



**LIRMM**

# Enabling Parallel Program Execution on NISQ Hardware

Quantum Computer Systems Lecture Series

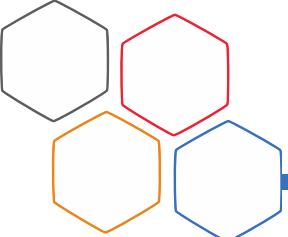
Siyuan Niu

University of Montpellier

[siyuan.niu@lirmm.fr](mailto:siyuan.niu@lirmm.fr)

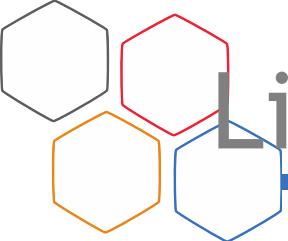
28/07/2022





# Outline

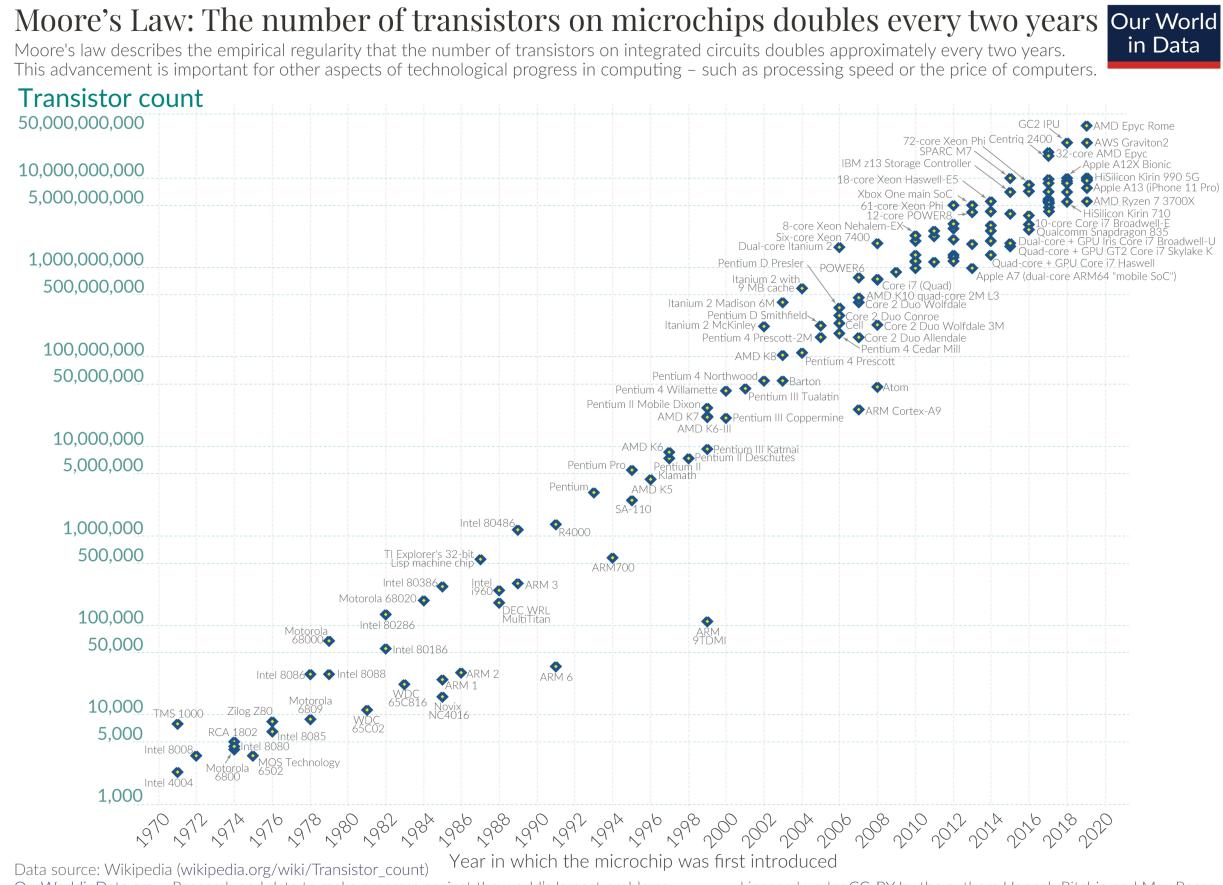
- Quantum computing: progress and perspectives
- NISQ hardware limitations
- Parallel program execution
  - Superconducting quantum hardware
  - Quantum annealing
- Conclusion



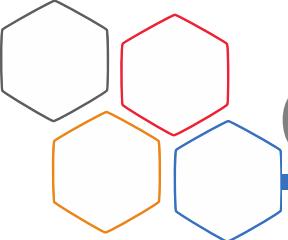
# Limitation of classical computers



December 1947  
First transister was  
born in Bell lab



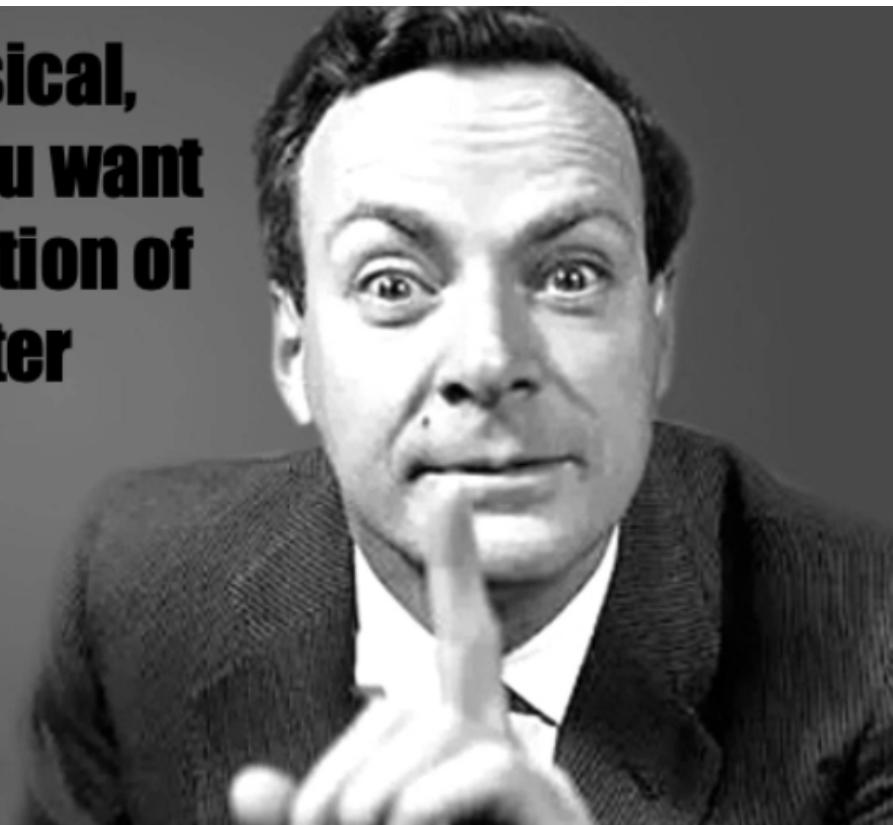
# Figures from wikipedia



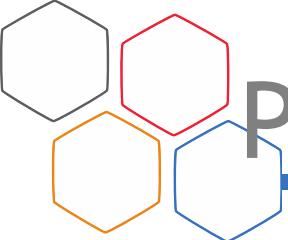
# Quantum computing - new paradigm

**Nature isn't classical,  
dammit, and if you want  
to make a simulation of  
nature, you'd better  
make it quantum  
mechanical!**

***Simulating Physics  
with Computers***  
**R. Feynman (1981)**

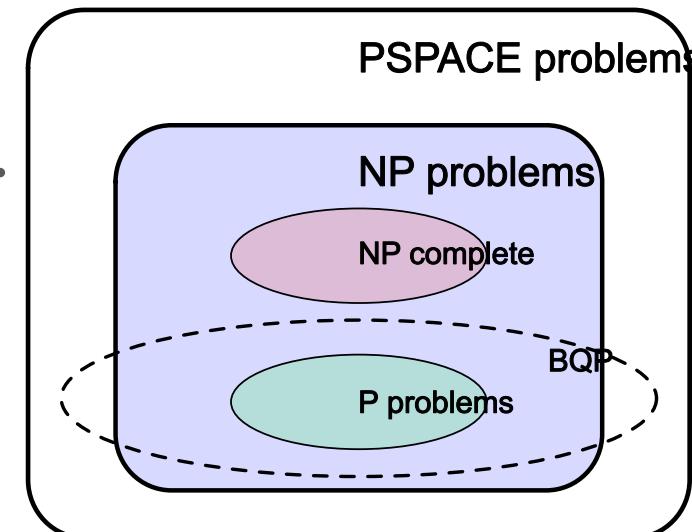


<https://medium.com/le-lab-quantique/what-are-quantum-computers-good-for-a7fa451969f>

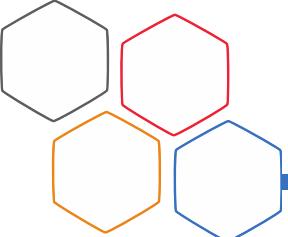


# Problems solved by QC efficiently?

- “Efficiently” usually means that a problem is solvable in polynomial time.
- Classes of decision problems
  - P: conventional computers can solve it in polynomial time
  - BQP: quantum computers can solve it in polynomial time
  - NP-Complete: very hard to solve it with conventional computers in polynomial time
- Some problems are thought not to be in P but in BQP
  - Integer factorization -> **Shor's algorithm**
- Does BQP include NP-complete?
  - Likely to be NO



Too many qubits are required!



