Overview of Quantum Optimization

Ruslan Shaydulin

JPMorganChase

OPTIMIZATION

- General problem: $\min_{x \in K} f(x)$
- Examples:
  - Find portfolio with maximum return and minimum risk
  - Find a shortest route between two points on a map

- General classes of optimization problems:
  - *Combinatorial optimization*: variables are discrete (bits, integers)
    - Often NP-hard!
  - *Continuous optimization*: variables are real (floating point)
    - Convex or non-convex

WHY QUANTUM FOR OPTIMIZATION?

- Straightforward to encode in a circuit
  - Objective function can often be encoded directly, no data loading necessary
- Many small hard problems
  - Some combinatorial optimization problems become intractable at a few hundred binary variables (e.g. Low Autocorrelation Binary Sequences)
- Clear value
  - Optimization is ubiquitous in industry
  - Doing optimization better directly connects to business value
- Evidence of broadly applicable speedups

QUANTUM SPEED-UPS FOR OPTIMIZATION

Combinatorial

- We don’t expect quantum computers to have exponential speedup for NP-hard problems
- But! Many provable polynomial speedups exist:
  - Quadratic speedups that leverage Grover’s algorithm as a subroutine
    - Branch-and-bound [Montanaro ’19, Chakrabarti et al. ’20]
    - Dynamic programming [Ambainis et al.’18]
    - …
  - Quantum walk algorithms
    - Quadratic speedup for backtracking [Montanaro’15]
    - …

QUANTUM SPEED-UPS FOR OPTIMIZATION

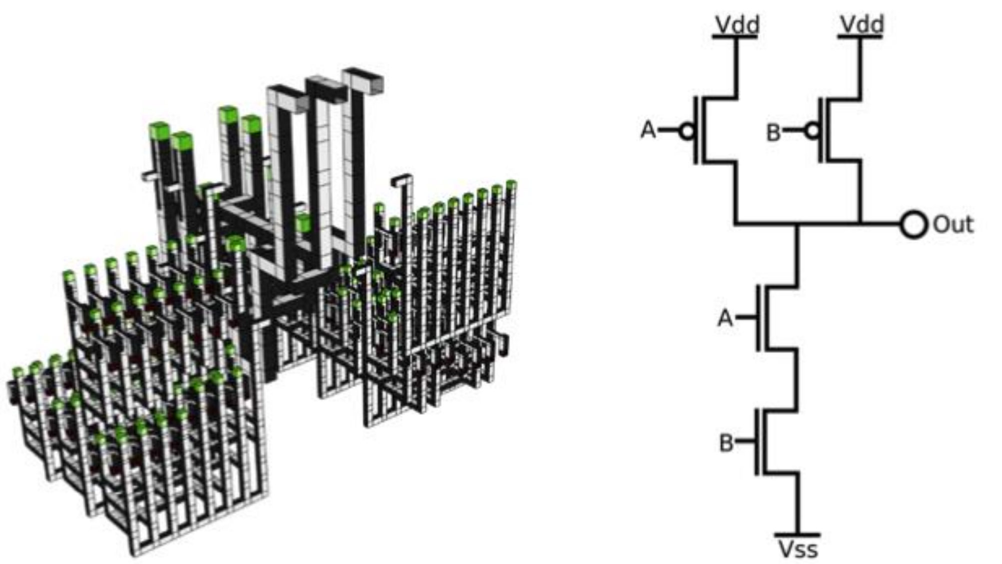
Combinatorial

- We don't expect quantum computers to have exponential speedup for NP-hard problems
- But! Many provable polynomial speedups exist:
  - Quadratic speedups that leverage Grover's algorithm as a subroutine
    - Branch-and-bound [Montanaro '19, Chakrabarti et al. '20]
    - Dynamic programming [Ambainis et al.'18]
    - …
  - Quantum walk algorithms
    - Quadratic speedup for backtracking [Montanaro'15]
    - …

⚠ Warning! ⚠
Quadratic speedups are likely to be insufficient in the near term due to overhead of error correction [Babbush et al '20]

