Steps Toward a Quantum Advantage in Combinatorial Optimization

Maxime Dupont — mdupont@rigetti.com

In collaboration with NASA, USRA, UC Berkeley, and Berkeley Lab

M. S. Alam, N. Didier, B. Evert, D. Feng, S. Grabbe, S. Hadfield, M. J. Hodson, S. Jeffrey, P. A. Lott, F. B. Maciejewski, Joel E. Moore, M. J. Reagor, E. G. Rieffel, B. Sundar, D. Venturelli, Z. Wang, and Y. Yamaguchi

Supported by DOE and DARPA

Some of this material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under agreement No. HR00112090058

A. Combinatorial optimization

What is it? Quantum algorithms?

B. Quantum advantage requires beating classical…

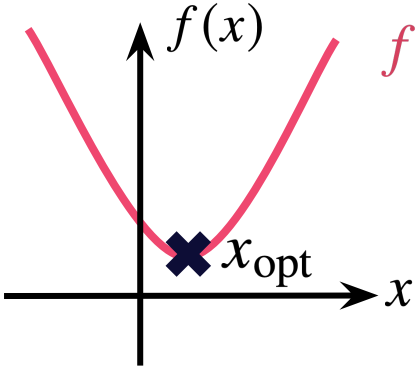
1. Circuit simulators
2. Algorithms

What is combinatorial optimization?

What is optimization?

"Max/min–imize something (while max/min–imizing something else (given these constraints))"

$$\min_{x \in \mathbb{R}} f(x)$$



$$f(x) = ax^2 + bx + c$$

$$\Rightarrow x_{\text{opt}} = -\frac{b}{2a}$$

Combinatorial optimization

$$\min_{x \in \mathbb{R}} f(x)$$

$$\min_{\boldsymbol{x} \in \mathbb{R}^N} f(\boldsymbol{x})$$

...

Optimizing over **discrete** variables

$$\min_{\boldsymbol{x} \in \{0,1\}^N} f(\boldsymbol{x})$$

$$\min_{\boldsymbol{x} \in \{\pm 1\}^N} f(\boldsymbol{x})$$

...

Continuous functions are typically nicer…

Generically, we don’t know how to solve efficiently

What does combinatorial optimization look like?

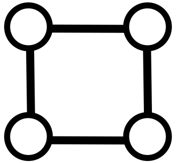
An Ising model

$$\min_{z_i=\pm 1} C(z) = \sum_{i<j} W_{ij} z_i z_j$$

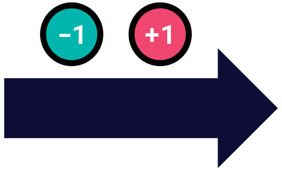
Problem defined by $W_{ij} \in \mathbb{R}$

Statistical physics, condensed matter, chemistry, biology, logistics, scheduling, planning, routing, finance...

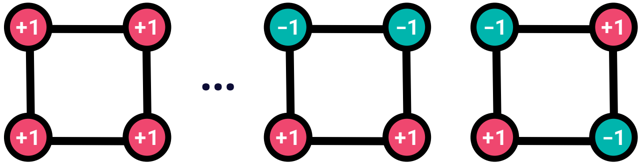
Example, 4 variables



W_{ij}=1 if variables connected; else W_{ij}=0



There are 2^4 possibilities



C=+4

C=0

C=-4 (Optimal)

Recap

- Combinatorial optimization = ground state of Ising model
  2^N possible solutions → Find the best! Can't try them all…

- Genericallly, we don't know how to find the optimal solution efficiently… *("spin glasses")*

- **Finding a *good* approximate solution?**
  → Can be hard too! Let's just find the best we can

What can quantum computers do?

Unlike, say, Shor algorithm for factoring integers, we don't know whether quantum algorithms can provide an advantage here

Some quantum algorithms

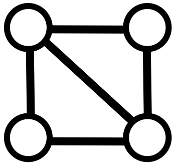
- **Quantum Approximate Optimization Algorithm 'QAOA'**
- Adiabatic quantum evolution or quantum annealing
- A few others, e.g., quantum search/Grover-like algorithms

Can return the optimal solution but **practical implementations** seek a (good) approximate one

⬇

Better than classical algorithms?... 👀

Mapping Ising variables onto qubits



- One variable ↔ one qubit
- Qubit value 0/1 to ±1 Ising variable:

$$\hat{Z}|0\rangle = +1|0\rangle, \quad \hat{Z}|1\rangle = -1|1\rangle$$

One-to-one mapping

$$|\text{bitstring}\rangle \Leftrightarrow z$$

Objective function as an operator

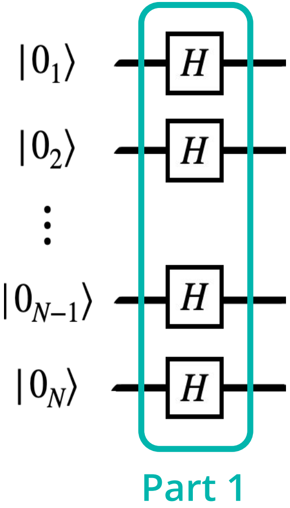
$$C(z) = \sum_{ij} W_{ij} z_i z_j \;\Rightarrow\; \hat{C} = \sum_{ij} W_{ij} \hat{Z}_i \hat{Z}_j$$

Cost of a bit string as an expectation value

$$\langle \text{bitstring} \,|\, \hat{C} \,|\, \text{bitstring} \rangle = C(z)$$

A (quantum) algorithm ideally returns good bit strings

Quantum Approximate Optimization Algorithm



Each layer has two angles: p layers → 2p angles

Equal superposition of all bit strings

Each run returns a bit string
= candidate solution

00110101…1
10001110…0
. . .
11000100…1

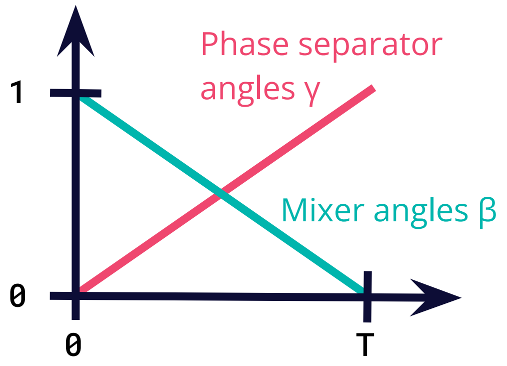
Set the angles β, γ such that the output are **good solutions**

Original QAOA paper: arXiv:1411.4028
Recent review paper: arXiv:2306.09198

Why this circuit structure for the QAOA?