# Where are we?

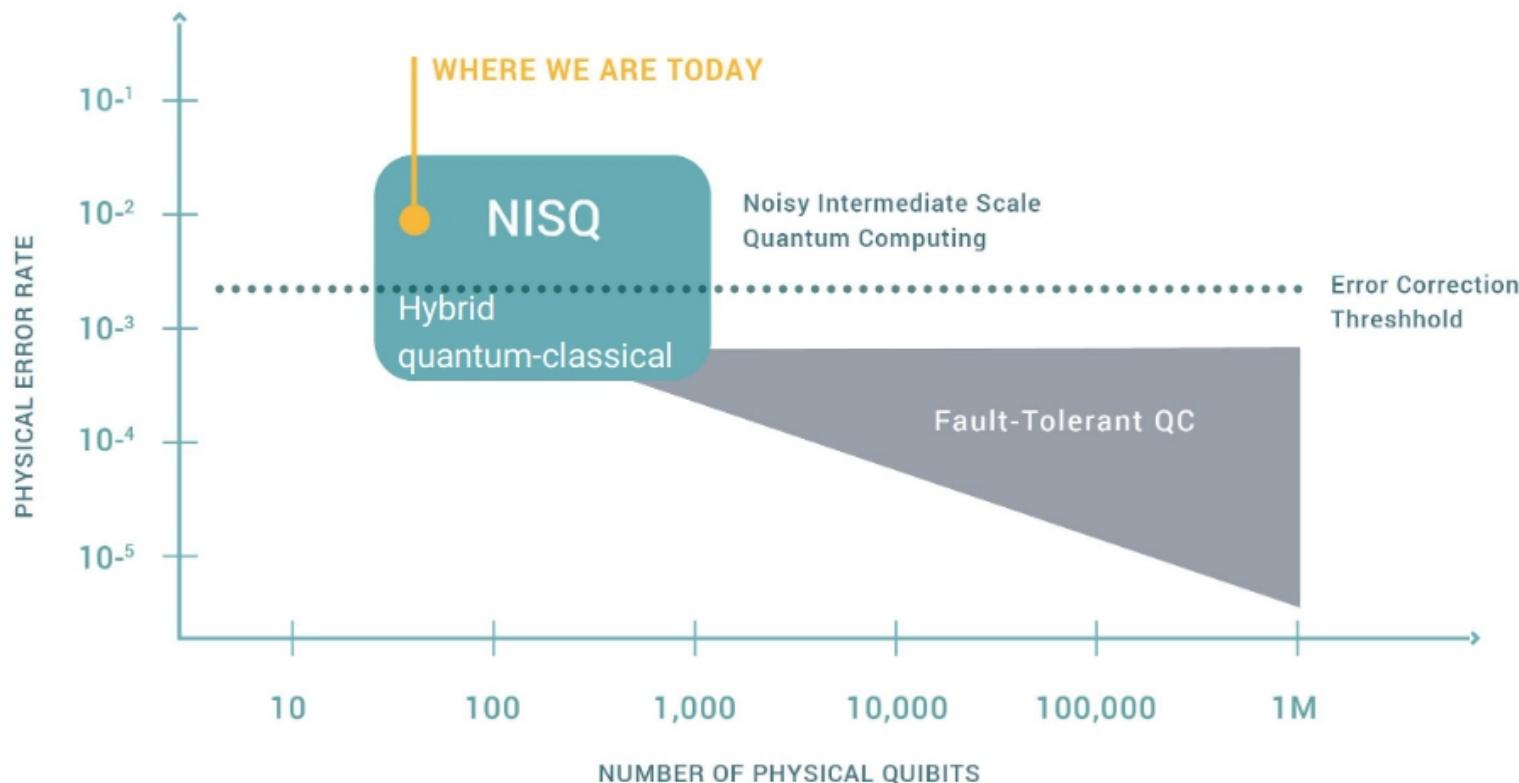
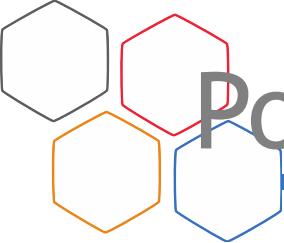
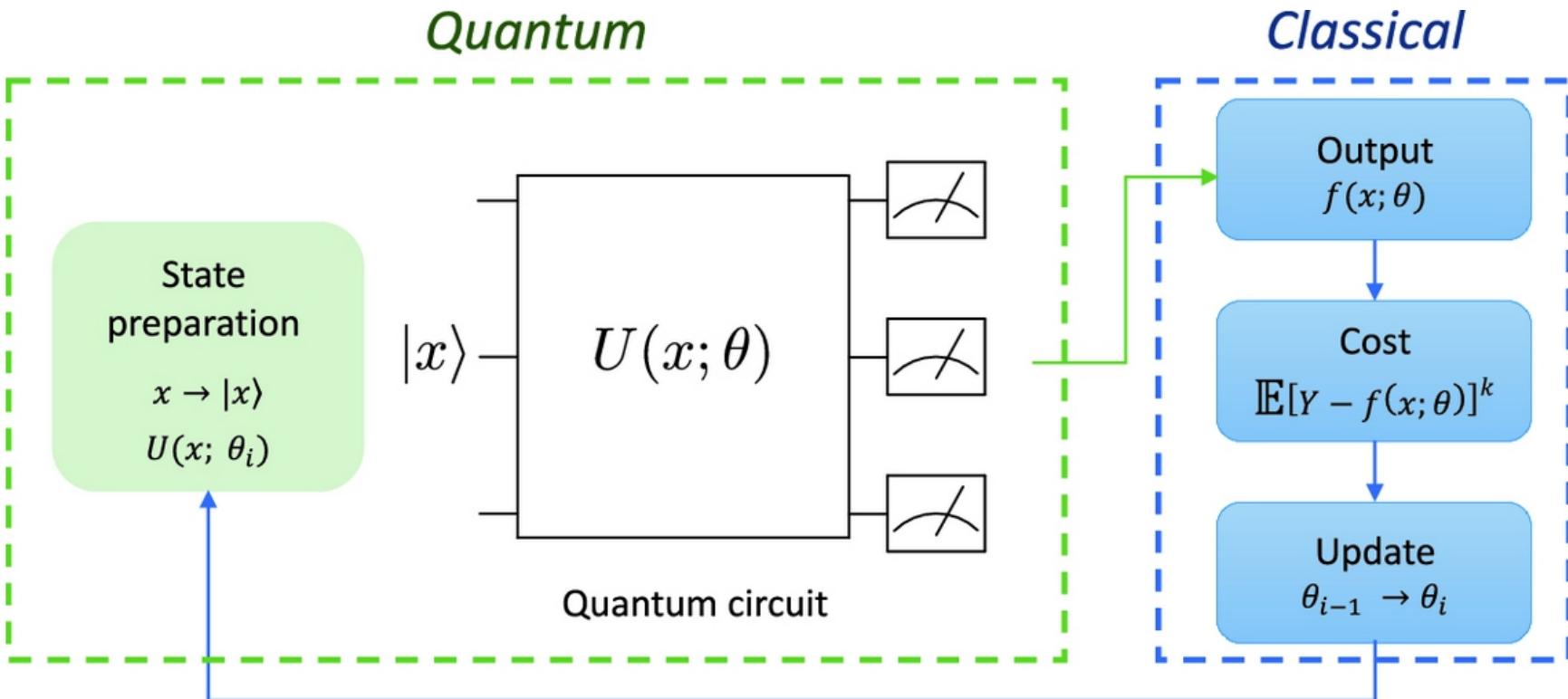


Figure from Rigetti computing

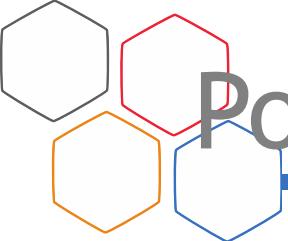


# Potential advantages in NISQ era?

## Variational Quantum Algorithms (VQAs)

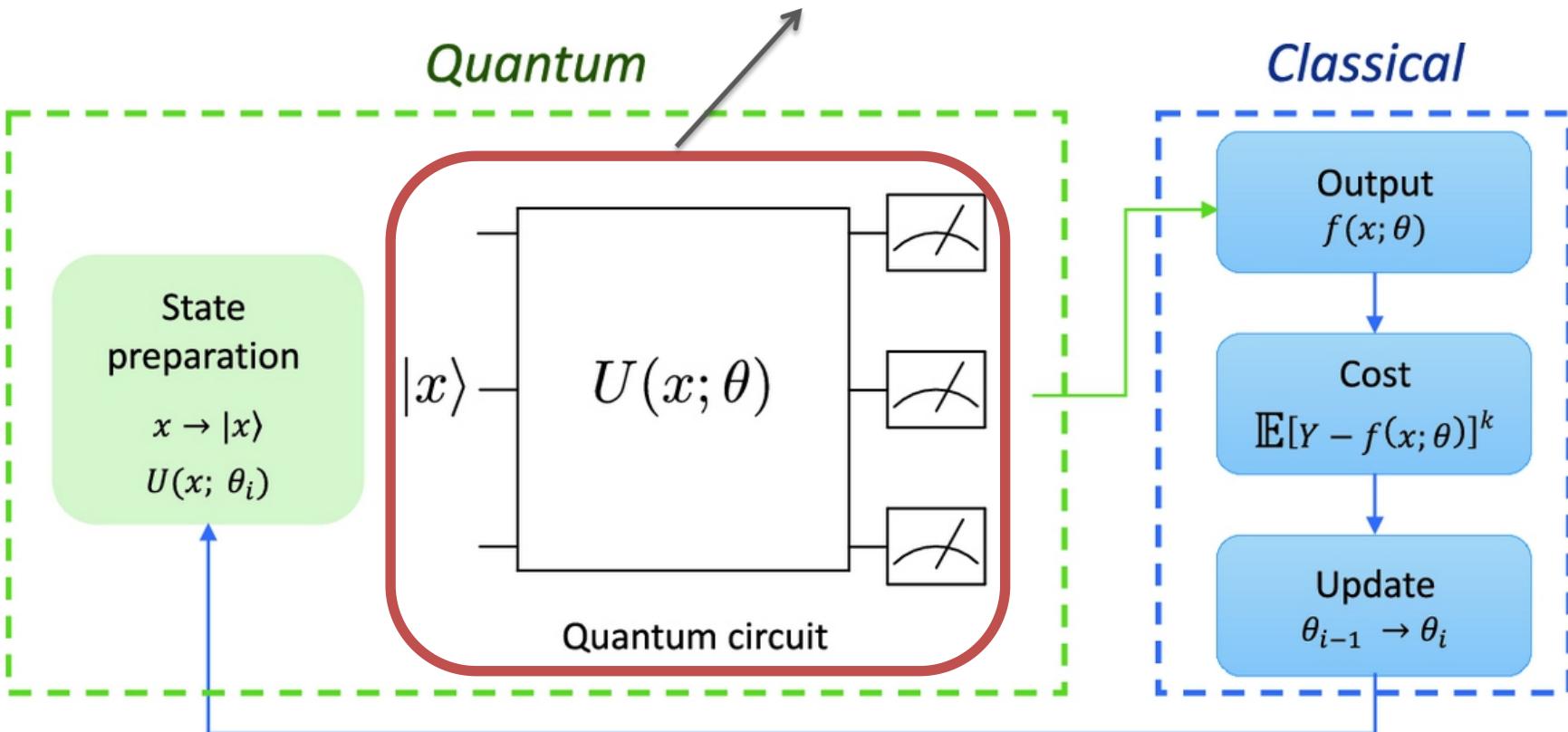


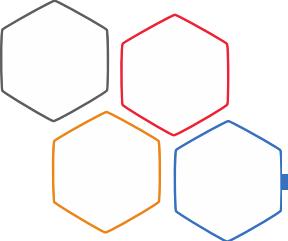
<https://github.com/ericardomuten/qml-vqa-library>



# Potential advantages in NISQ era?

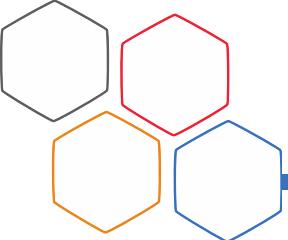
Shallow-depth circuit  $\rightarrow$  NISQ hardware!





