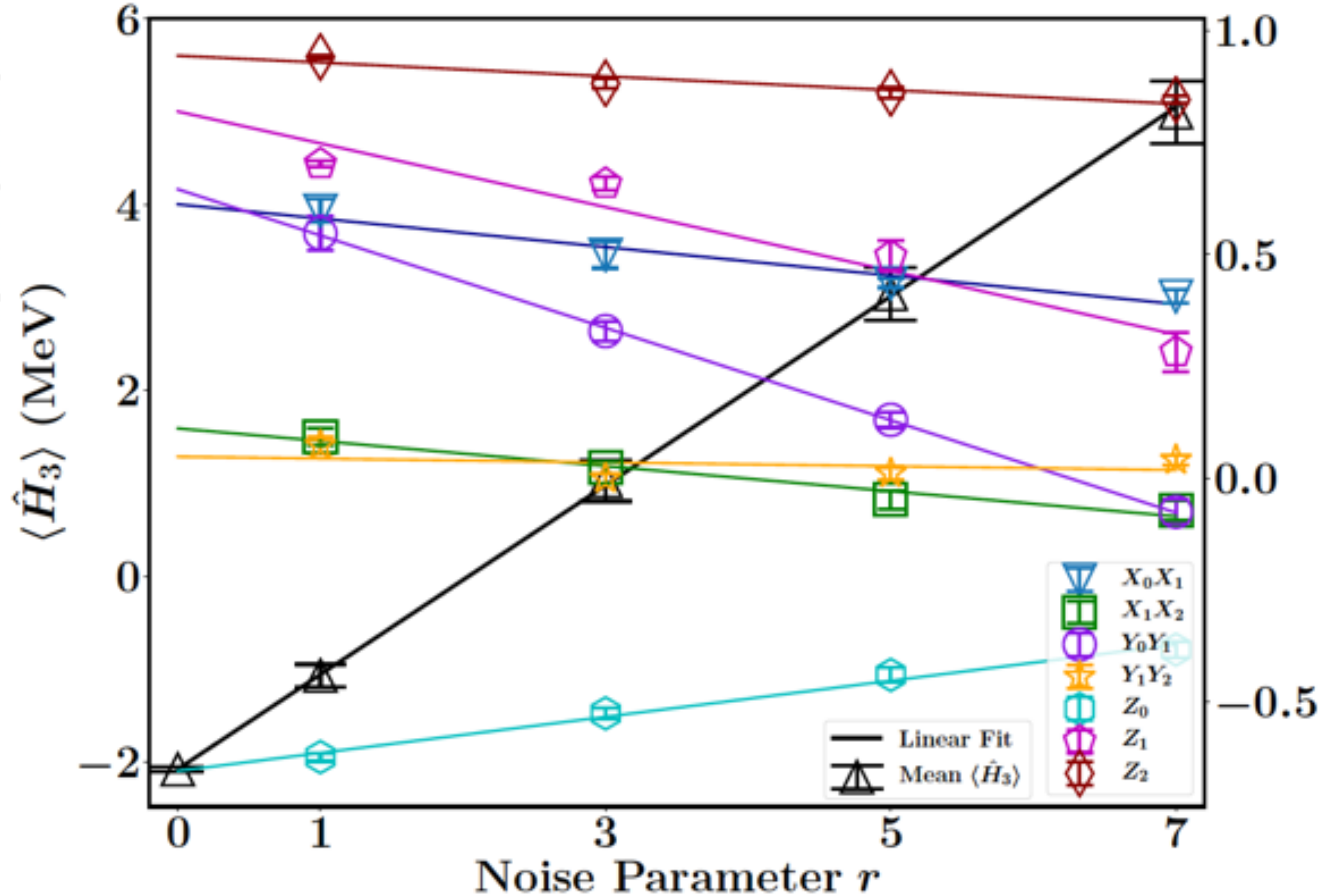
$$H_3 = H_2 + 9.625(I - Z_2) - 3.913119\,(X_1 X_2 + Y_1 Y_2)$$