QUANTUM SPEED-UPS FOR OPTIMIZATION

Combinatorial



(a) "Quantum NAND" > 10 qubitseconds

(b) "Classical NAND" $< 10^{-9}$ transistorseconds

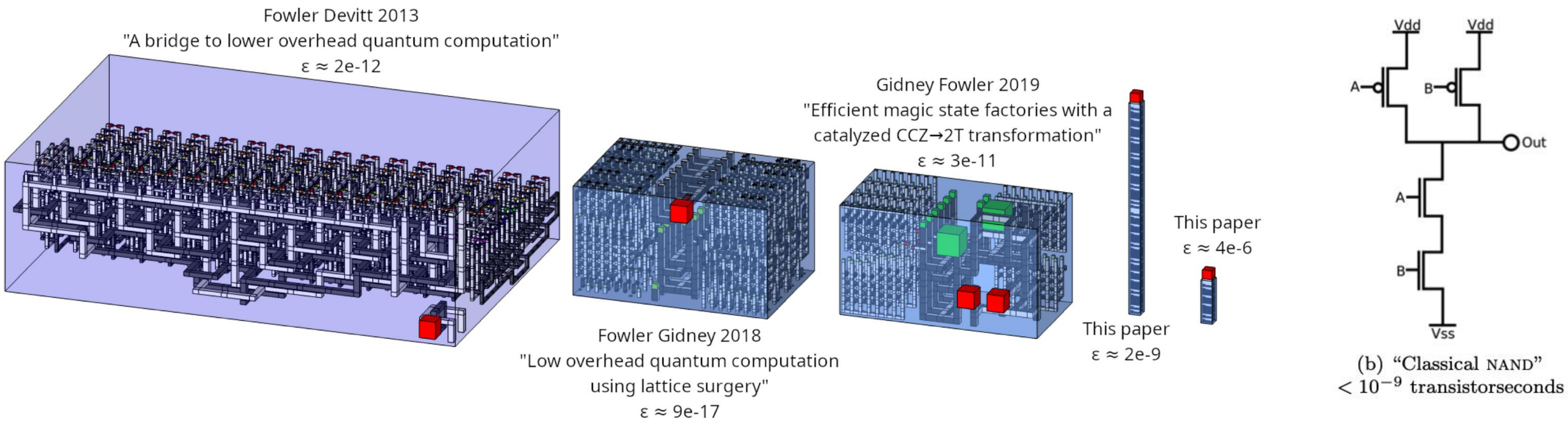
polynomial degree d	parallelism speedup S	resource "lower bound"	
		iterations M	runtime $\mathcal{T}^\star$
Quadratic, $d = 2$	1	5.2×10^5	2.4 hours
	10^3	5.2×10^8	100 days
	10^6	5.2×10^{11}	280 years
Cubic, $d = 3$	1	7.2×10^2	12 seconds
	10^3	2.3×10^4	6.4 minutes
	10^6	7.2×10^5	3.4 hours
Quartic, $d = 4$	1	8.0×10^1	1.4 seconds
	10^3	8.0×10^2	14 seconds
	10^6	8.0×10^3	2.3 minutes

⚠ Warning! ⚠

Quadratic speedups are likely to be insufficient in the near term due to overhead of error correction [Babbush et al '20]

QUANTUM SPEED-UPS FOR OPTIMIZATION

Combinatorial



[Gidney et al '24]

⚠ Warning! ⚠

Quadratic speedups are likely to be insufficient in the near term due to overhead of error correction [Babbush et al '20]

QUANTUM SPEED-UPS FOR OPTIMIZATION

Combinatorial

- We don't expect quantum computers to have exponential speedup for NP-hard problems
- Quadratic speedups are likely to be insufficient in the near term
- NEW Super-quadratic speedups
  - Slightly-better-than-quadratic speedup over brute-force-search by starting Grover from a "warm-start" state [Dalzell et al '22]

QUANTUM SPEED-UPS FOR OPTIMIZATION

Combinatorial

- We don’t expect quantum computers to have exponential speedup for NP-hard problems
- Quadratic speedups are likely to be insufficient in the near term
- NEW Super-quadratic speedups
  - Slightly-better-than-quadratic speedup over brute-force-search by starting Grover from a “warm-start” state [Dalzell et al ’22]

Problem	**Our quantum algo**	**Best classical algo**
3-CNF-SAT	$0.5 - (5.2 \times 10^{-7})$	0.39 [3]
k-CNF-SAT	$0.5 - \Omega(2^{-3k} k^{-3})$	$1 - \Omega(k^{-1})$ [3]
SK model	$0.5 - (2.7 \times 10^{-5})$	0.45 [11]
k-spin	$0.5 - \Omega(k^{-3})$	1

QUANTUM SPEED-UPS FOR OPTIMIZATION

Combinatorial

- We don't expect quantum computers to have exponential speedup for NP-hard problems
- Quadratic speedups are likely to be insufficient in the near term
- NEW Super-quadratic speedups
  - Slightly-better-than-quadratic speedup over brute-force-search by starting Grover from a "warm-start" state [Dalzell et al '22]
  - NEW NEW Slightly-better-than-quadratic speedup over Markov Chain search for a broad range of constrained problems [Chakrabarti et al '24]
    - Super-quadratic speedup over *any* classical search with a polynomial-time Gibbs sampler for a certain class of problems