QAOA is a discretized and variational version of quantum annealing

- Initial layer of Hadamard gates prepares the N-qubit system in the ground state of $-\sum_{i=1}^{N} \hat{X}_i$
- **Imagine T layers with 2T parameters**
  Mixer angle from 1 to 0 and phase separator angle from 0 to 1 in **T** steps
- If we can set the angles to whatever we want, **perhaps there are better values than a simple interpolating strategy?**



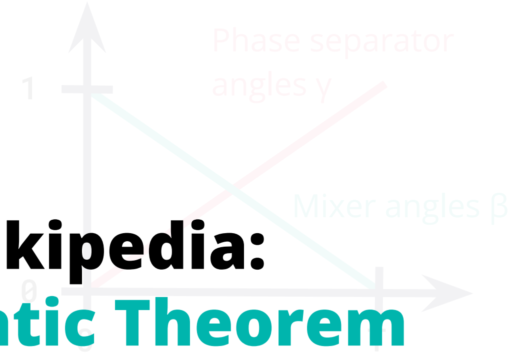
$$e^{i\cdots}\boxed{-\sum_i \hat{X}_i} \rightarrow e^{i\cdots}\boxed{\hat{C}}$$

If slow enough (=adiabatic), the system will go from one ground state to another

Why this circuit structure for the QAOA?

QAOA is a discretized and variational version of quantum annealing

- Initial layer of Hadamard gates prepares the N-qubit system in the ground state of $-\sum_{i=1}^{N} \hat{X}_i$
- **Imagine T layers with 2T parameters** Mixer angle … separator angle from 0 to 1 in T steps.
- If we can set the angles to whatever we want, **perhaps there are better values than a simple interpolating strategy?**



$$e^{i\cdots\left(-\sum_i \hat{X}_i\right)} \rightarrow e^{i\cdots \hat{C}}$$

If slow enough (=adiabatic), the system will go from one ground state to another

If you wanna look on Wikipedia:

~~Quantum Annealing~~ Adiabatic Theorem

Running the QAOA in practice

How to set the angles β, γ to good values?

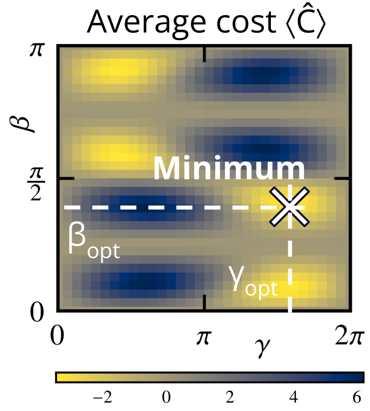
Run at fixed angles
Collect bit strings
Do statistics

$$\langle \hat{C} \rangle \approx \frac{1}{K} \sum_{k=1}^{K} C(z_k)$$

Angles leading on average to the best solutions?

More angles → even better solutions
But more gates leads to more noise?... 😒

Noise-depth trade-off



Recap

- Quantum advantage in combinatorial optimization?
  →**Research question**

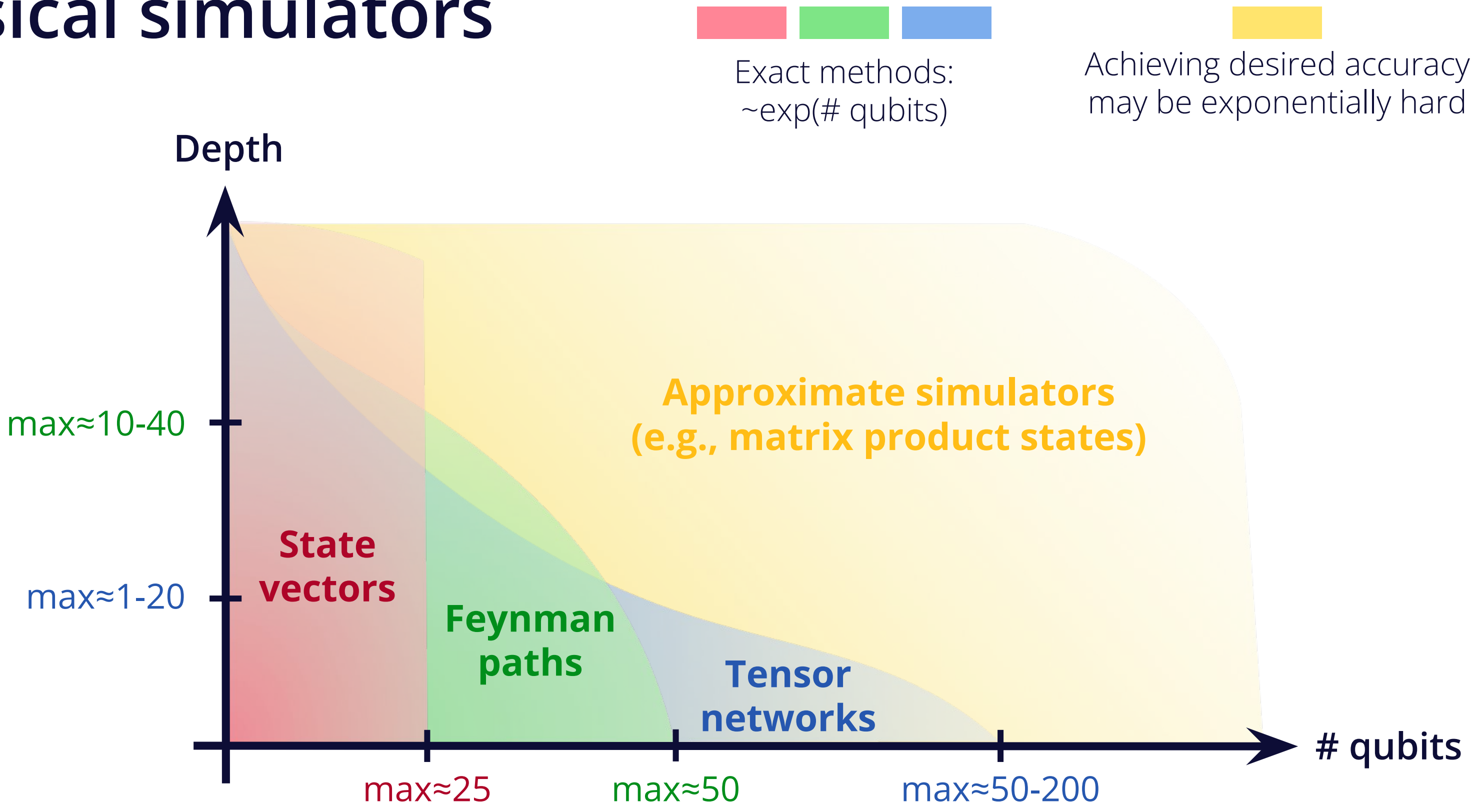
- Developing quantum algorithms returning high quality solutions, albeit not optimal→ *Proving it? Heuristics?*