Three Qubits

$$H_3 = H_2 + 9.625(I - Z_2) - 3.913119\,(X_1X_2 + Y_1Y_2)$$





$$\mathcal{E}(\rho) = (1-\varepsilon)\rho + \varepsilon I/4$$

$$\mathcal{E}_r(\rho) = (1-r\varepsilon)CX\rho CX + r\varepsilon I/4 + \mathcal{O}(\varepsilon^2)$$

FIG. 3. (Color online) Noise extrapolation of the $N = 3$ qubit problem run on the QX5. The H_3 energy (left axis, black line) and individual Pauli expectation values (right axis) are given as a function of the number CNOT gate scaling factor r.

Three qubits have more noise. Insert *r* pairs of CNOT (hermitian operators) to extrapolate to *r*=0. [See, e.g., Temme '17, Li '17, Kandala '18]

Final Results

Deuteron ground-state energies from a quantum computer compared to the exact result, E_∞=-2.22 MeV.

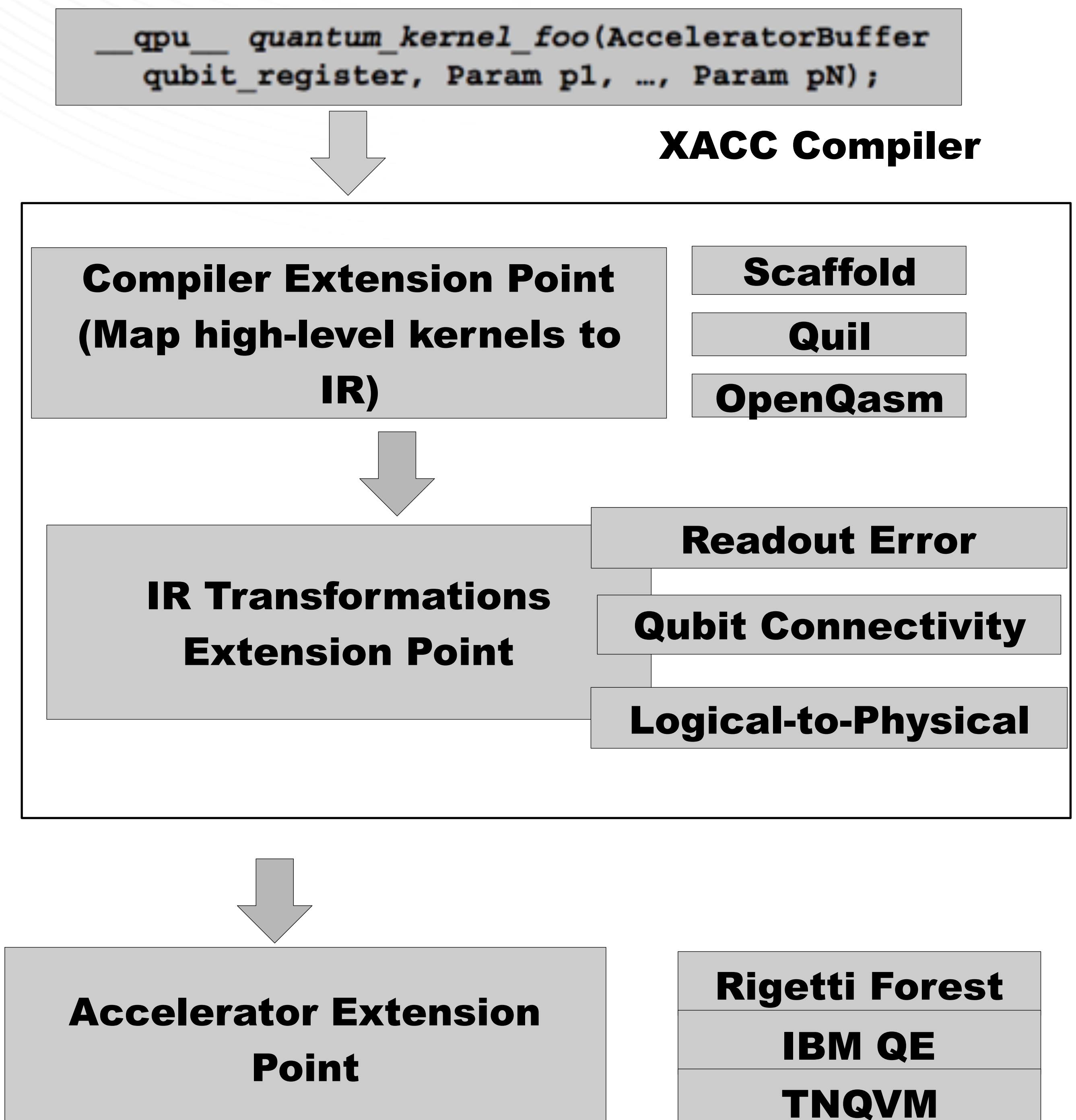
E from exact diagonalization				
N	E_N	$\mathcal{O}(e^{-2kL})$	$\mathcal{O}(kLe^{-4kL})$	$\mathcal{O}(e^{-4kL})$
2	-1.749	-2.39	-2.19	
3	-2.046	-2.33	-2.20	-2.21
E from quantum computing				
N	E_N	$\mathcal{O}(e^{-2kL})$	$\mathcal{O}(kLe^{-4kL})$	$\mathcal{O}(e^{-4kL})$
2	$-1.74(3)$	$-2.38(4)$	$-2.18(3)$	
3	$-2.08(3)$	$-2.35(2)$	$-2.21(3)$	$-2.28(3)$