# VQAs and applications

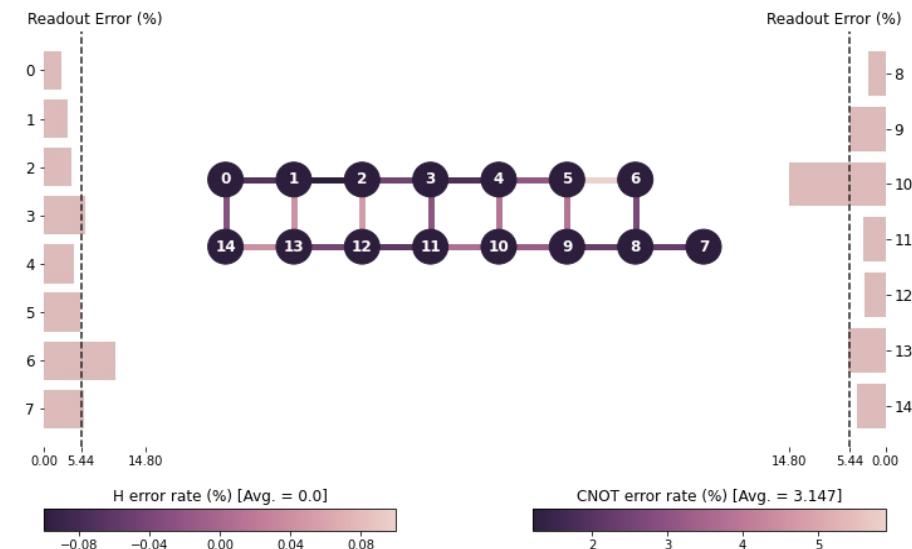
- **Variational Quantum Eigensolver (VQE)**
  - Find the ground state energy of a molecular Hamiltonian.
- **Quantum Approximate Optimization Algorithm (QAOA)**
  - Solve the combinatorial optimization problem.
- **Variational Quantum Linear Solver (VQLS)**
  - Solve systems of linear equations.
- **Variational Quantum Classifier (VQC)**
  - Quantum machine learning algorithm for classification problem.

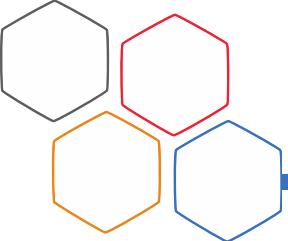


# Limitations in NISQ era

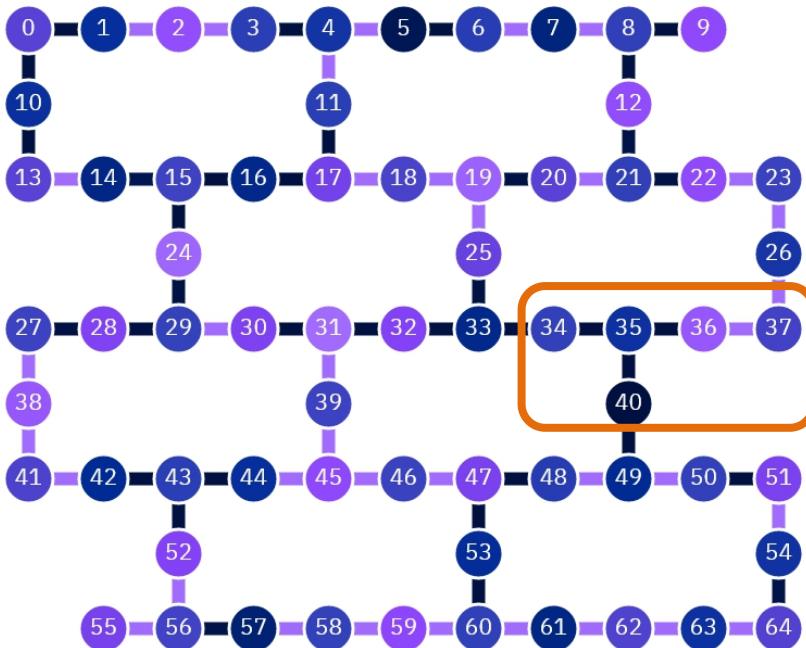
- Nearest-neighbor connectivity.
  - Superconducting
  - Silicon quantum dot
- Noisy quantum operations.
  - Readout error
  - 1-Q, 2-Q error
  - Different gate error across qubits

ibmq\_16\_melbourne Error Map



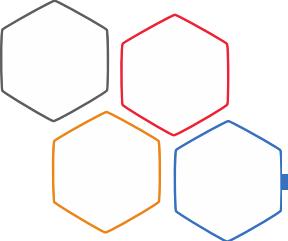


# Limitations in NISQ era

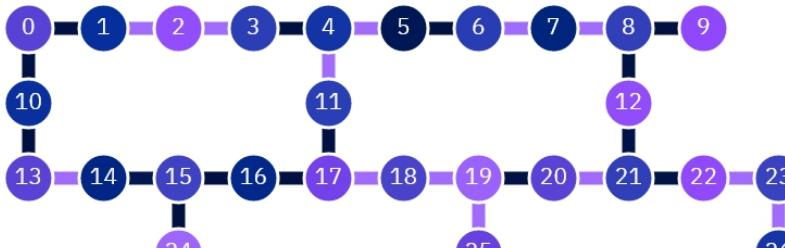


- Execute a 5-qubit circuit on IBM Q 65 Manhattan.
- Hardware utilization: **8%**
- Total pending jobs: **1038 jobs**

- Only **small circuits with shallow depth** can get reliable results.
- Growing demand to access quantum hardware leads to long **waiting time**.

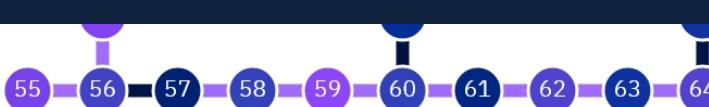


# Limitations in NISQ era

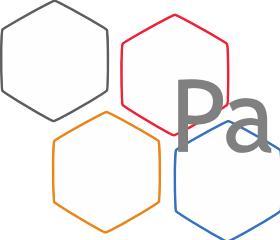


- Execute a 5-qubit circuit on IBM Q 65 Manhattan.
- Hardware throughput: 8%

How do we use quantum hardware more efficiently while maintaining the output fidelity?

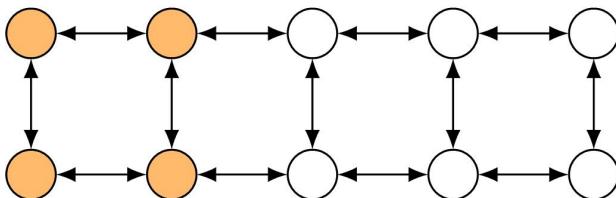


- Only small circuits with shallow depth can get reliable results.
- Growing demand to access to quantum hardware leads to long waiting time.

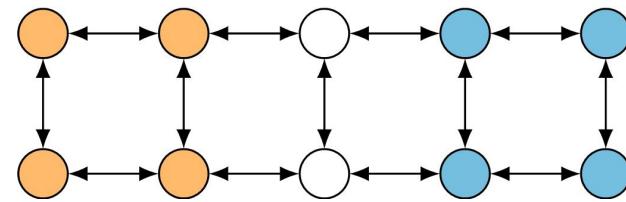


# Parallel program execution in NISQ era

QC1 QC2

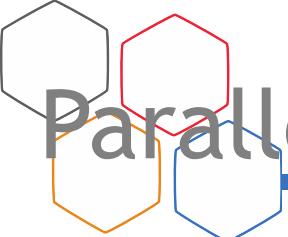


(a) Run one 4-qubit circuit on a 10-qubit chip. Hardware usage is **40%**.

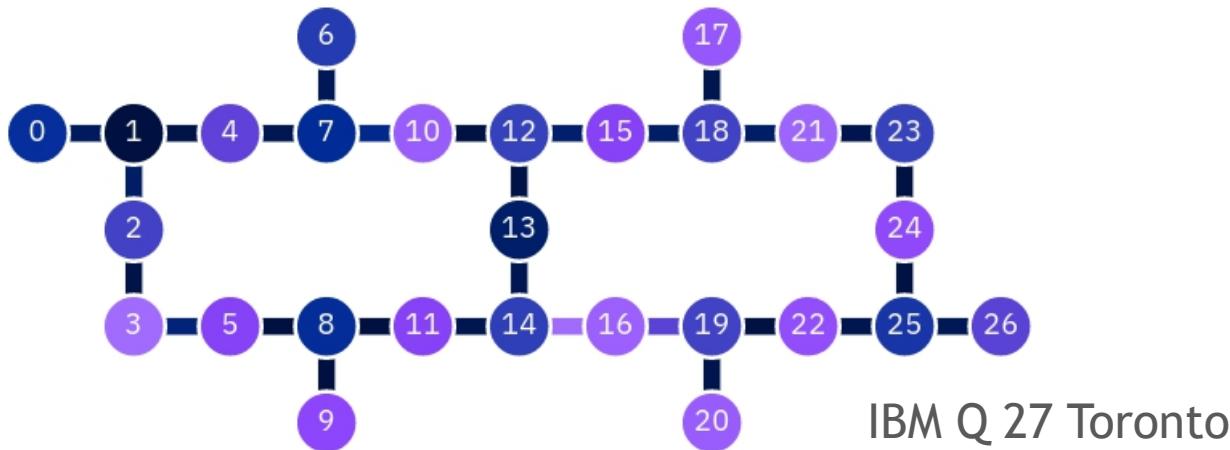


(b) Run two 4-qubit circuits on a 10-qubit chip. Hardware usage is **80%**.

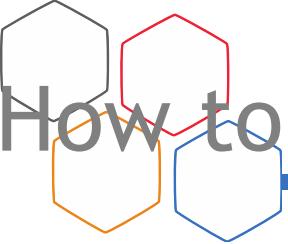
- Key idea: Execute multiple programs on a quantum chip simultaneously.
- Hardware demonstration
  - IBM superconducting quantum hardware.
  - DWAVE annealing machine.
- Advantages
  - Improve the hardware usage.
  - Reduce the total program runtime.