- Developing near-term friendly quantum algorithms
  → *Noise robustness, error mitigation...*

Getting to quantum advantage means… **beating classical simulators**

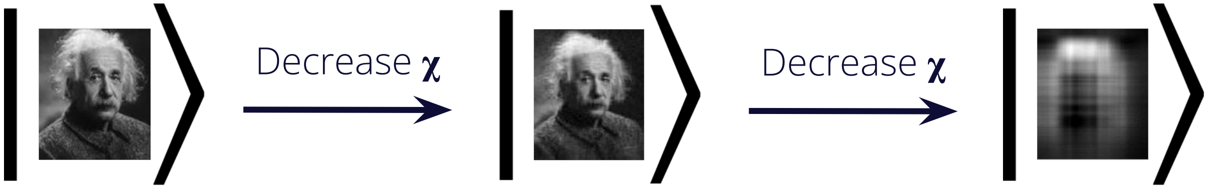
A quantum computer needs to be better than a laptop (or a National lab' supercomputer) at running a quantum circuit

Classical simulators



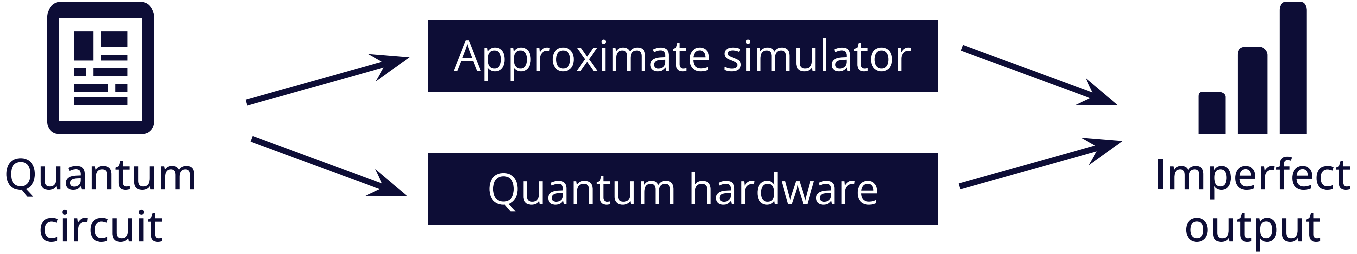
Approximate simulators

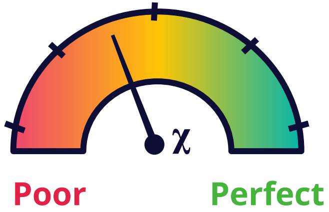
Control parameter χ for the **execution fidelity** of a circuit



Compressing the **entanglement** with **matrix product states**

Noisy quantum hardware is also imperfect



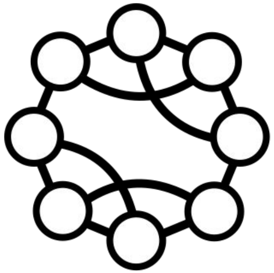


What level of compression χ corresponds to the quantum computer output?

Is this level of compression χ out of reach for an approximate classical simulator?

PRX Quantum 3, 040339 (2022)

Execution of a one-layer QAOA circuit



N=8 unit-weight
3-regular graph
1 QAOA layer

$$\text{"Fidelity"} = \frac{\langle \hat{C} \rangle_{\text{exp}}}{\langle \hat{C} \rangle_{\chi_{\text{exact}}}} = \frac{-1.9(1)}{-4.2} \approx 46(3)\,\%$$

