advantage 2020

Variational Quantum Linear Solver

Ryan LaRose

Department of Computational Mathematics, Science, and Engineering, Michigan State University

Motivation

- Solving linear systems Ax = b is a standard computational problem.
- Quantum algorithms for linear systems [1, 2] have **huge** overhead [3].

4x4 linear system

HHL [1]

p	Number of gates
5	580,000
6	660,000
7	780,000
8	900,000

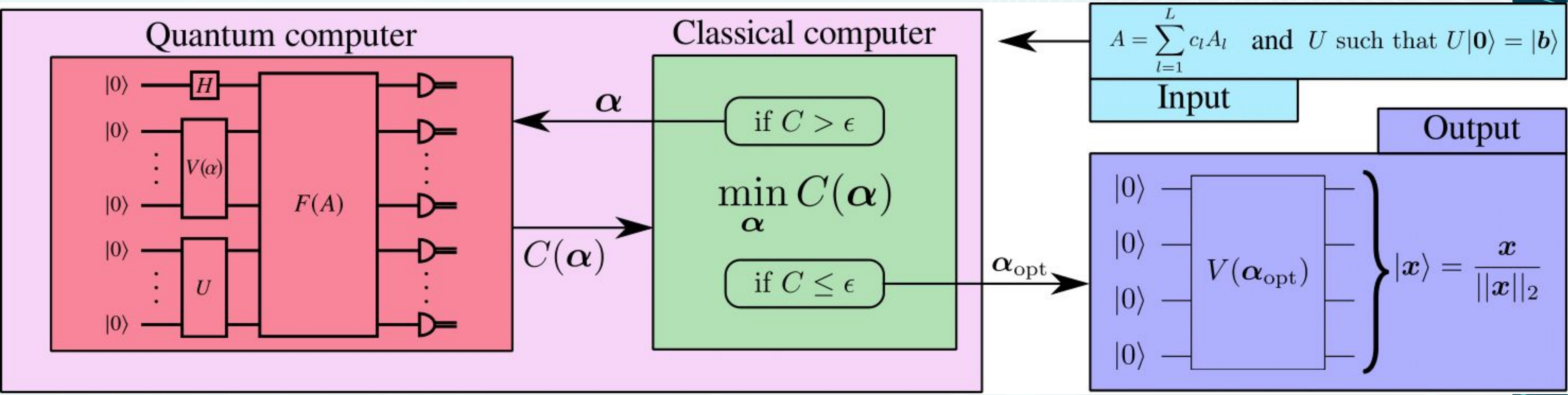
QSVE Linear Solver [2]

p	Number of gates
5	19,000
6	50,000
7	150,000
8	496,000

- Can we hope to do anything (useful) in the near-term?

VQLS: Algorithm Overview

- The Variational Quantum Linear Solver (VQLS) can be run on near-term processors.

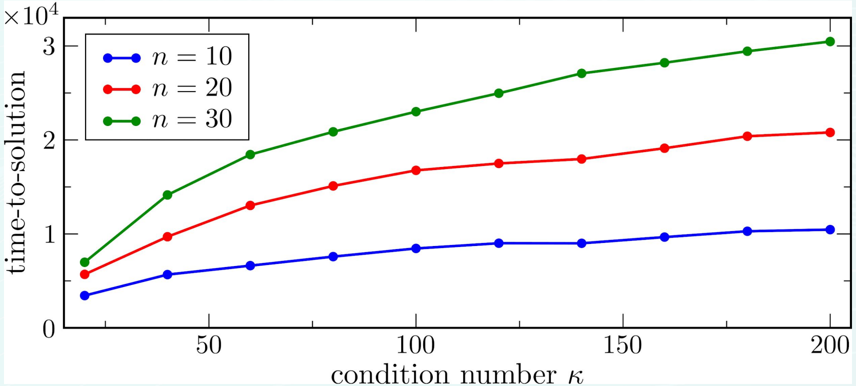


VQLS can serve as an interesting **problem-specific benchmark** for near-term quantum computers.

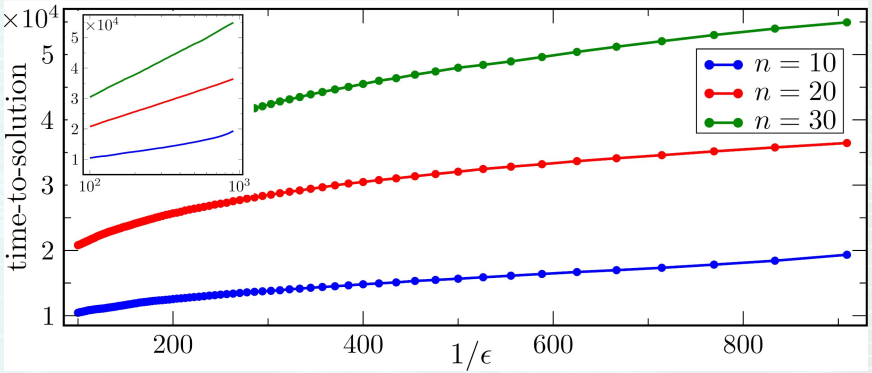
VQLS: Algorithm Overview

- Computing the cost function is DQC1-hard.
- Heuristic results for scaling in:

Condition number of input matrix



Desired precision of solution



VQLS: Results from Rigetti Aspen-4

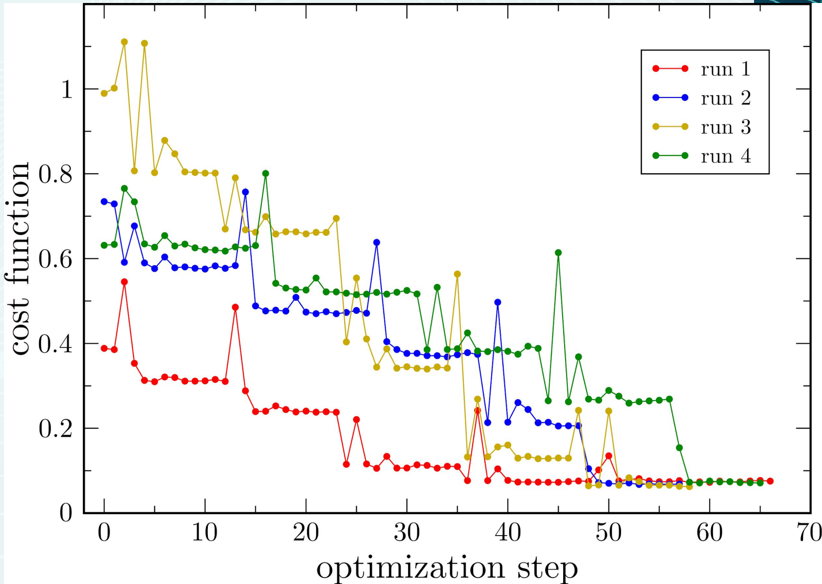
- To run on QPUs, we use the **effective Hamiltonian** approach

$$H_{A,\mathbf{b}} := A^\dagger(I - |\mathbf{b}\rangle\langle\mathbf{b}|)A$$

- We looked at a linear system on 3-5 qubits on **Aspen-4**

$$A = I + 0.2X_1Z_2 + 0.2X_1$$
$$b = H|0\rangle$$

  - Able to successfully solve the linear systems on Aspen-4



Results on Rigetti Aspen-7

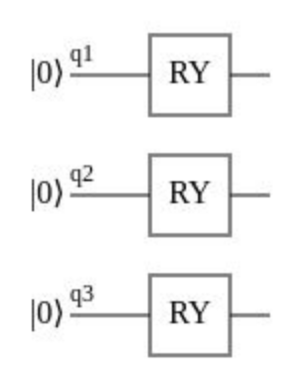
Experiments

1. Cost landscape with no entanglement, 3 qubit LS
2. Optimization with entanglement, 3 qubit LS
3. Cost landscape with no entanglement, 5 qubit LS
4. Optimization with entanglement, 5 qubit LS
5. Ising model LS, 8-10 qubits

Three qubit linear system

$$A = I + 0.2X_1Z_2 + 0.2X_1$$
$$b = H|0\rangle$$

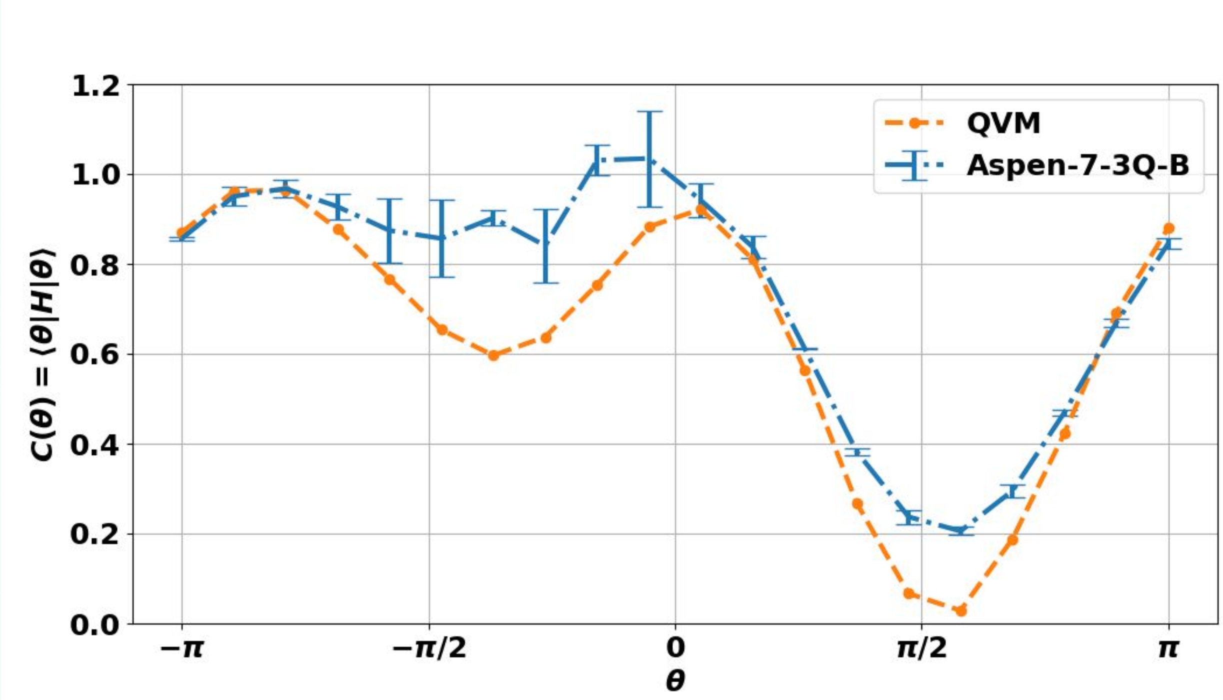
Cost landscape matches (noiseless) simulator quite well.