# Parallel program execution on SC hardware

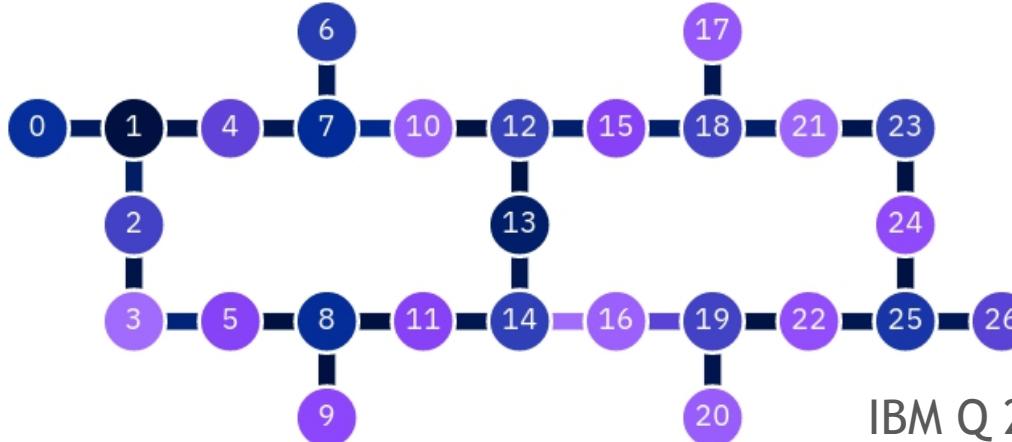


- **Challenges**
  - How to find reliable partitions for each circuit?
  - How to mitigate crosstalk impact?
  - How to trade-off between hardware utilization and circuit fidelity?



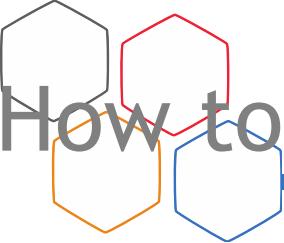
# How to find reliable partitions for each circuit?

Find a reliable subset of hardware



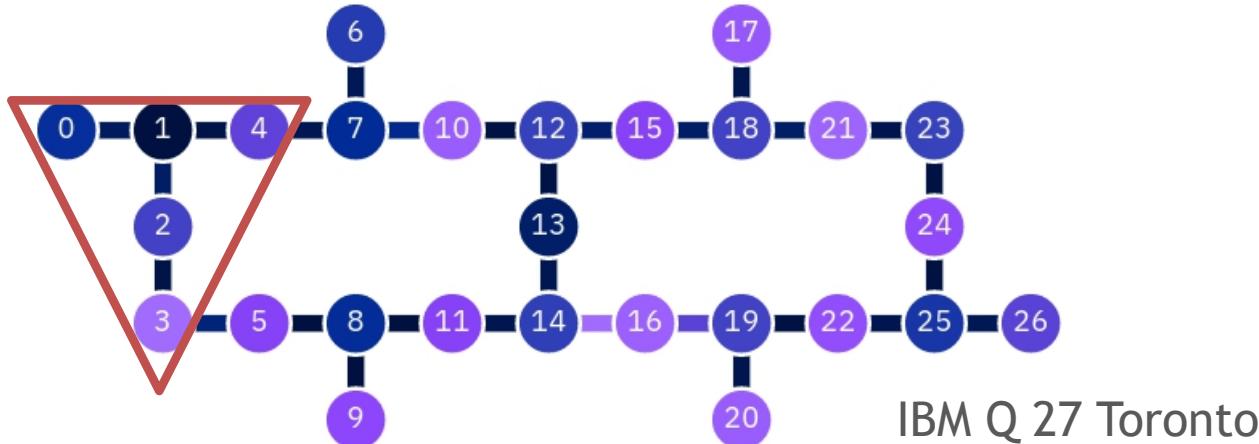
IBM Q 27 Toronto

- **Connectivity** of the subset
  - Graph diameter
- **Error rate** of the subset
  - Readout error rate
  - 1Q, 2Q error rate
  - #gate operations
- **Interaction** across circuits
  - Crosstalk
  - Globally good solution for all circuits

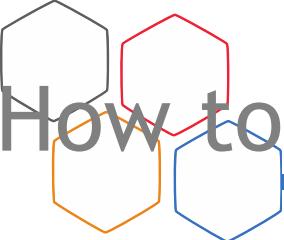


# How to find reliable partitions for each circuit?

## Find a reliable subset of hardware



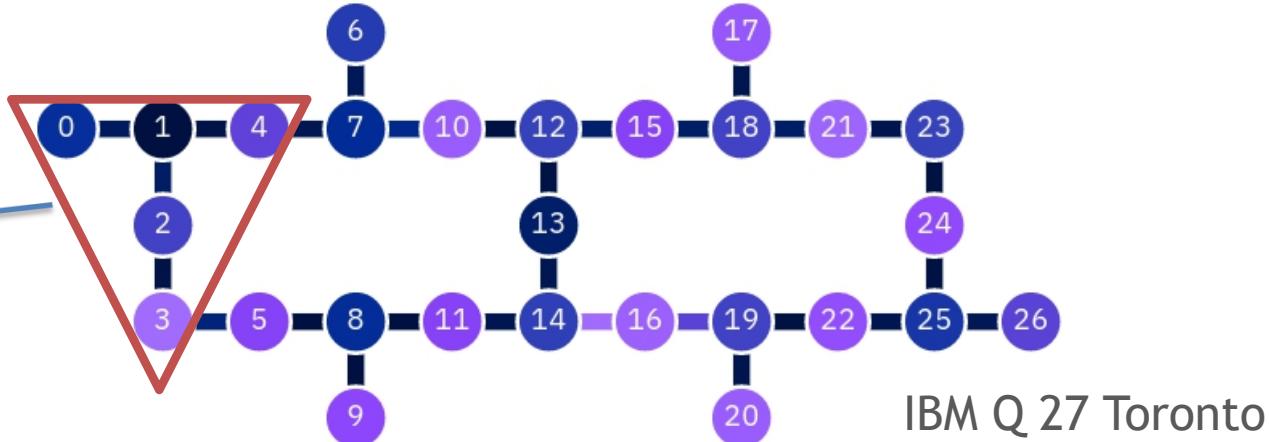
- **Connectivity** of the subset
  - Graph diameter
- **Error rate** of the subset
  - Readout error rate
  - 1Q, 2Q error rate
  - #gate operations
- **Interaction** across circuits
  - Crosstalk
  - Globally good solution for all circuits



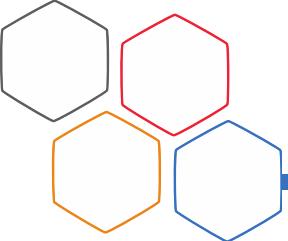
# How to find reliable partitions for each circuit?

## Find reliable subsets of circuits

Good candidate

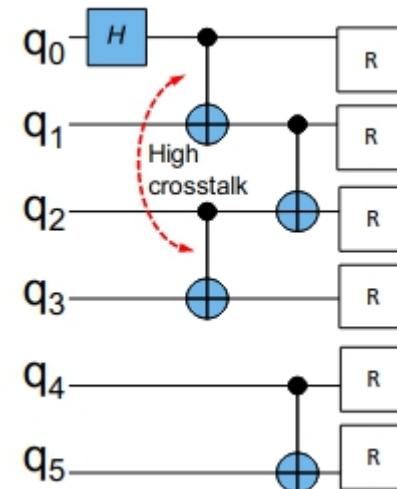
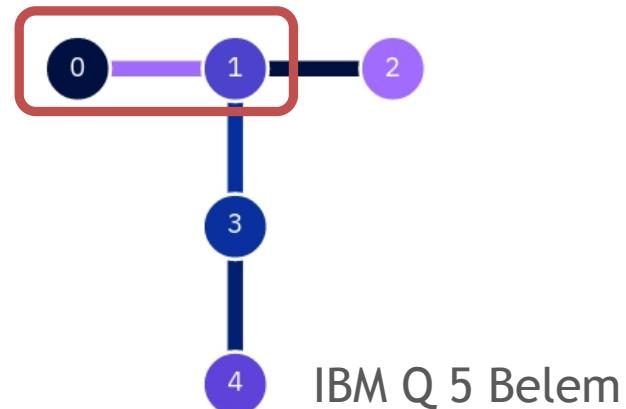


- **Connectivity** of the subset
  - Graph diameter
- **Error rate** of the subset
  - Readout error rate
  - 1Q, 2Q error rate
  - #gate operations
- **Interaction** with other circuits
  - Crosstalk
  - Globally good solution for all circuits

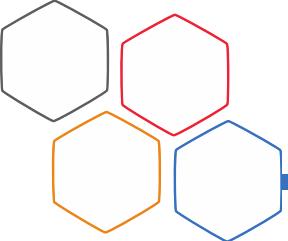


# Introduction to crosstalk error

- Local crosstalk
  - Always on ZZ interaction
  - Driven crosstalk from 2Q gate
- Global crosstalk
  - Always on ZZ from coupled qubit
  - Simultaneous gate operations

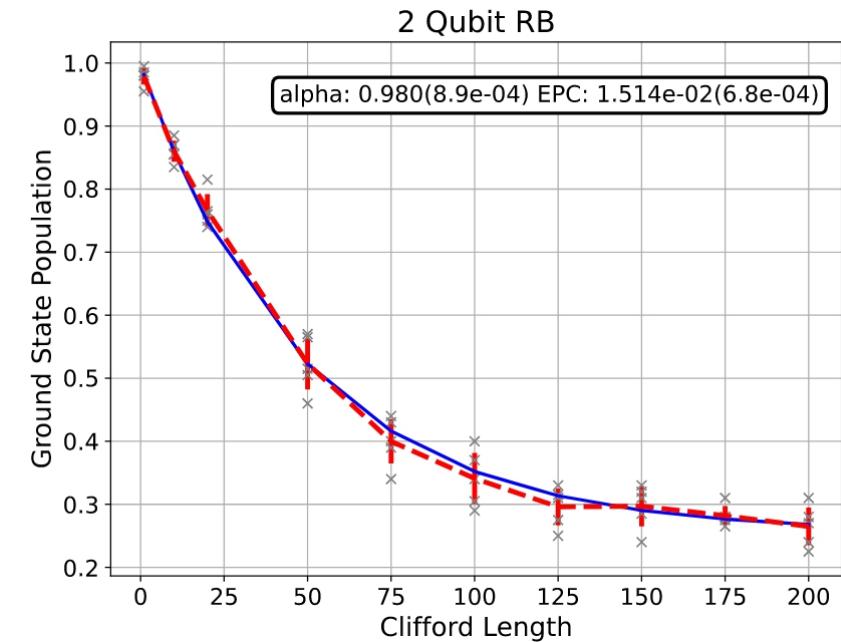
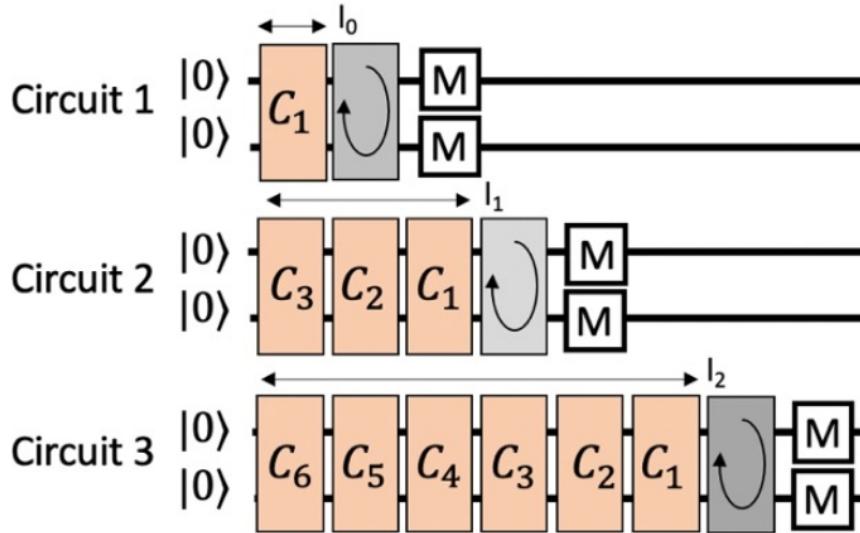


Murali et al. ASPLOS 2020.