$$E_N = -\frac{\hbar^2 k^2}{2m}\left(1 - 2\frac{\gamma^2}{k}e^{-2kL} - 4\frac{\gamma^4 L}{k}e^{-4kL}\right) + \frac{\hbar^2 k\gamma^2}{m}\left(1 - \frac{\gamma^2}{k} - \frac{\gamma^4}{4k^2} + 2w_2 k\gamma^4\right)e^{-4kL}$$

[Dumitrescu, McCaskey, Hagen, Jansen, Morris, Papenbrock, Pooser, Dean, Lougovski, PRL 120.210501]

Quantum Programming with XACC

Users define Quantum Kernels (Kernel in the GPU sense, a C-like Function)



XACC:

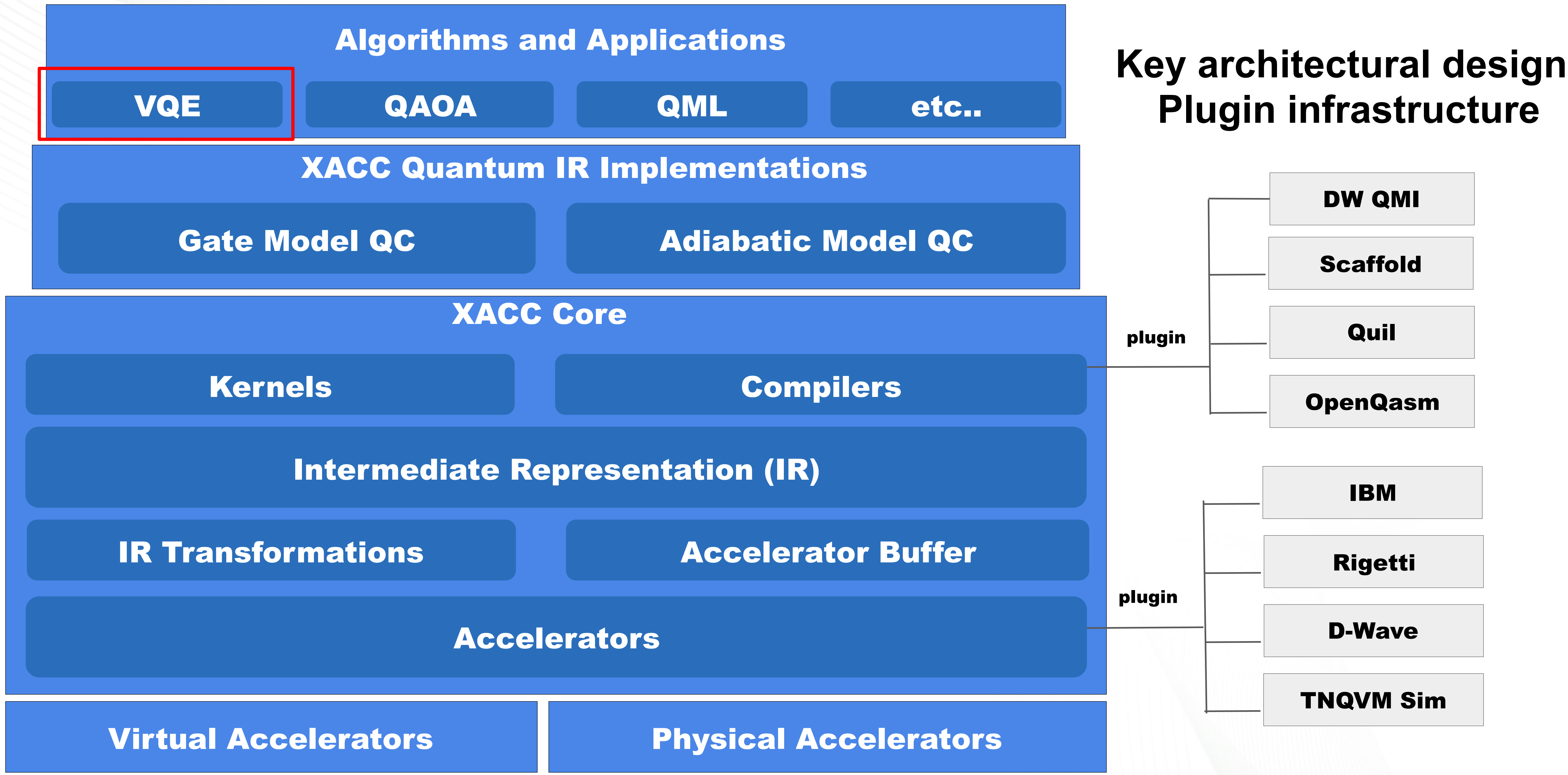
- Open-source at https://github.com/eclipse/xacc
- Familiar API and model
  - OpenCL-like, hardware agnostic
- Key Abstractions
  - Kernels
  - Compilers
  - Intermediate Representation
  - Accelerators
- OSGI C++ Plugin Architecture (Python API available)

```
// Get reference to the QPU, and allocate a buffer of qubits
auto qpu = xacc::getAccelerator("ibm");
auto buffer = qpu->createBuffer("qreg", 2);

// Create and compile the Program from the kernel
// source code. Get executable Kernel source code
auto kernelSourceCode = "__qpu__ foo(double theta) {....}";
xacc::Program program(qpu, kernelSourceCode);
program.build();
auto kernel = program.getKernel<double>("foo");

// Execute over theta range
for (auto theta : thetas) kernel.execute(buffer, theta);
```

Current XACC Application Stack



XACC is the ONLY quantum programming model and framework that enables hardware-agnostic quantum programming, compilation, and execution. ORNL owns the only integration framework for quantum computing.