3 QUBITS

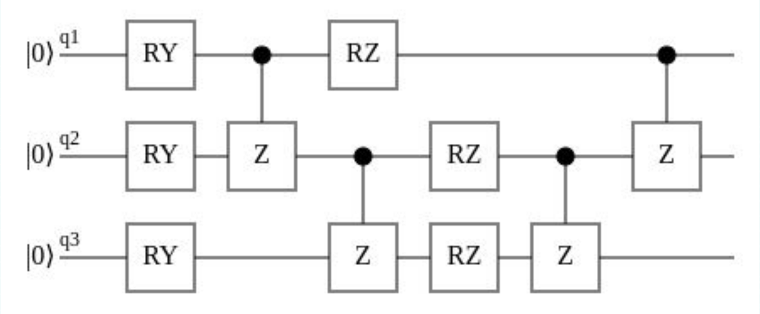
Aspen-7-3Q-B

T1	29 μs
T2	27 μs
f1QRB	99.92% ± 0.02%
f1Q sim. RB	99.02% ± 0.1%
fActiveReset	98.3%
fRO	95.13%
fCZ	89.9% ± 0.66%



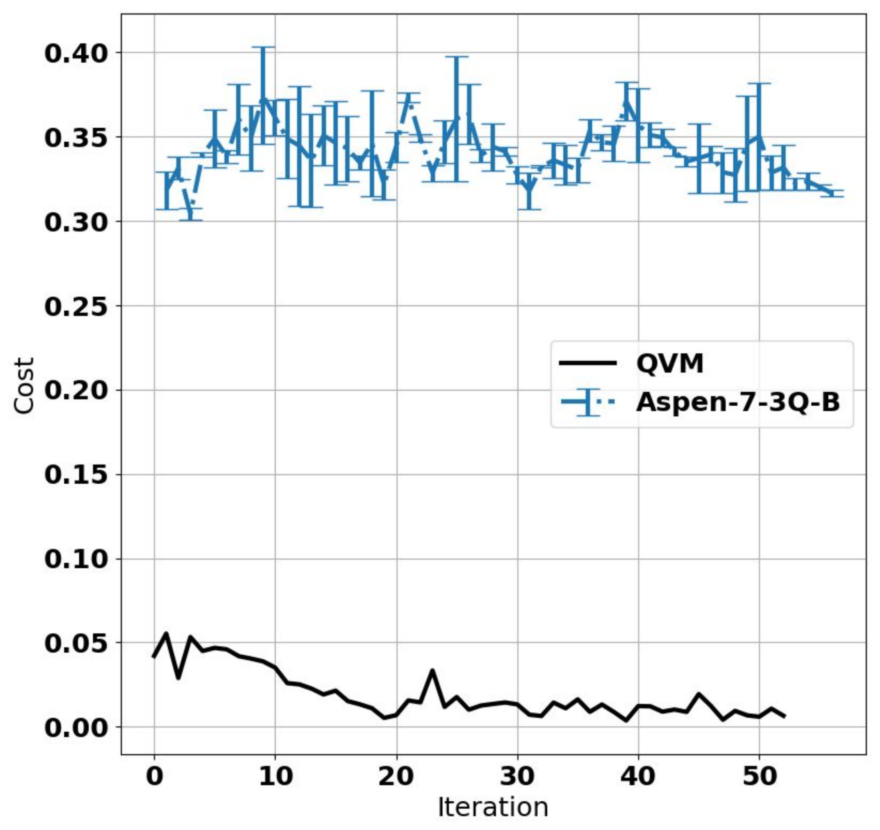
Three qubit linear system

Add entangling gates to ansatz:



Solution to LS is representable by ansatz.

CZ gates increase noise.