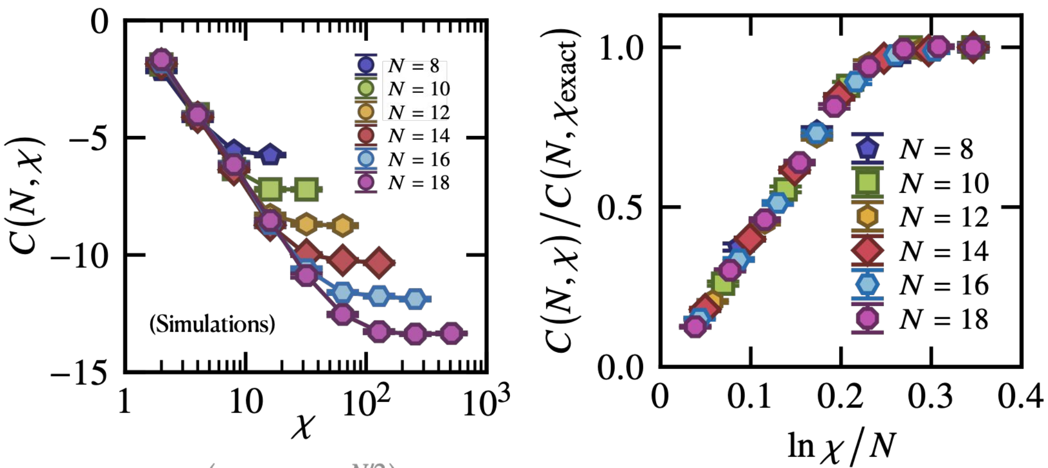
Extending the analysis

PRX Quantum 3, 040339 (2022)

- Set the graph type
- Set the # of QAOA layers

→ Average the minimum cost over randomly generated graphs $\Rightarrow C(N,\chi)$

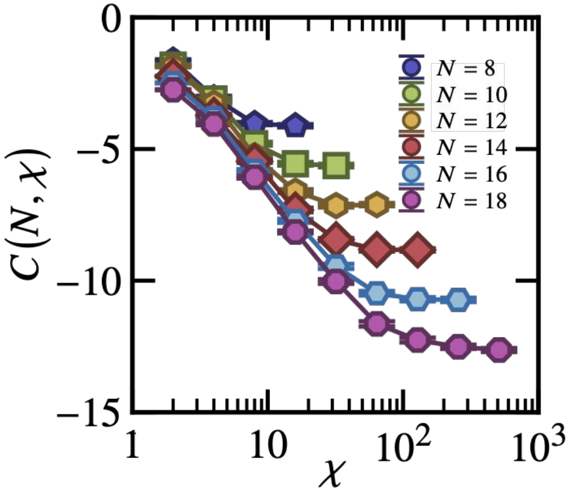
Random unit-weight
3-regular graphs
2 QAOA layers

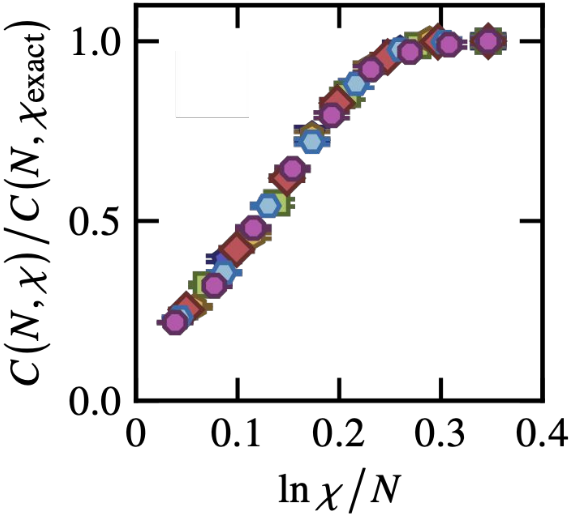


$(\chi_{\text{exact}} = 2^{N/2})$

The fidelity is governed by a scaling relation

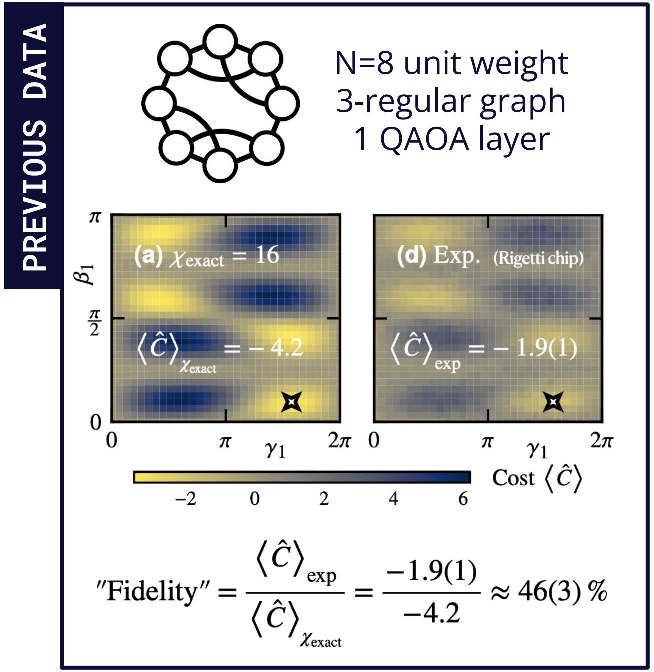
All-to-all graph with random $W_{ij} \in [0,1]$
4 QAOA layers

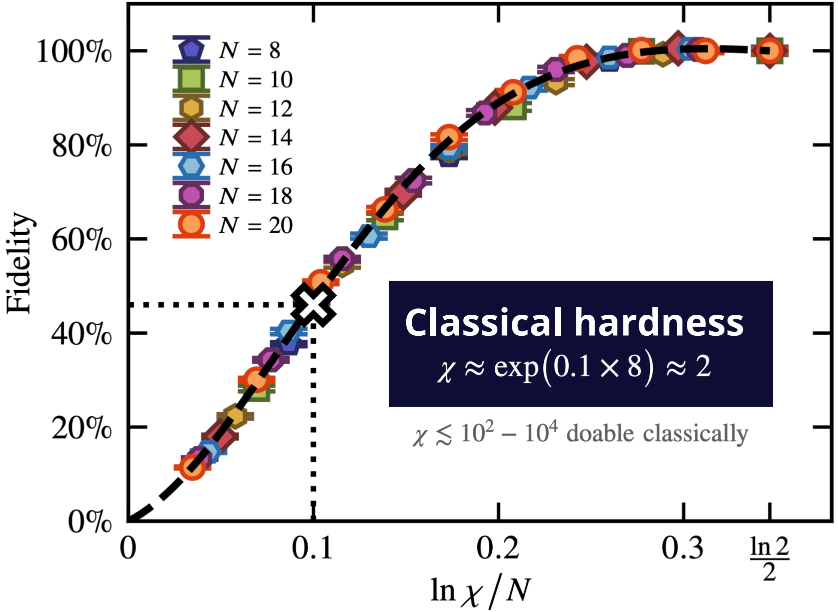




$$\text{"Fidelity"} = \frac{C(N,\chi)}{C(N,\chi_{\text{exact}})} = \mathscr{F}\left(\ln \chi / N\right)$$

How hard is it to beat the quantum computer?