QUANTUM SPEED-UPS FOR OPTIMIZATION

Combinatorial

- We don't expect quantum computers to have exponential speedup for NP-hard problems
- Quadratic speedups are likely to be insufficient in the near term
- NEW Super-quadratic speedups
  - Slightly-better-than-quadratic speedup over brute-force-search by starting Grover from a "warm-start" state [Dalzell et al '22]
  - NEW NEW Slightly-better-than-quadratic speedup over Markov Chain search for a broad range of constrained problems [Chakrabarti et al '24]
  - Quartic speedup for planted kXOR [Schmidhuber et al '24]
- NEW Exponential space advantage in streaming setting [Kallaugher et al '23]
- NEW NEW Potential exponential speedup for restricted family of problems which are not known to be NP-hard [Jordan et al '24]

QUANTUM SPEED-UPS FOR OPTIMIZATION

Combinatorial

- We don't expect quantum computers to have exponential speedup for NP-hard problems
- Quadratic speedups are likely to be insufficient in the near term
- NEW Super-quadratic speedups
  - Slightly-better-than-quadratic speedup over brute-force-search by starting Grover from a "warm-start" state [Dalzell et al '22]
  - NEW NEW Slightly-better-than-quadratic speedup over Markov Chain search for a broad range of constrained problems [Chakrabarti et al '24]
  - Quartic speedup for planted kXOR [Schmidhuber et al '24]
- NEW Exponential space advantage in streaming setting [Kallaugher et al '23]
- NEW NEW Potential exponential speedup for restricted family of problems which are not known to be NP-hard [Jordan et al '24]
  - Easy to study empirically by studying classical decoders!

QUANTUM HEURISTICS FOR OPTIMIZATION

Combinatorial

- Heuristics have been proposed that appear to do well numerically
  - Quantum Approximate Optimization Algorithm [Hogg ’00, Farhi ’14]
    - Some numerical evidence of speedup is available [Boulebnane ’22, Shaydulin ’23]

Solver	Fit	Error
WalkSAT QAOA	$-3.232 + 0.295n$	0.011
QAOA ($p = 14$)	$-1.064 + 0.326n$	0.008
QAOA ($p = 60$)	$-2.842 + 0.302n$	0.007
walksatlm	$-0.309 + 0.325n$	0.008
maplesat	$1.531 + 0.461n$	0.004
glucose4	$2.998 + 0.498n$	0.005

[Boulebnane ’22]

	QAOA+QMF	QAOA	Memetic Tabu Reproduced	Memetic Tabu (23, 30)
Fit	**1.21**	1.46	1.35	1.34
CI	(1.19, 1.23)	(1.42,1.50)	(1.33,1.38)	N/A

[Shaydulin ’23]

QUANTUM HEURISTICS FOR OPTIMIZATION

Combinatorial

- Heuristics have been proposed that appear to do well numerically
  - Quantum Approximate Optimization Algorithm [Hogg ’00, Farhi ’14]
    - Some numerical evidence of speedup is available [Boulebnane ’22, Shaydulin ’23]
  - Quantum Adiabatic Algorithm [Farhi ’00]
    - Proving performance is challenging, but can be run heuristically
  - Quantum Counterdiabatic Driving [Berry ’09]
- Experiments on hardware will tell if these work

QUANTUM SPEED-UPS FOR OPTIMIZATION

Continuous

- Provable polynomial speedups for convex problems
  - Semidefinite programming [Brandao-Svore ’16, van Apeldoorn-Gilyen ’18]
  - Linear programming [Kerenidis-Prakash ’18, Augustino et al. ’23]

“Even if quantum computers one day match the gigahertz-level clock-speeds of modern classical computers, **10^{24} layers of T gates would take millions of years to execute**. By contrast, the PO problem can be easily **solved in a matter of seconds on a laptop for n = 100 stocks**.”