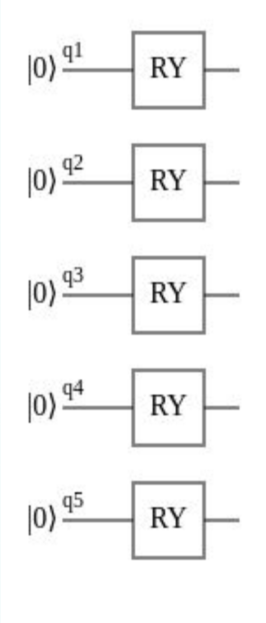


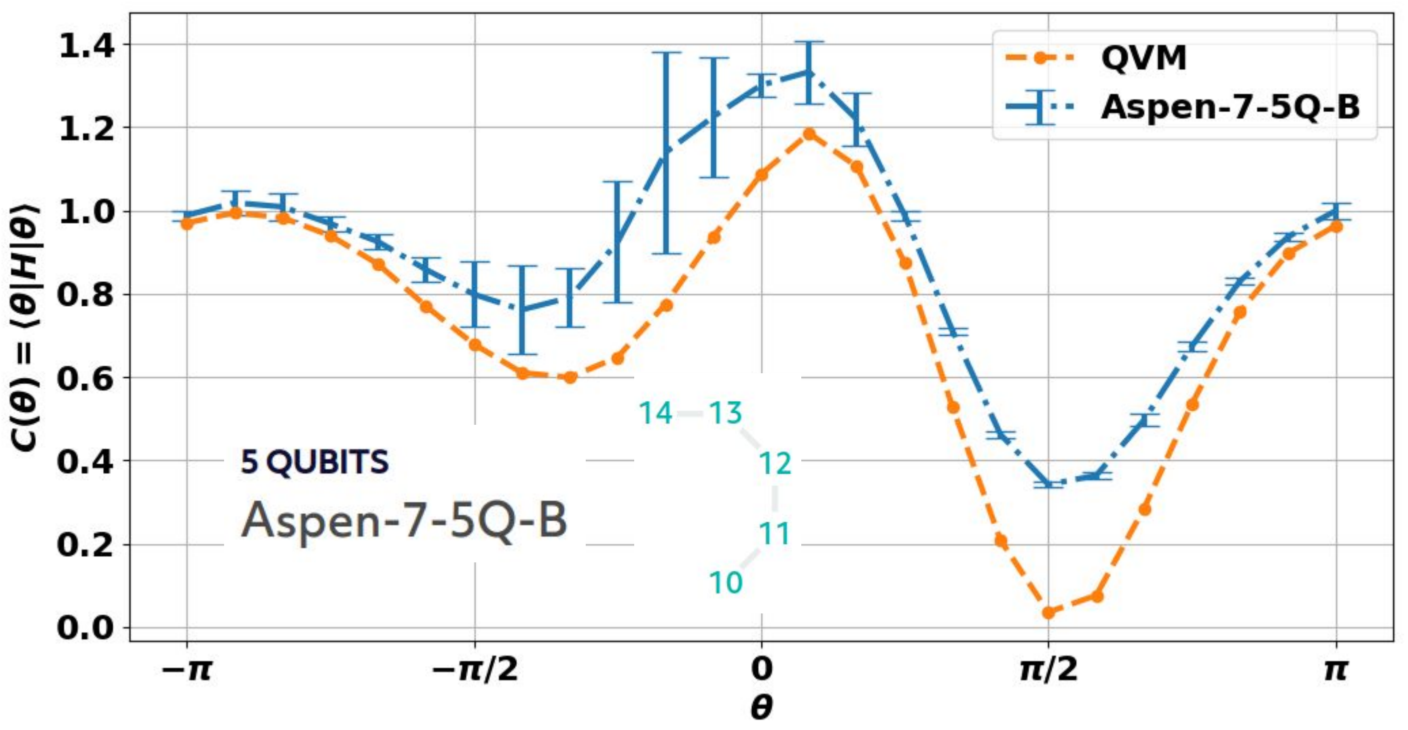
Start near optimal parameters

Five qubit linear system

$$A = I + 0.2X_1Z_2 + 0.2X_1$$
$$b = H|0\rangle$$

Cost landscape still matches (noiseless) simulator well, as for 3 qubits

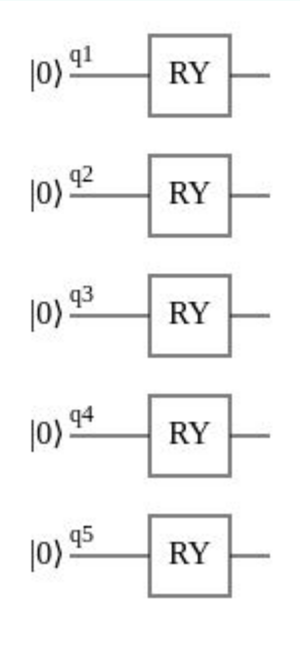


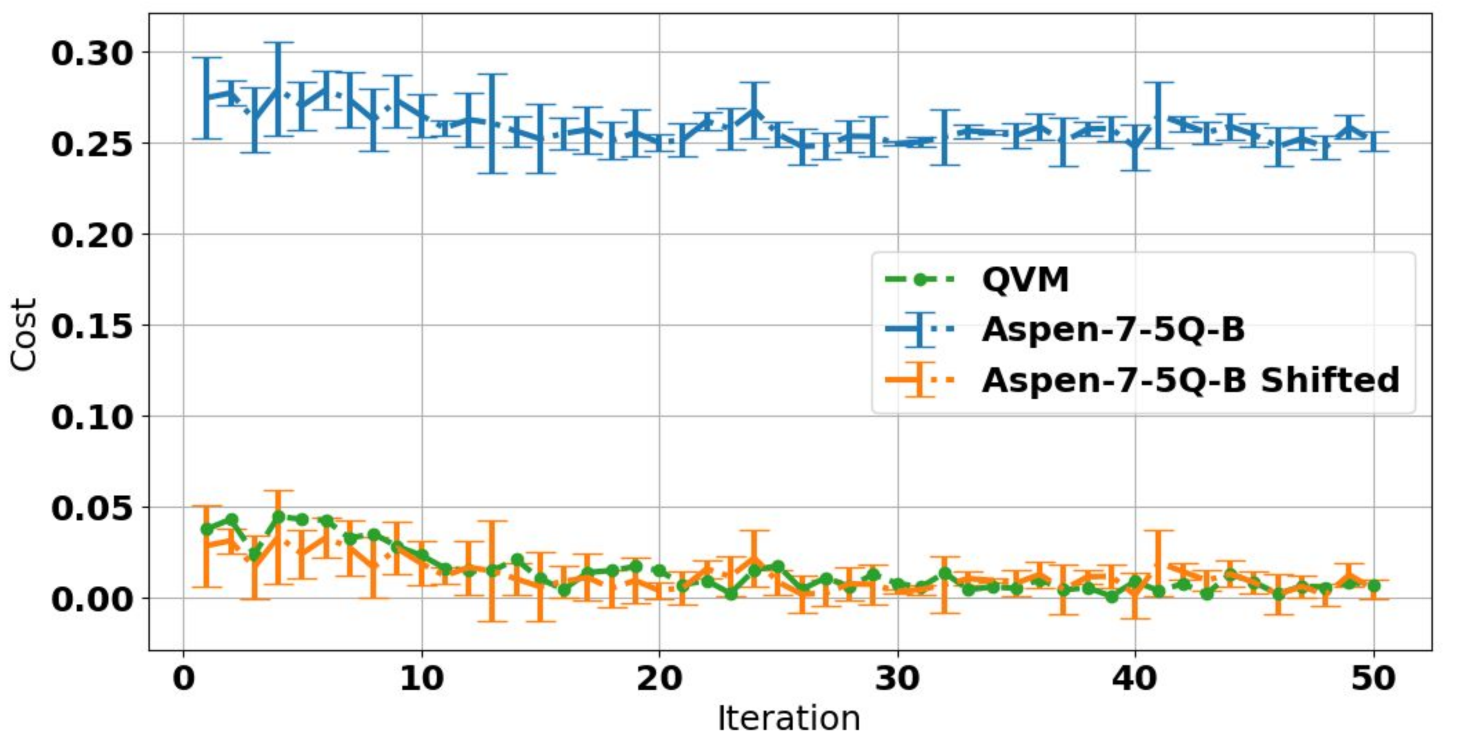