# How to characterize crosstalk?

- Simultaneous randomized benchmarking (SRB)
  - Quantify the impact of parallel instructions.
- Randomized benchmarking (RB)



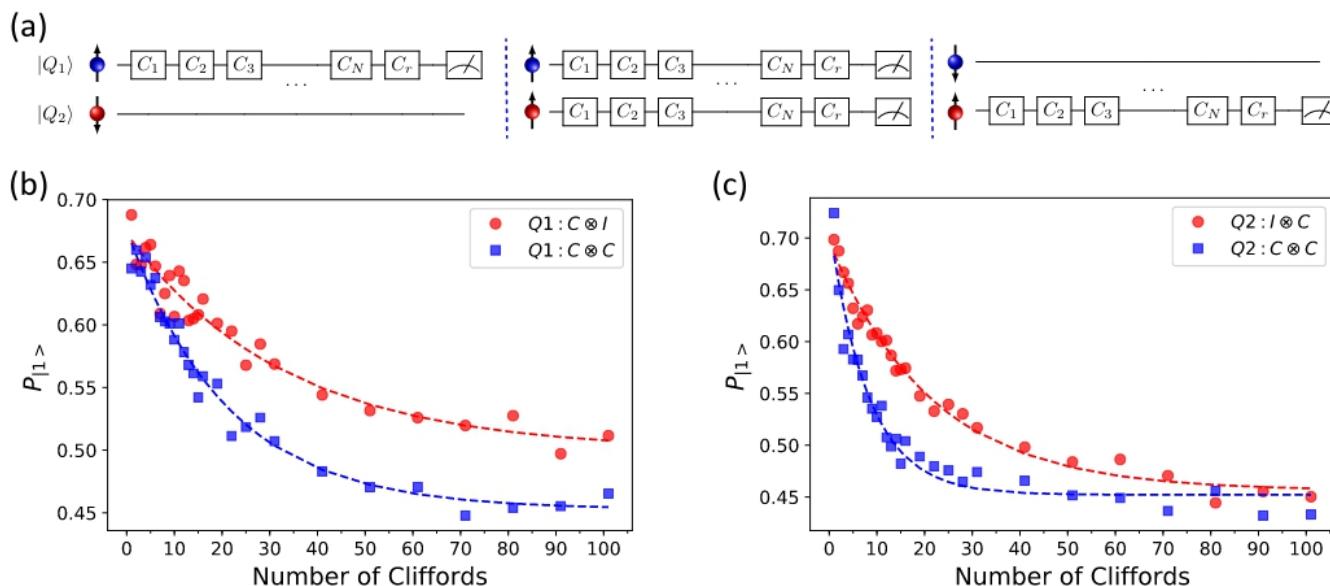
- Scalable method (in the number  $n$  of hardware qubits)
- Insensitive to SPAM errors
- Only give the average level of error

McKay et al. PRL, 2019.

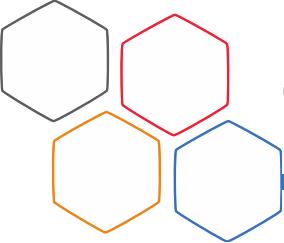
<https://thequantumaviary.blogspot.com/2021/01/how-to-measure-errors-on-ibm-quantum.html>



# Simultaneous randomized benchmarking

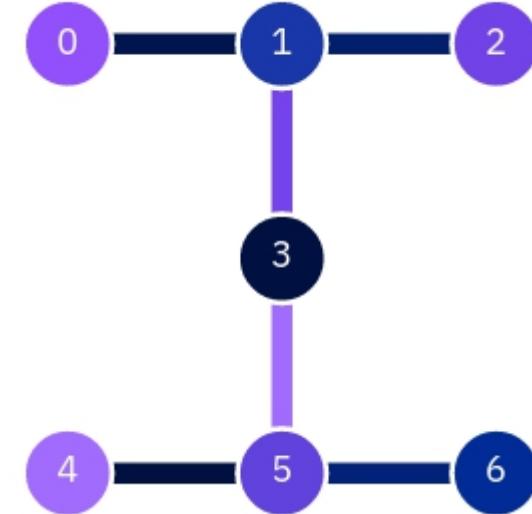
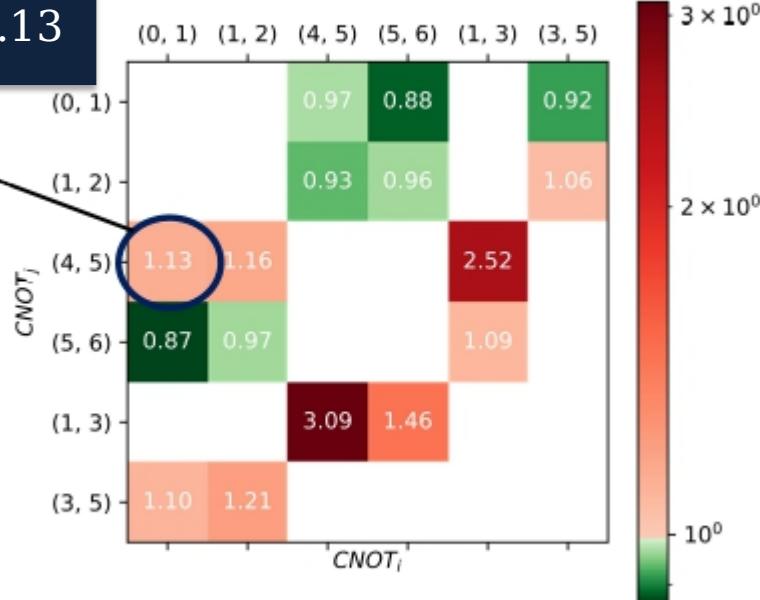


- Perform Randomized Benchmarking (RB) on gates separately
  - Independent error:  $E(g_i)$ ,  $E(g_j)$
  - Apply SRB on both qubit pairs
    - Correlated error:  $E(g_i|g_j)$ ,  $E(g_j|g_i)$
  - If  $E(g_i|g_j) > E(g_i)$  or  $E(g_j|g_i) > E(g_j) \Rightarrow$  Crosstalk!



# SRB result on IBM quantum chip

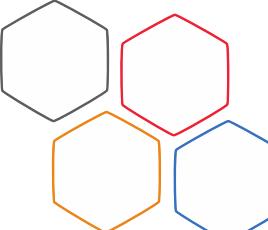
$$\frac{E(gi|gj)}{E(gi)} = 1.13$$



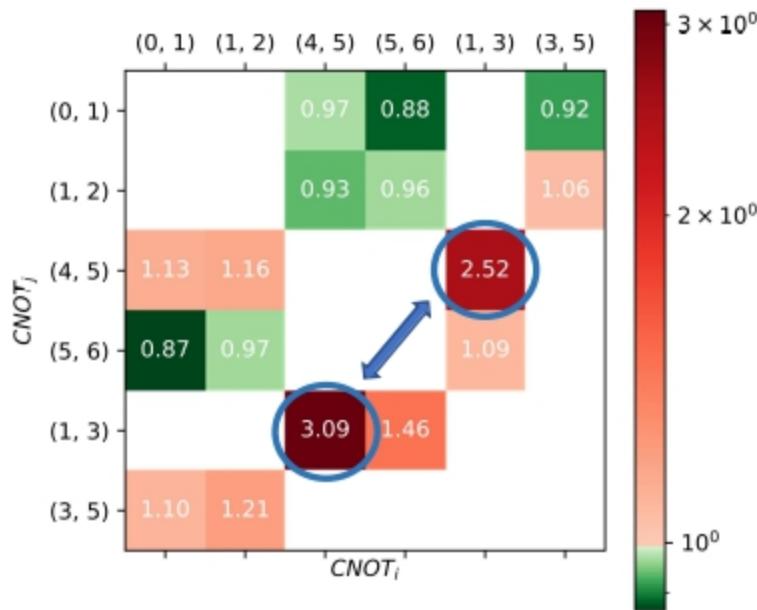
IBM Q 7 Casablanca

- Correlated error / independent error
- SRB is only performed on CNOTs that can be executed in parallel (no share qubits).
- Blank spot: No SRB experiment.

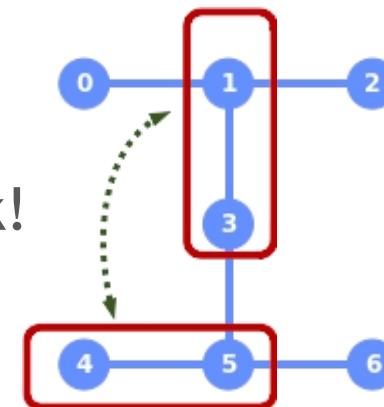
Niu et al. ISVLSI, 2021.



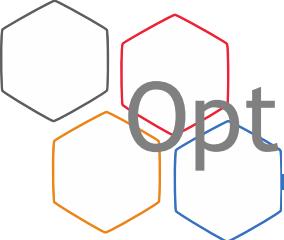
# SRB result on IBM quantum chip



High crosstalk!