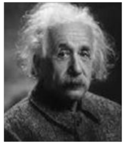
PRX Quantum 3, 040339 (2022)

An entanglement perspective

PRX Quantum 3, 040339 (2022)
PRA 106, 022423 (2022)

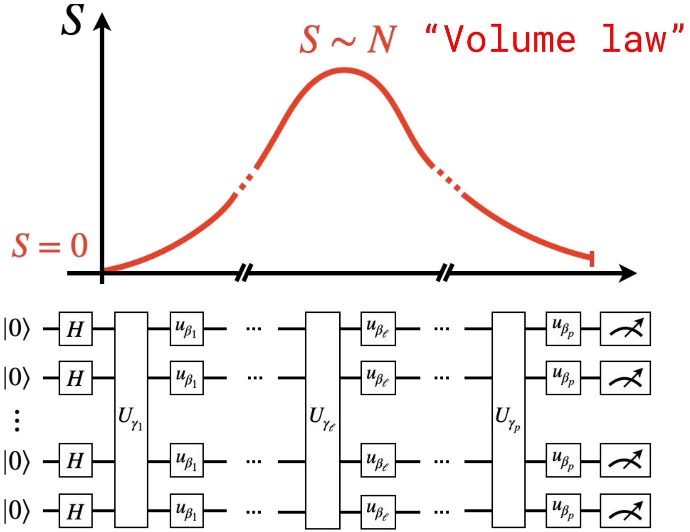
$$\text{"Fidelity"} = \frac{C(N,\chi)}{C(N,\chi_{\text{exact}})} = \mathscr{F}\left(\ln\chi/N\right)$$

Control parameter χ bounds the amount of entanglement S



Decrease χ

$S \leq \ln(\chi)$



Entanglement S after each circuit layer

Engineering efforts → classical hardness

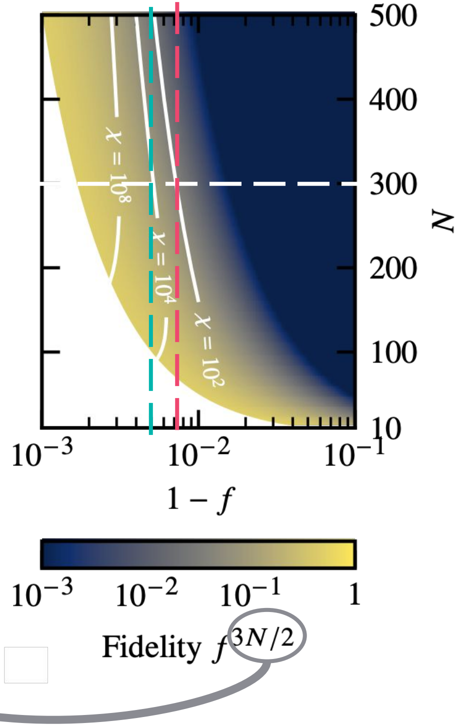
PRX Quantum 3, 040339 (2022)

1 QAOA layer for 3-regular graphs

3N/2 edges in 3-regular graphs

K operations = MINIMUM of 3N/2 2Q gates

(most likely more in practice)



$$\text{Fidelity} \approx \prod_{\text{Operations } i} f_i \rightarrow f^K \text{ for } K \text{ operations}$$

N=300	
1-f=0.7%	1-f=0.5%
Fidelity ≈ 4%	Fidelity ≈ 10%
χ ≈ 100	χ ≈ 10,000

$\chi \lesssim 10^2 - 10^4$ (classically doable)

Recap

- Class of simulators that are approximate and **compress the quantum state → main competitors nowadays**

- Beating approximate classical simulators

- Hardware noise deteriorates the quality of solutions
  → ***Improve hardware… AND develop new algorithms***

Getting to quantum advantage means… **beating classical algorithms**

People didn’t wait for quantum computers
to try solving combinatorial optimization problems!…

Classical solvers

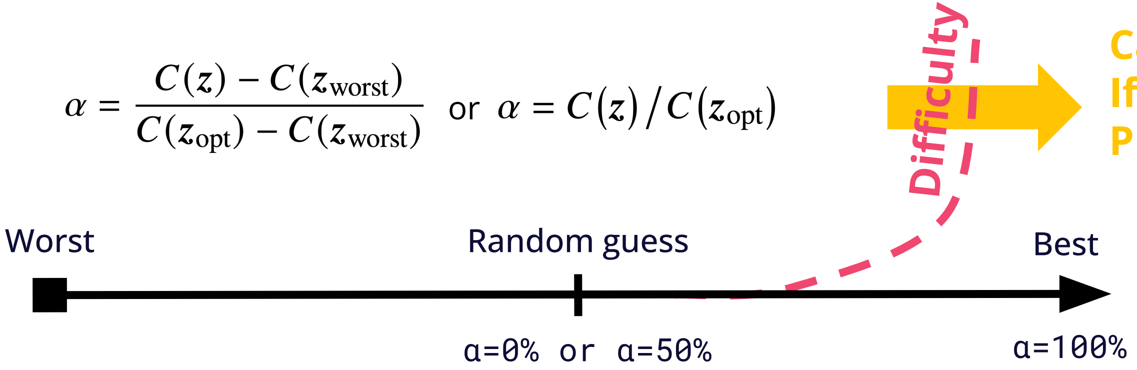
Heuristic algorithms
"try and see how well it works"

Simulated annealing/Markov chain Monte Carlo, tabu search, genetic algorithms...