Five qubit linear system

$$A = I + 0.2X_1Z_2 + 0.2X_1$$
$$b = H|0\rangle$$

Cost vs iteration, start with optimal parameters found by landscape search

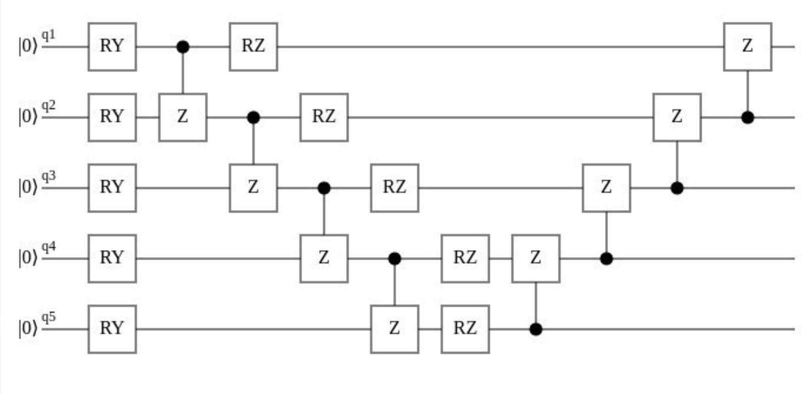


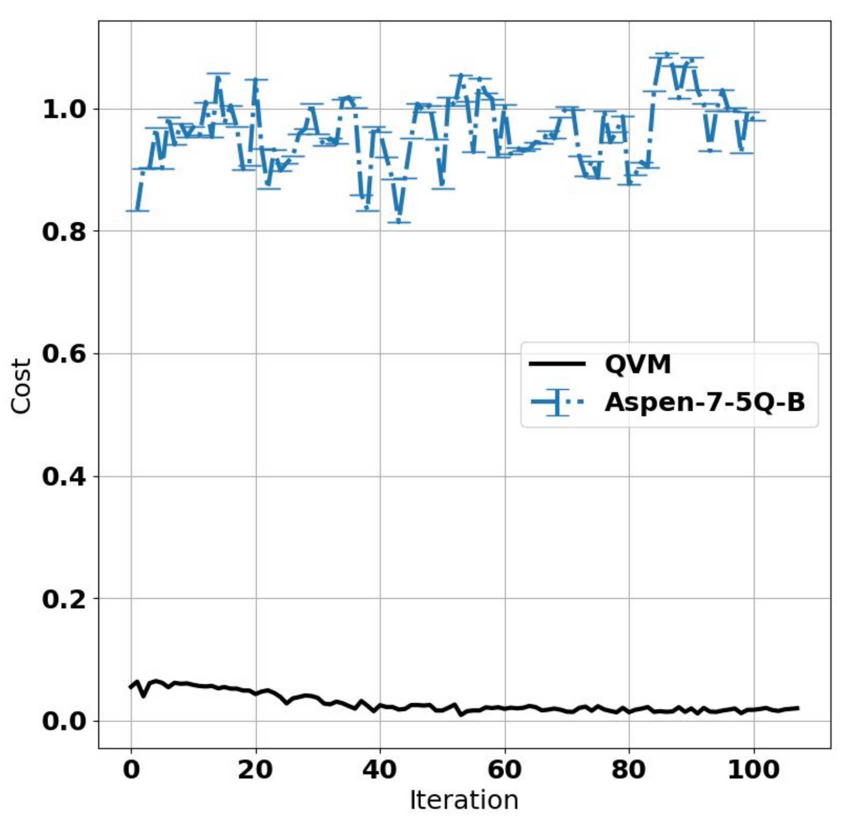


Five qubit linear system

$$A = I + 0.2X_1Z_2 + 0.2X_1$$
$$b = H|0\rangle$$

Add entangling gates to ansatz.