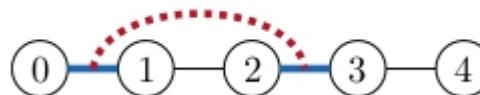


IBM Q 7 Casablanca

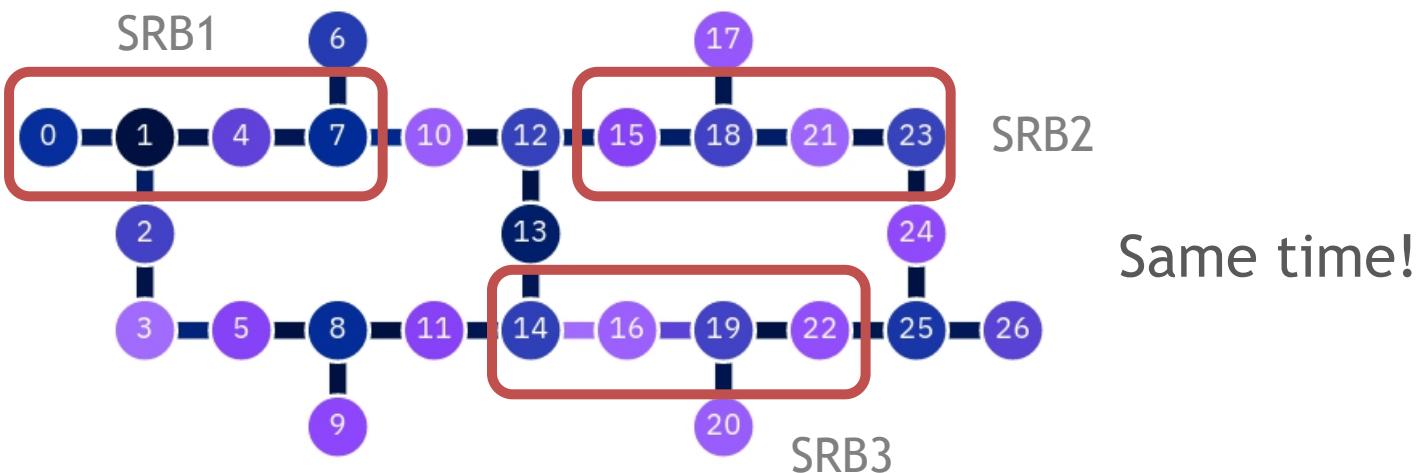


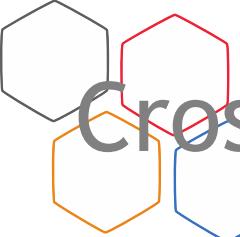
# Optimizations for SRB characterization

- Characterize only 1-hop pairs



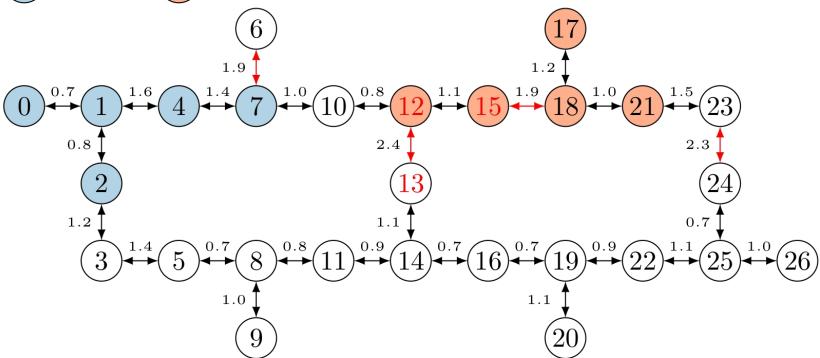
- Parallelize SRB experiments of multiple qubit pairs.
- Characterize high crosstalk pairs only.





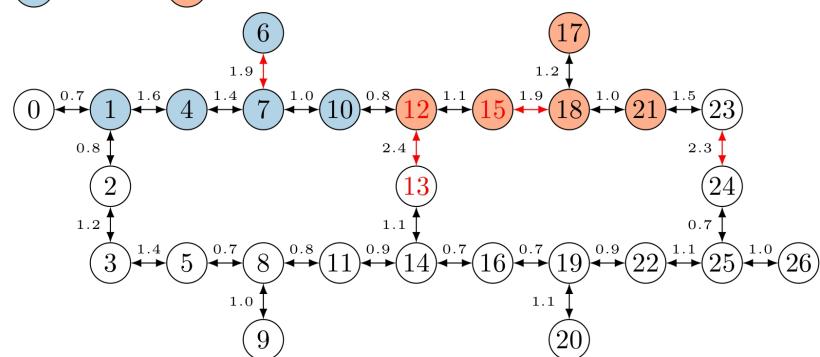
# Crosstalk on parallel program execution

P1      P3

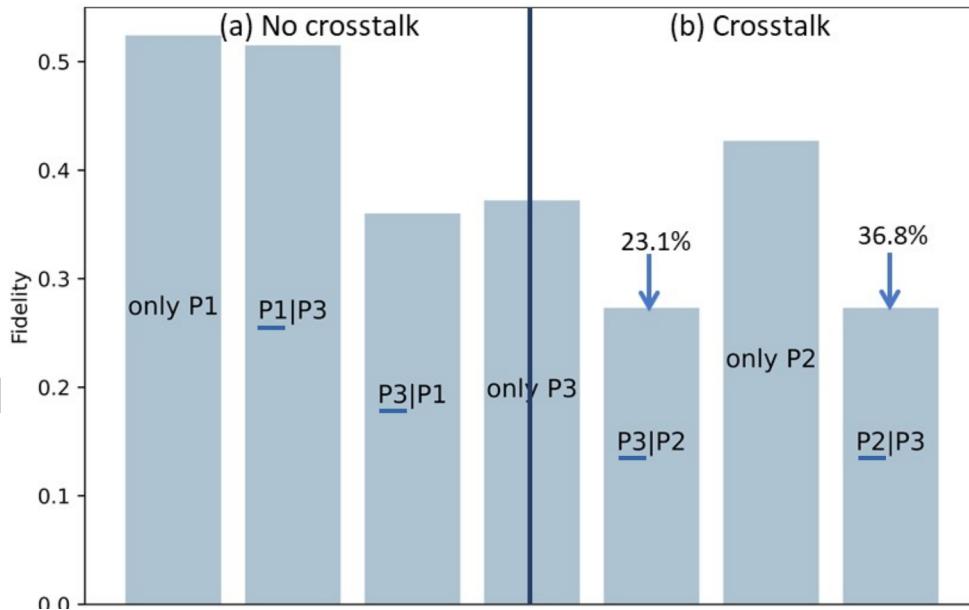


(a) No crosstalk between partition P1 and partition P2.

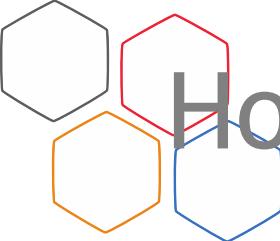
P2      P3



(b) Crosstalk exists between partition P2 and partition P3.

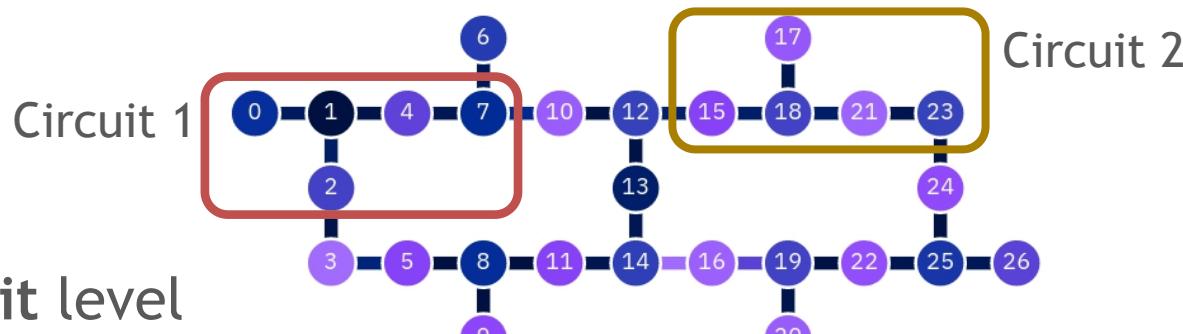


**Results:**  
Crosstalk decrease the fidelity of P3 and P2 by **23.1%** and **36.8%** respectively.

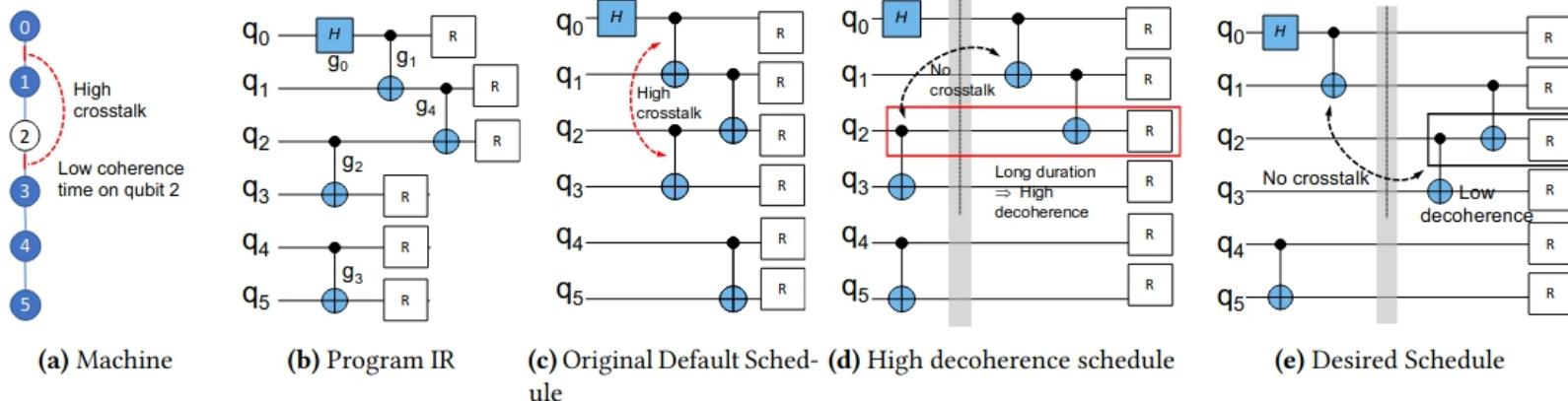


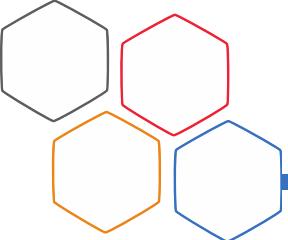
# How to mitigate crosstalk impact?