Algorithms with provable performances

Random guessing, some greedy algorithms, semidefinite programming...

$$\alpha = \frac{C(z) - C(z_{\text{worst}})}{C(z_{\text{opt}}) - C(z_{\text{worst}})} \quad \text{or} \quad \alpha = C(z)/C(z_{\text{opt}})$$



Can quantum do better?
If not, same but faster?
Practically or mathematically?

Quality of a solution = approximation ratio α

Classical solvers

Heuristic algorithms
"try and see how well it works"

Algorithms with provable performances

Simulated annealing/Markov chain Monte Carlo, tabu search, genetic algorithms...

Random guessing, some greedy algorithms, semidefinite programming...

Previously

How good versus an exact simulation

Now

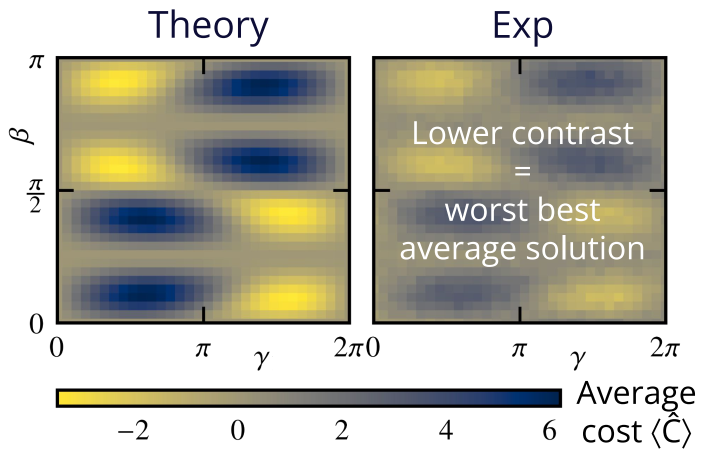
How good versus the optimal solution

$$\alpha = \frac{C(z) - C(z_{\text{worst}})}{C(z_{\text{opt}}) - C(z_{\text{worst}})} \quad \text{or} \quad \alpha = C(z)/C(z_{\text{opt}})$$

Quality of a solution = approximation ratio α

What's a big problem right now?

Strong noise
Average solution no better than random 😒

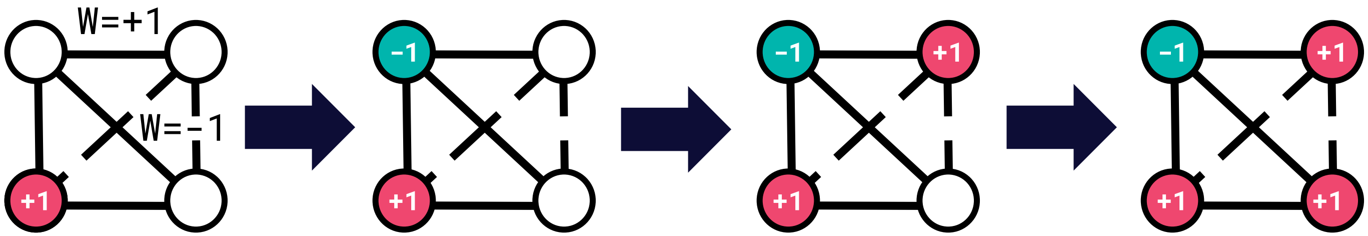


Option A: Wait until hardware gets better

Option B: Be creative and develop new algorithms

Simple classical alg. does better than noisy quantum!

Greedy approach



$$C(z) = \sum_{i<j} W_{ij} z_i z_j$$

C=-1 =-3 =-4

Select node **at random** → Try ±1 → Keep the best → Repeat

All-to-all graphs with random ±1 weights
=Sherrington-Kirkpatrick spin glasses

(Famous stat-mech problem)