Vertical shift in cost notably higher than for 3 qubit system

Ising model linear system

Consider the linear system formed by the transverse field Ising model:

$$A := \frac{1}{\zeta}\left(\eta I + \sum_{j=1}^{n} X_j + J \sum_{j=1}^{n-1} Z_j Z_{j+1}\right)$$

Why? For small J, solution can be represented by MPS.

We choose

$$\zeta = \eta = 1$$

$$J = 0.1$$

$$|b\rangle = |0\rangle$$

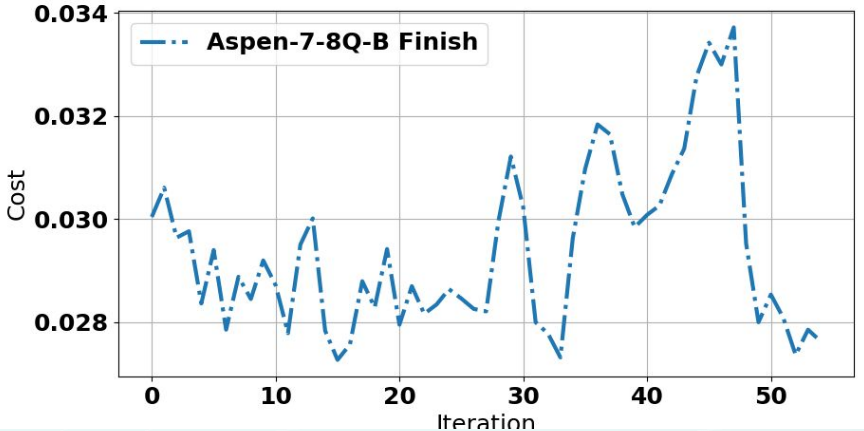
```
A =
 [[0.3  0.25 0.25 0.   0.25 0.   0.   0.  ]
 [0.25 0.25 0.   0.25 0.   0.25 0.   0.  ]
 [0.25 0.   0.2  0.25 0.   0.   0.25 0.  ]
 [0.   0.25 0.25 0.25 0.   0.   0.   0.25]
 [0.25 0.   0.   0.   0.25 0.25 0.25 0.  ]
 [0.   0.25 0.   0.   0.25 0.2  0.   0.25]
 [0.   0.   0.25 0.   0.25 0.   0.25 0.25]
 [0.   0.   0.   0.25 0.   0.25 0.25 0.3 ]]
```

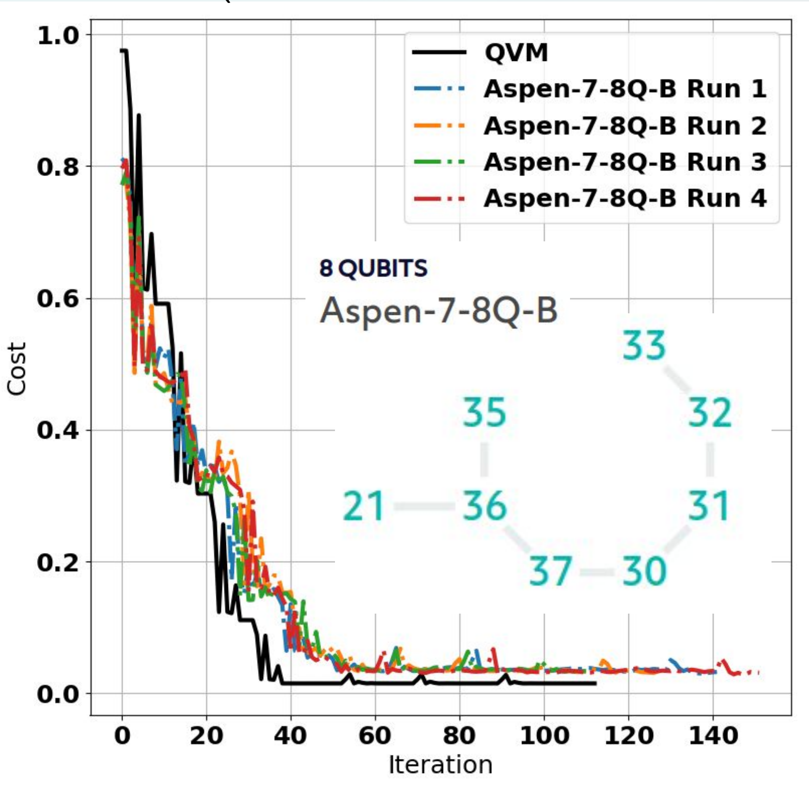
Ising model linear system: 8 qubits

$$A := \frac{1}{\zeta}\left(\eta I + \sum_{j=1}^{n} X_j + J\sum_{j=1}^{n-1} Z_j Z_{j+1}\right)$$

Optimization on hardware matches noiseless simulator very well.

Able to solve **a** 256 x 256 linear system.