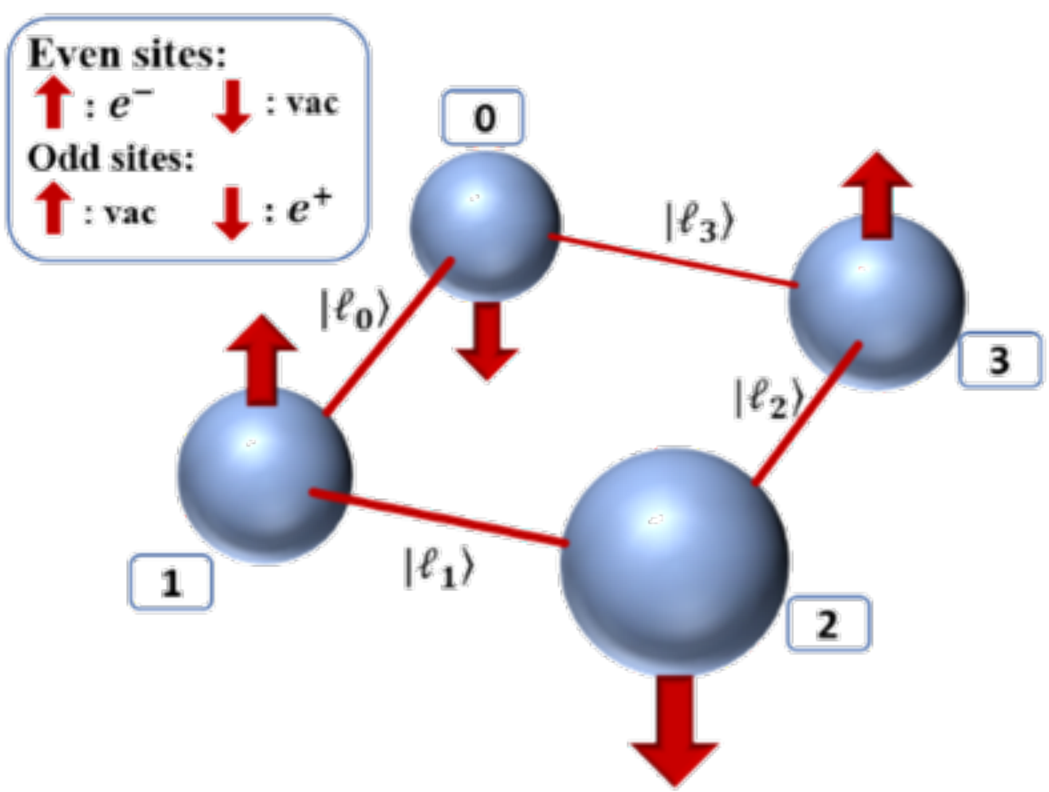
Quantum Field Theories

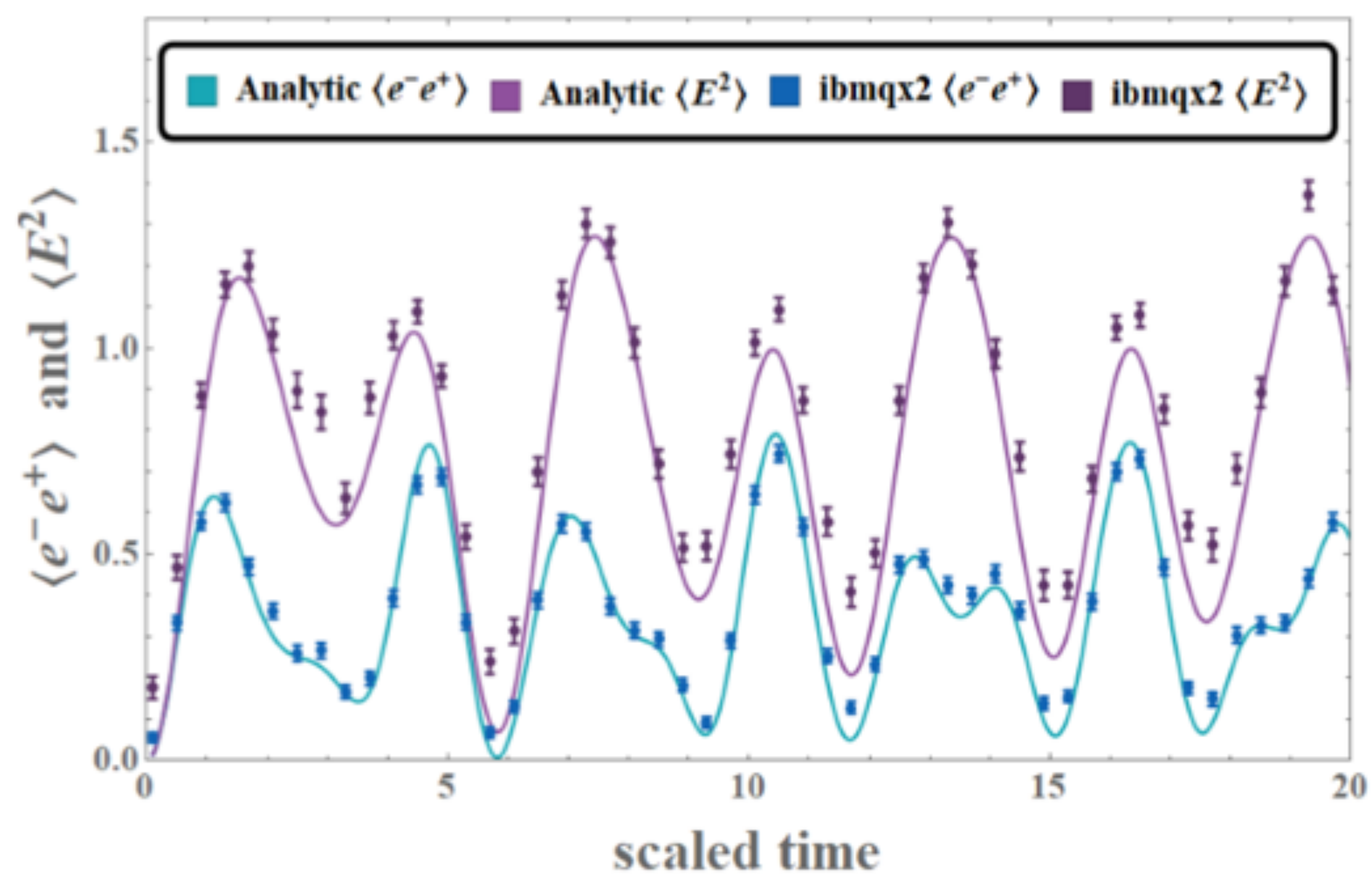
Quantum-Classical Dynamical Calculations of the Schwinger Model using Quantum Computers

N. Klco, E. F. Dumitrescu, A. J. McCaskey, T. D. Morris, R. C. Pooser, M. Sanz, E. Solano, P. Lougovski, M. J. Savage

(Submitted on 8 Mar 2018)

We present a quantum-classical algorithm to study the dynamics of the two-spatial-site lattice Schwinger model on IBM's quantum computers. Using rotational symmetries, total charge, and parity, the number of qubits needed to perform calculations is reduced by a factor of ~5 when compared with the naive lattice formulations, thereby dramatically reducing the dimensionality of the Hilbert space by removing exponentially-large unphysical sectors. Our work opens an avenue for calculations in larger lattice quantum field theories where classical computation is used to find symmetry sectors in which the quantum computer evaluates the dynamics of quantum fluctuations.





Summary

- Testing the low-depth hybrid quantum-classical model of computation (alternative is to patiently wait for fault tolerance)
- Use qubit registers ‘naturally’ encode many-body state space with poly resource cost.
- Apply variational principle, sample energy functional on quantum device, control quantum program with a classical program.
- Application to the binding energy of the deuteron
- Gave details of steps needed to achieve accurate result. Multi-layered workflow involving quantum and classical routines.
- Aim is to correctly solve simplest problems, then test limits of scalability. Establishing meaningful scientific benchmarks (across a variety of quantum computing platforms)
- Along with hardware progress, there is a need for progress in robust hybrid algorithms.

Thank You!