Random guess: α=50%

Average solution α=84.8% of optimal

arXiv:2303.05509

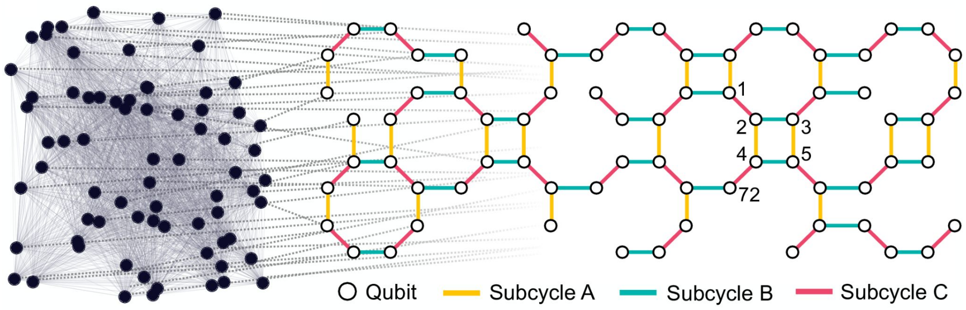
A quantum-enhanced greedy solver

Quantum computer guides freezing strategy

1. Run the QAOA at good angles
2. Find the "best" node to freeze. Freeze greedily
3. Repeat...

Problem

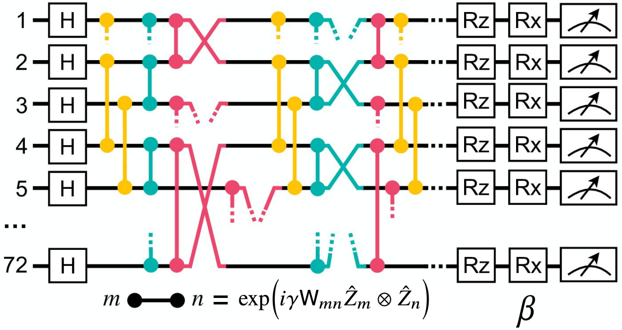
Mapping to hardware



72 variables

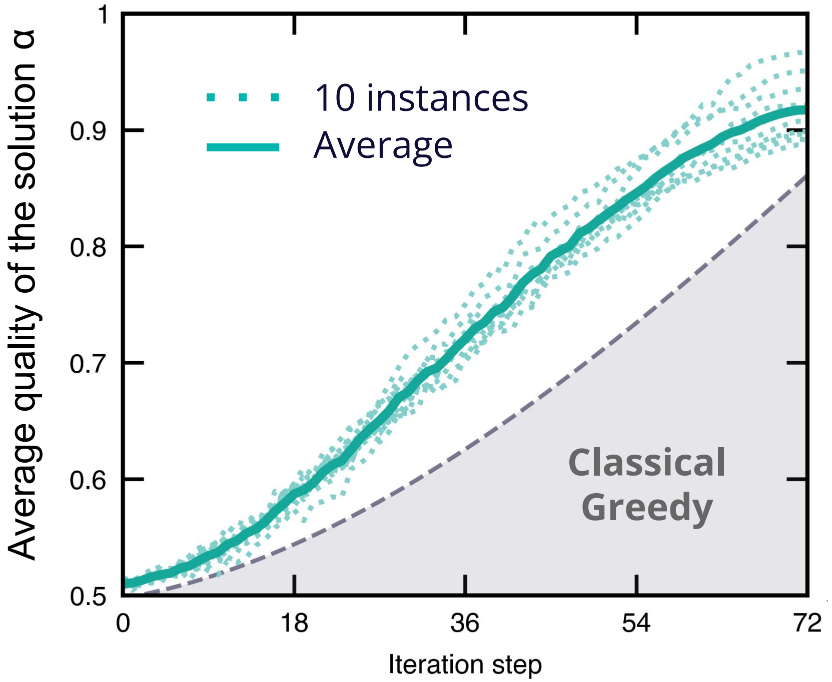
72 superconducting qubits

Truncated QAOA circuit, two variational angles



~400 2Q gates + ~5,000 1Q gates

Performance guarantees with noise



1. QAOA → `00110101…1`, `10001110…0`, …

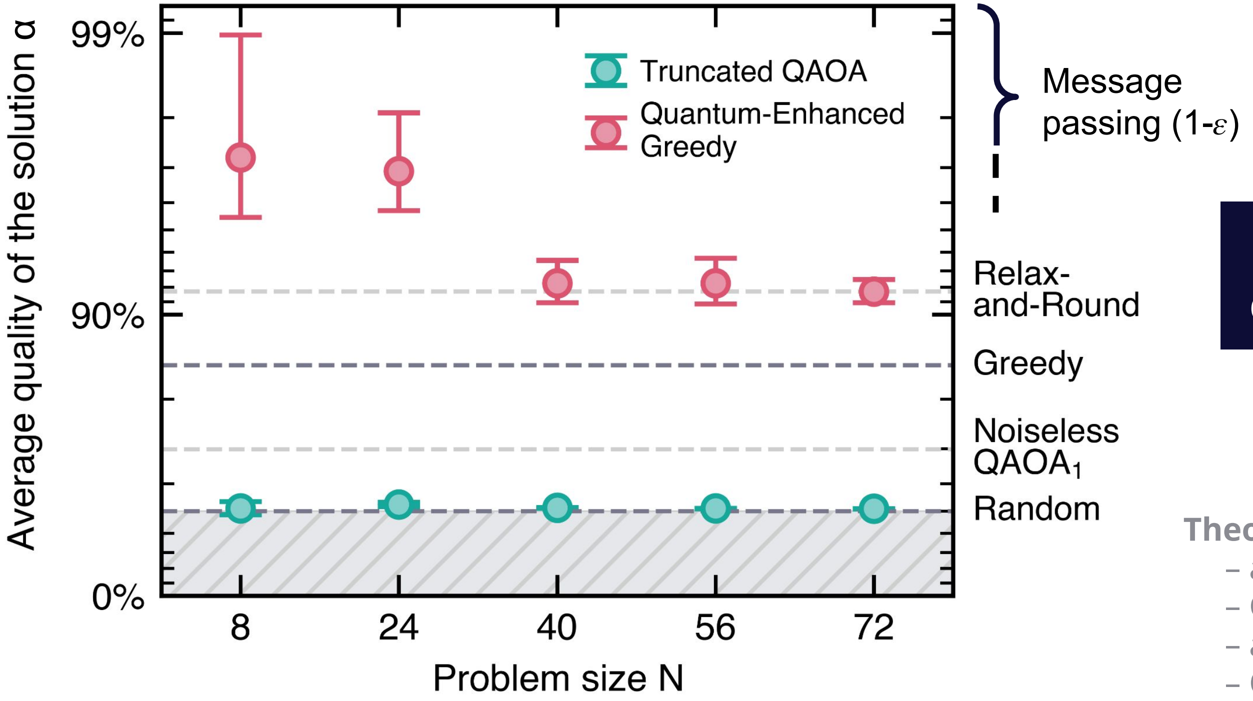
2. Select $\max_k \sum_{i \neq k} \left| \langle \hat{Z}_i \hat{Z}_k \rangle \right|$ *"Majority vote++"*

3. k is best frozen to 0 or 1?

4. Remove k and repeat

Random selection = classical
We beat it!