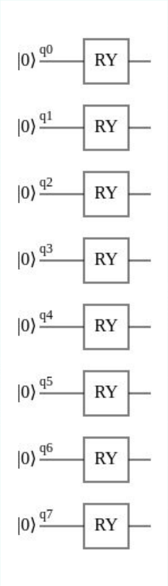


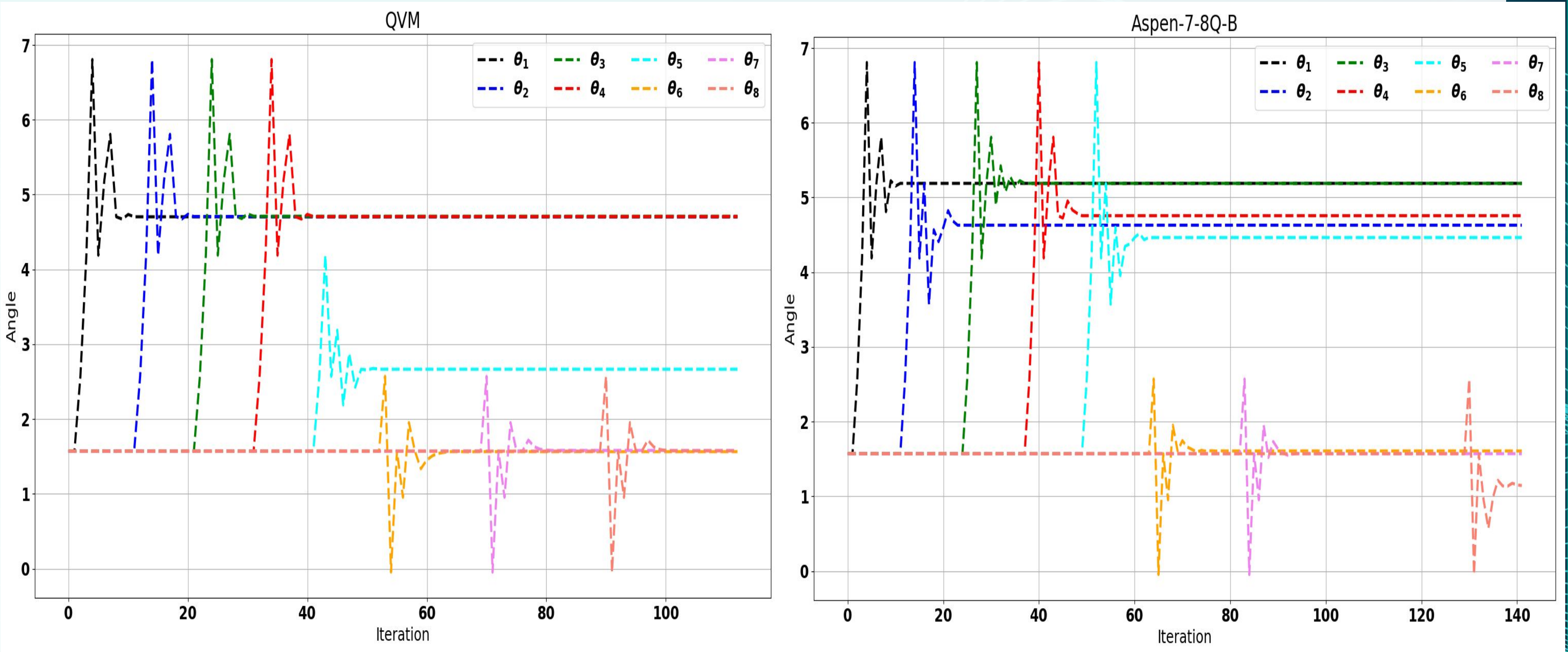


Ising model linear system: 8 qubits

$$A := \frac{1}{\zeta}\left(\eta I + \sum_{j=1}^{n} X_j + J\sum_{j=1}^{n-1} Z_j Z_{j+1}\right)$$

Parameter evolution over optimization iterations





Ising model linear system: 10 qubits

$$A := \frac{1}{\zeta}\left(\eta I + \sum_{j=1}^{n} X_j + J\sum_{j=1}^{n-1} Z_j Z_{j+1}\right)$$

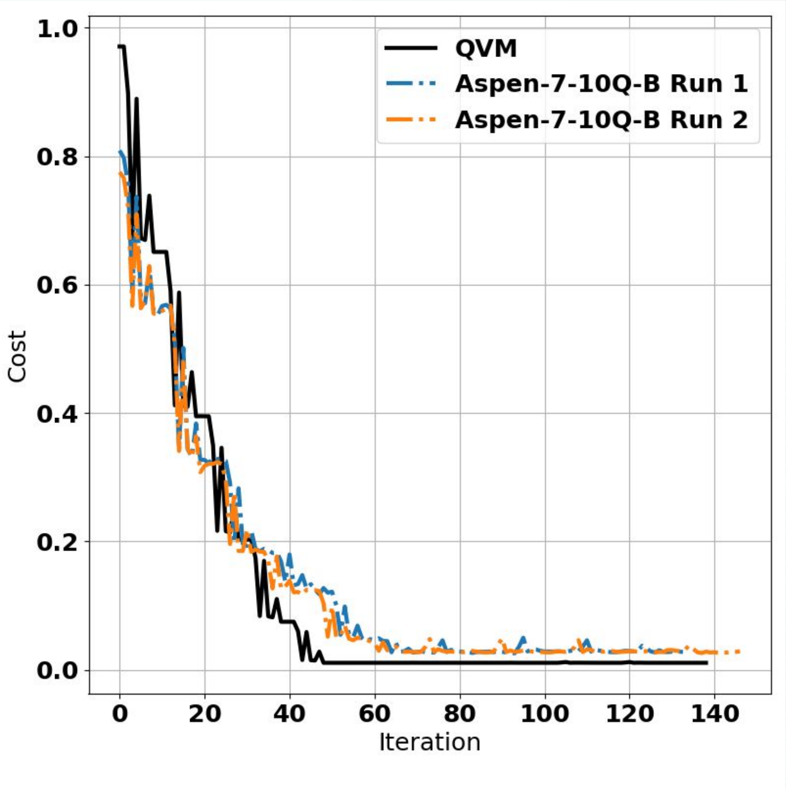
Optimization on hardware for 10 qubits also matches simulator very well.