- At partition level
  - Avoid adjacent partitions

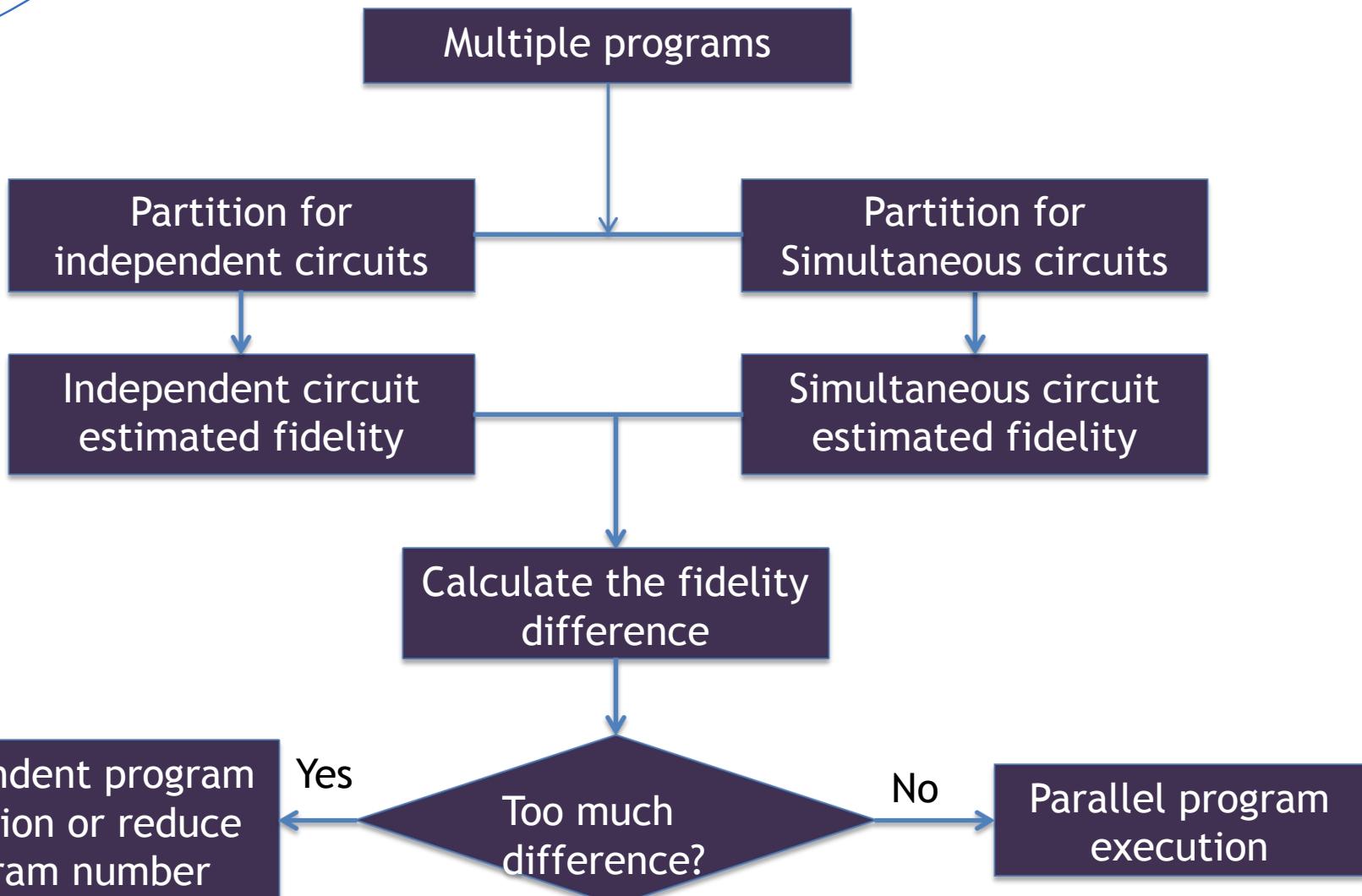


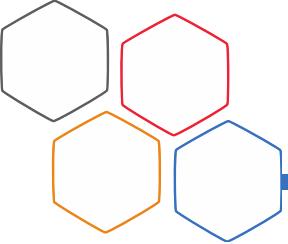
- At circuit level
  - Clever scheduling





# Trade-off between hardware usage and circuit fidelity?

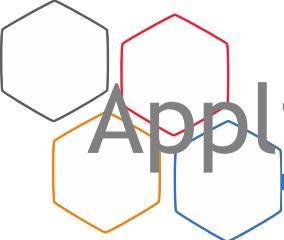




# Litterature

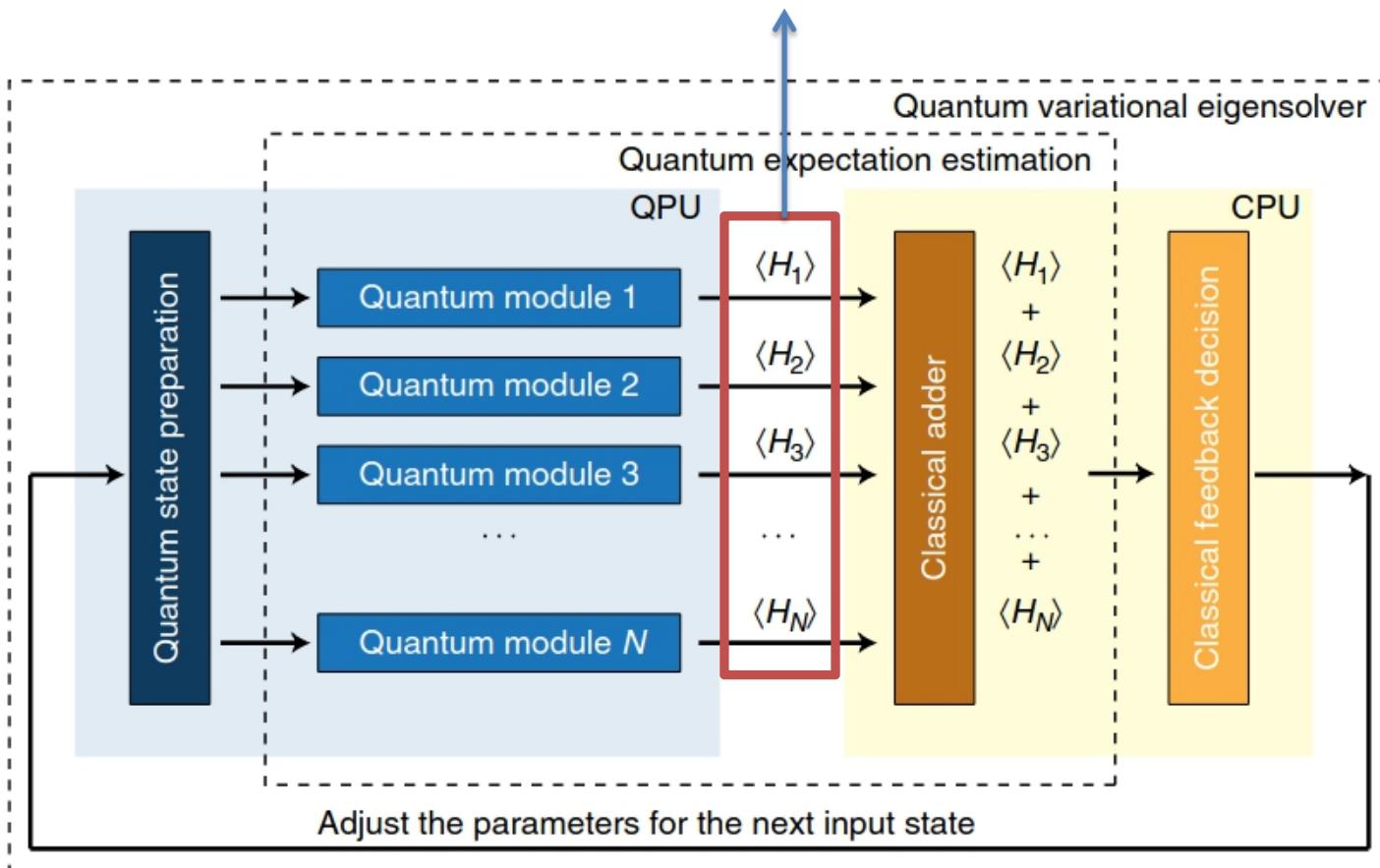
---

- Das et al. “A case for Multi-programmng Quantum Computers”, MICRO, 2019.
- Liu et al. “QuCloud”. HPCA 2021.
- Niu et al. “How parallel circuit execution can be useful for NISQ computing?”, DATE 2022.
- Ohkura et al. “Simultaneous execution of quantum circuits on current and near-future NISQ systems”, IEEE TQE, 2022.

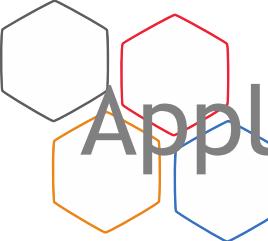


# Application: reduce #measurement circuits

Where parallel program execution can help.



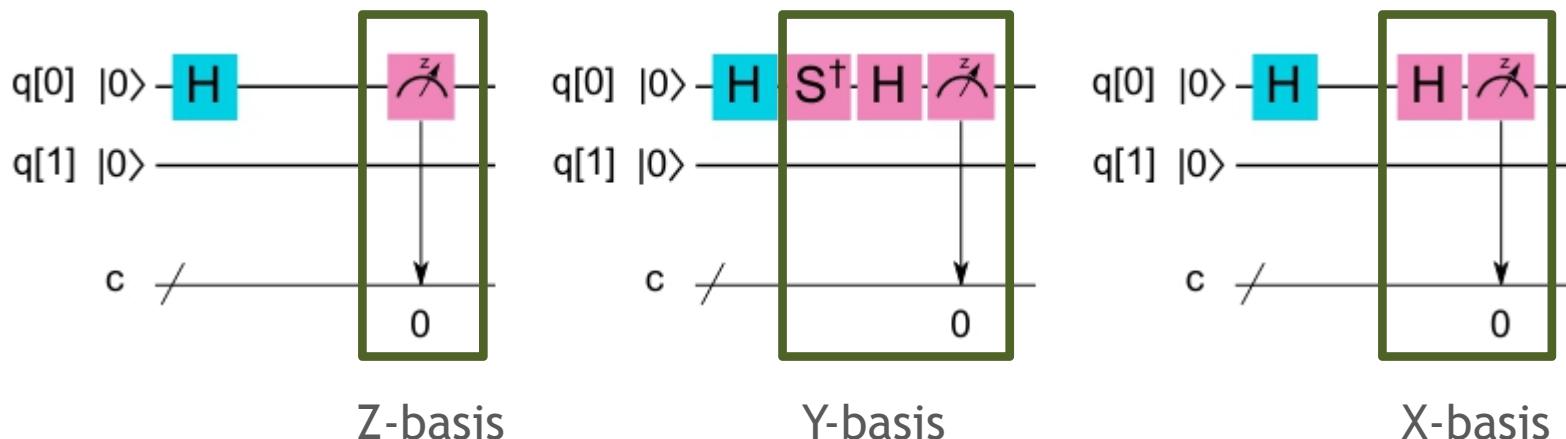
Peruzzo et al. Nature communicatons. 2014.