Comparing to other solvers with provable guarantees

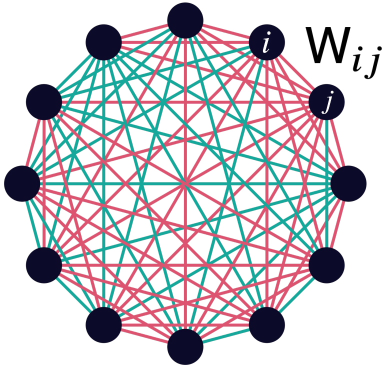


Quantum-enhancing other algorithms? 🤔

Theoretical performance bounds:

- arXiv:1812.10897
- Commun. Math. Phys. 112, 3 (1987)
- arXiv:2303.05509
- Quantum 6, 759 (2022)

What's behind the relax-and-round solver?



$$\min_{z_i=\pm 1} C(z) = \sum_{ij} W_{ij} z_i z_j$$
$$= z^T W z$$

Relax the ±1 constraint

$$\min_{\|z\|=1} C(z) = z^T W z$$

Eigenvalue problem:
Diagonalize **W**

$$z_i \leftarrow \mathrm{sign}(z_i)$$

Round eigenvectors,
keep the best

Round back
to ±1

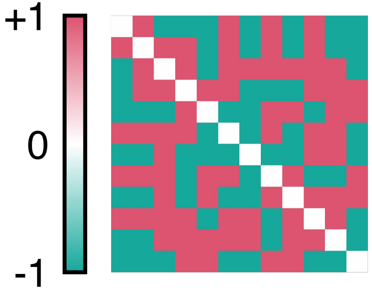
Best for 31 years for solving Sherrington-Kirkpatrick spin glasses: ***91.7% optimal***

Commun. Math. Phys. 112, 3 **(1987)** → arXiv:1812.10897 **(2018)**

Just quantum, relax, and round

arXiv:2307.05821

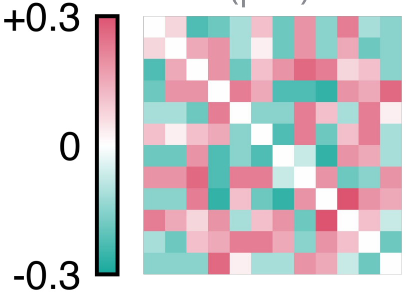
Adjacency matrix **W** of the graph



QAOA$_p$(W)

p layers
2p variational angles

(p=1)



Correlation matrix **Z$^{(p)}$**

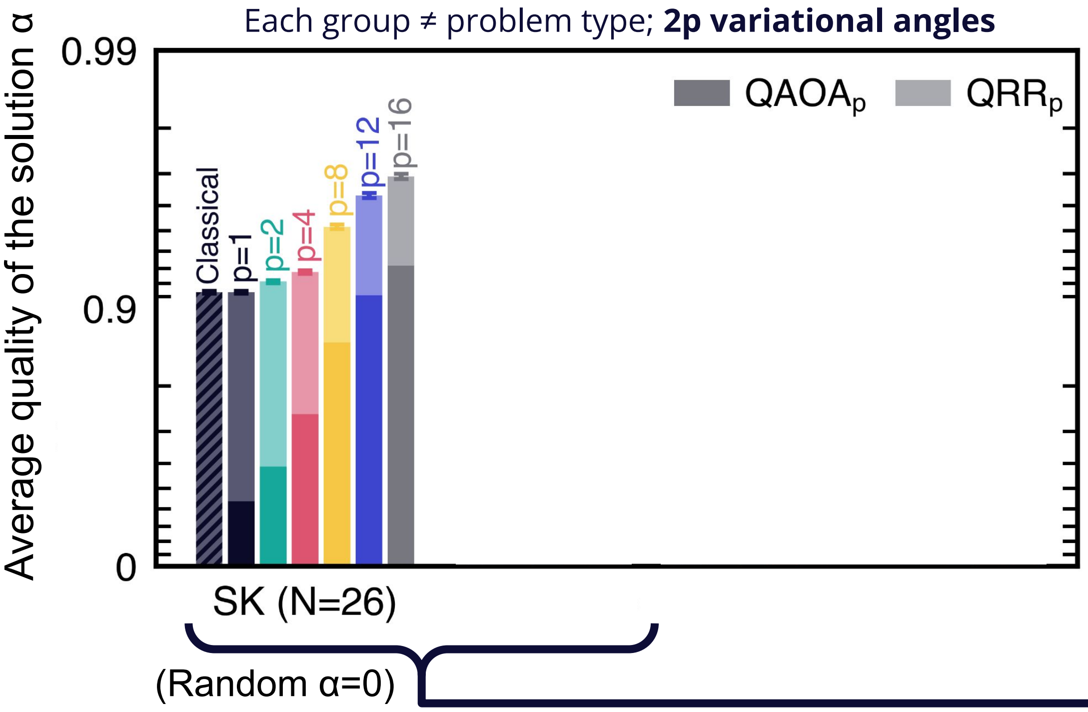
$$Z^{(p)}_{i\neq j} = -\langle \hat{Z}_i \hat{Z}_j \rangle_{\mathrm{QAOA}_p}$$

Relax and round with
Z$^{(p)}$ instead of **W**

$$\min_{\|z\|=1} z^T Z^{(p)} z \quad \textbf{and} \quad z_i \leftarrow \mathrm{sign}(z_i)$$

Performance on par with classical

arXiv:2307.05821



Show the adjacency and correlation matrices have the same eigenvectors → show they commute

Quantum = classical at p=1 (two angles)

Recap

- **Research on developing new quantum algorithms**
  Near-term friendly, better performing, maths/guarantees?

- Beating classical heuristics also needed
  Likely, more qubits needed to (perhaps?) start beating those

- Error mitigation?

Thank you!

Maxime Dupont
mdupont@rigetti.com

with NASA, USRA, UC Berkeley,and Berkeley Lab

M. S. Alam, N. Didier, B. Evert, D. Feng, S. Grabbe, S. Hadfield, M. J. Hodson, S. Jeffrey, P. A. Lott, F. B. Maciejewski, Joel E. Moore, M. J. Reagor, E. G. Rieffel, B. Sundar, D. Venturelli, Z. Wang, and Y. Yamaguchi

Combinatorial optimization → Quantum algorithms → Beating classical simulators → Beating classical algorithms

Quantum advantage? 👀