Able to solve **a** 1024 x 1024 LS.

10 QUBITS

Aspen-7-10Q-B

T1	39.69 µs
T2	32.17 µs
f1QRB	99.8% ± 0.26%
f1Q sim. RB	98.97% ± 0.12%
fActiveReset	99.8%
fRO	96.96%
fCZ	90.63% ± 0.29%



Conclusions

- Product ansatze perform well on Aspen-7 for variational algorithms
  - Up to 10 qubits demonstrated here

Conclusions

- ❑ Product ansatze perform well on Aspen-7 for variational algorithms
  - ❑ Up to 10 qubits demonstrated here
- ❑ Outlook on VQLS:
  - ❑ Bottleneck: how difficult it is to prepare the solution (same for all variational algorithms).

Conclusions

- Product ansatze perform well on Aspen-7 for variational algorithms
  - Up to 10 qubits demonstrated here
- Outlook on VQLS:
  - Bottleneck: how difficult it is to prepare the solution (same for all variational algorithms).
  - VQLS can likely scale on hardware to 100s of qubits for product states.

Conclusions

- Product ansatze perform well on Aspen-7 for variational algorithms
  - Up to 10 qubits demonstrated here

- Outlook on VQLS:
  - Bottleneck: how difficult it is to prepare the solution (same for all variational algorithms).
  - VQLS can likely scale on hardware to 100s of qubits for product states.
  - For general linear systems, a (hardware) efficient ansatz may not be plausible, and VQLS may not perform well.

Conclusions

- Product ansatze perform well on Aspen-7 for variational algorithms
  - Up to 10 qubits demonstrated here

- Outlook on VQLS:
  - Bottleneck: how difficult it is to prepare the solution (same for all variational algorithms).
  - VQLS can likely scale on hardware to 100s of qubits for product states.
  - For general linear systems, a (hardware) efficient ansatz may not be plausible, and VQLS may not perform well.
  - Although there may be many terms in the effective Hamiltonian, truncation + simultaneous measurements provide very good approximations.

Future applications of VQLS

- Solving linear systems has many applications beyond benchmarking:
  - (Partial) differential equations
  - Machine learning
  - Spectral graph theory, etc.

Future applications of VQLS

- Solving linear systems has many applications beyond benchmarking:
  - (Partial) differential equations
  - Machine learning
  - Spectral graph theory, etc.

- With VQLS, we can potentially solve large (2^50 on 50 qubits) linear systems in the (very) near-term **if**:
  - Their solutions **x** can be represented by low-depth ansatze.
  - Training time is not too bad (see previous scaling heuristics).
  - The number of terms in the effective Hamiltonian is not too large.

Future applications of VQLS

- Solving linear systems has many applications beyond benchmarking:
  - (Partial) differential equations
  - Machine learning
  - Spectral graph theory, etc.

- With VQLS, we can potentially solve large (2^50 on 50 qubits) linear systems in the (very) near-term **if**:
  - Their solutions **x** can be represented by low-depth ansatze.
  - Training time is not too bad (see previous scaling heuristics).
  - The number of terms in the effective Hamiltonian is not too large.

This can lead to advantage **if** no classical computer can do the same.

Acknowledgements

Thanks to my co-authors on VQLS:

- Carlos Bravo-Prieto (LANL/Univ. of Barcelona/BSC)
- Marco Cerezo (LANL)
- Yigit Subasi (LANL)
- Lukasz Cincio (LANL)
- Patrick Coles (LANL)

Thanks to Rigetti for Quantum Cloud Services + help with running VQLS, especially to Tom Lubowe for help with running these experiments.

References

[1] Aram Harrow, Avinatan Hassidim, Seth Lloyd, Quantum algorithm for solving linear systems of equations, Phys. Rev. Lett. vol. 15, no. 103, pp. 150502 (2009).

[2] Leonard Wossnig, Zhikuan Zhao, Anupam Prakash, A quantum linear system algorithm for dense matrices, Phys. Rev. Lett. vol. 120, pp. 050502 (2018).

[3] Ryan LaRose, [Talk] Quantum singular value estimation and its applications, IBM Quantum Research Seminar, 2019.

VQLS Paper: https://arxiv.org/abs/1909.05820

My code for these results: https://github.com/rmlarose/rigettivqls