[Dalzell ‘22 on Kerenidis-Prakash ‘18]

QUANTUM SPEED-UPS FOR OPTIMIZATION

Continuous

- Provable polynomial speedups for convex problems
  - Semidefinite programming [Brandao-Svore ’16, van Apeldoorn-Gilyen ’18]
  - Linear programming [Kerenidis-Prakash ’18, Augustino et al. ’23]

Table 1: Complexity to solve the primal-dual pair (P)-(D) to precision ε

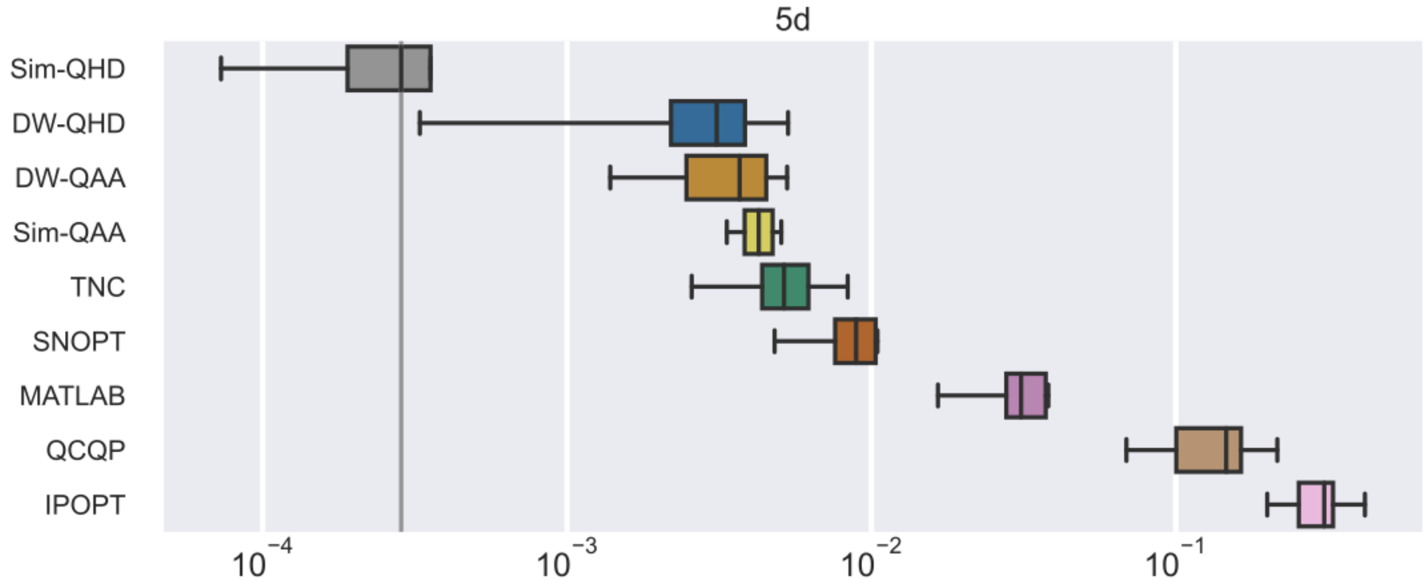
Algorithm	Complexity	QRAM	Notes
IPM [CLS21, vdB20]	$\widetilde{\mathcal{O}}_{n,\frac{1}{\varepsilon}}((m+n)^{\omega})$	-	
QMMWU [BGJ+23]	$\widetilde{\mathcal{O}}\left(\sqrt{m+n}r^{2.5}\varepsilon^{-2.5}+\varepsilon^{-3}\right)$	✓	$r \geq \|x_*\|_1$
IR-QIPM [MFWT23]	$\widetilde{\mathcal{O}}_{n,\kappa(Q),\frac{1}{\varepsilon}}\left((m+n)^{2.5}\kappa(Q)^2\|Q\|\|x_*\|^5\right)$	✓	
IR-QCPM (this work)	$\widetilde{\mathcal{O}}_{m,n,\kappa(\mathcal{M}),\frac{1}{\varepsilon}}\left((m+n)\,\mathrm{nnz}(A)\kappa(\mathcal{M})\right)$	✗	

[Augustino et al. ’23]

QUANTUM SPEED-UPS FOR OPTIMIZATION

Continuous

- Provable polynomial speedups for convex problems
  - Semidefinite programming [Brandao-Svore ’16, van Apeldoorn-Gilyen ’18]
  - Linear programming [Kerenidis-Prakash ’18, Augustino et al. ’23]
- Promising heuristics for non-convex problems
  - Quantum Hamiltonian Descent [Leng et al ’23]
  - Quantum Langevin Dynamics [Chen et al ‘23]



[Leng et al ’23]

TAKEAWAY

- Optimization is a promising domain for quantum algorithms due to availability of broadly applicable speedups and promising heuristics
- Speedups available in both discrete and continuous setting
  - Mostly polynomial, though recent results suggest possibility of exponential separations
- Experiments on early fault-tolerant devices will show the power of heuristics