# Application: reduce #measurement circuits

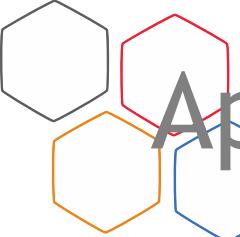
$$H = a^*ZI + b^*IZ + c^*XX + d^*YY$$

$$\langle H \rangle = a^*\langle ZI \rangle + b^*\langle IZ \rangle + c^*\langle XX \rangle + d^*\langle YY \rangle$$



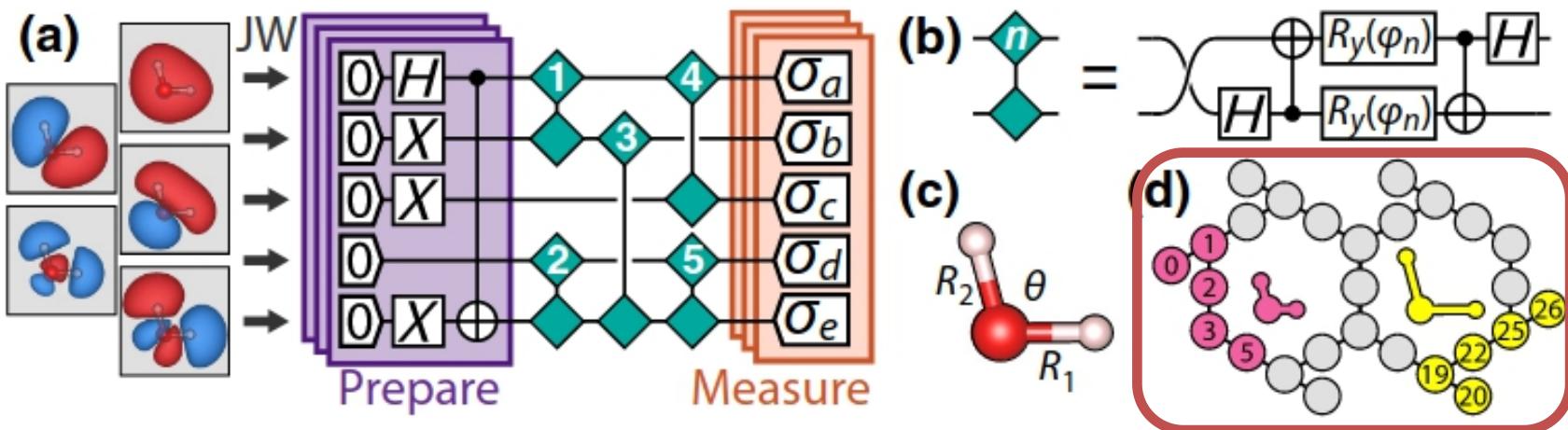
- Three measurement circuits to calculate expectation value once.
- Reduce #measurement circuits by parallel program execution!

Adedoyin et al. arxiv.1804.03719

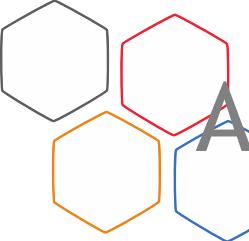


# Application: entanglement forging

Simulate a given quantum system using only half as many qubits on a quantum computer.

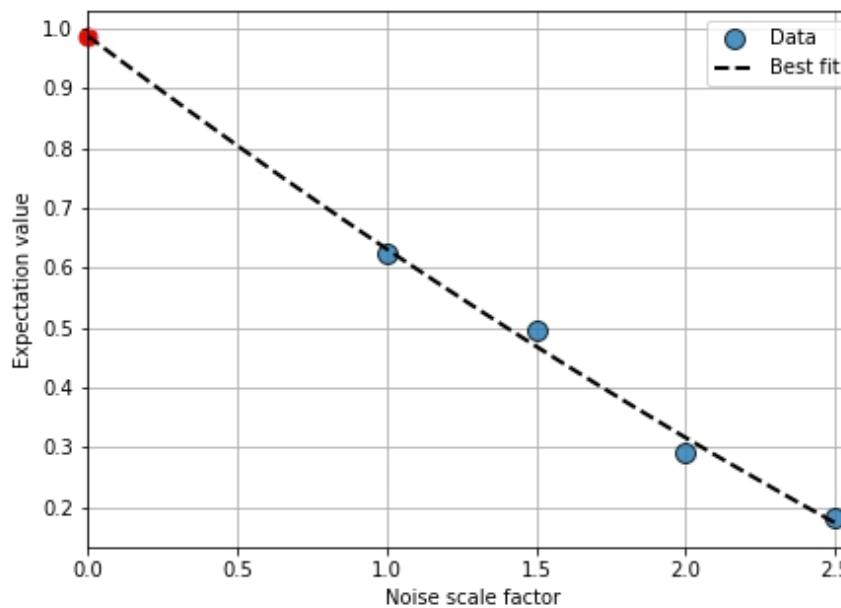


Eddins et al. PRX 2022.

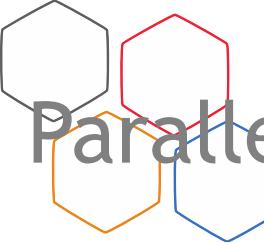


# Application: zero noise extrapolation

- Zero noise extrapolation error mitigation method.
  - Noise-scaling
  - Extrapolation
- Limitation: overhead of executing one circuit multiple times with various depths for noise-free extrapolation.
- Execute circuits with different depths in parallel to realize error mitigation in one big circuit.



Li et al. PRX 2017.

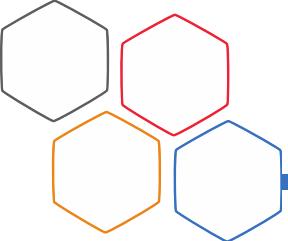


# Parallel program execution in quantum annealing

- Quantum annealing
  - Find the global minimum of a combinatorial optimization problem.

$$\mathcal{H}_{ising} = \underbrace{-\frac{A(s)}{2} \left( \sum_i \hat{\sigma}_x^{(i)} \right)}_{\text{Initial Hamiltonian}} + \underbrace{\frac{B(s)}{2} \left( \sum_i h_i \hat{\sigma}_z^{(i)} + \sum_{i>j} J_{i,j} \hat{\sigma}_z^{(i)} \hat{\sigma}_z^{(j)} \right)}_{\text{Final Hamiltonian}}$$

- DWAVE machine
  - Implement quantum annealing on its hardware

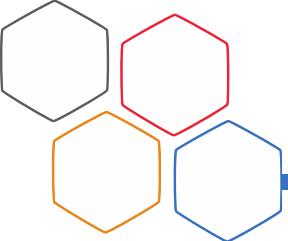


# DWAVE machines

Large number of qubits!

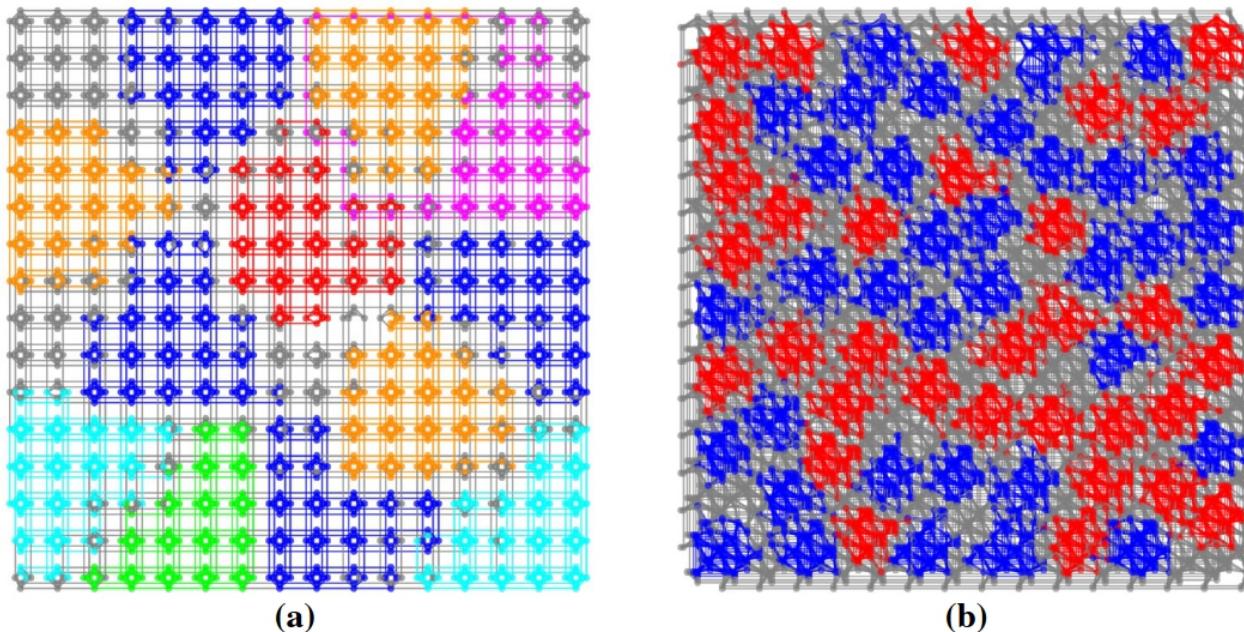
		2000Q	Advantage
<b>Performance</b>			
Better Solutions (Satisfiability problems)	--	3x more often than 2000Q	
Time-To-Solution (3D lattice problems)	--	10x faster than 2000Q	
<b>Annealing Quantum Processor Design</b>			
Qubits	2000+	5000+	
Couplers	6000+	35000+	
Couplers Per Qubit	6	15	
<b>Topology</b>			
Graph	Chimera	Pegasus	
Graph Size	C16	P16	
Connectivity	Degree 6	Degree 15	

Limited connectivity!

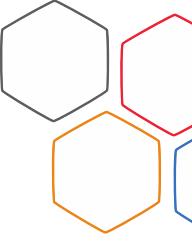


# Parallel quantum annealing

Key challenge: Find **minor embeddings** for multiple QUBO problems

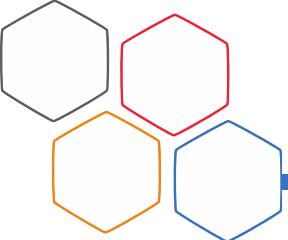


Clique of size 20 embedded 12 times into the hardware of DW 2000Q, which uses Chimera graph topology (a), and 68 times into DW Advantage, which has a Pegasus hardware graph (b).



# Conclusion

- Parallel program execution can help improve the hardware usage and reduce the time for solving problems / total circuits runtime.
- Parallel program execution can be useful on sc hardware
  - VQE algorithm
    - Reduce the number of measurement circuits
    - Entanglement forging
  - Zero noise extrapolation
    - Realize error mitigation in one big circuits
- Parallel program execution can be useful for quantum annealing
  - Solve multiple QUBO problems at the same time



# Ackowledgement

Thanks for your attention!

