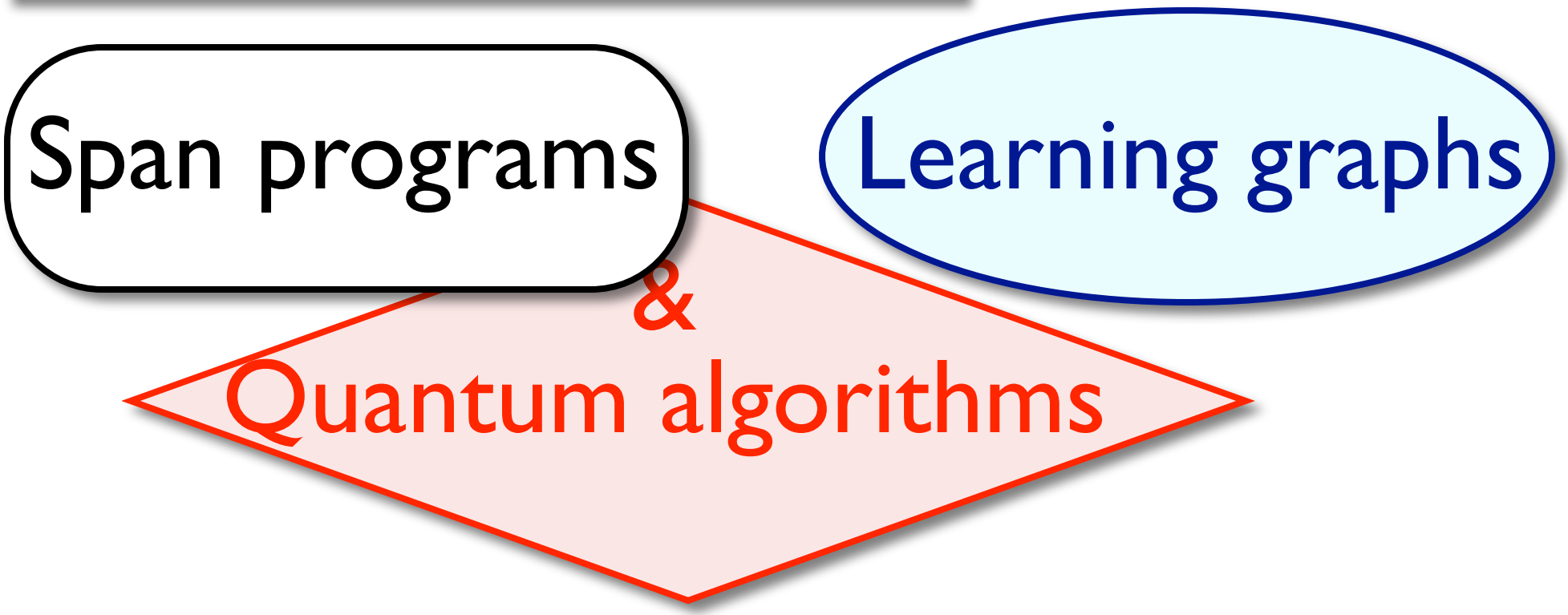
The adversary bound



Ben Reichardt

University of Southern California

New quantum algorithms

General formula evaluation	[Rei11b]
Almost-balanced formulas	[RŠ12, Rei11c]
AND-OR formulas ("game trees")	[Rei11a]
Large AND-OR formulas with inputs satisfying certain promises	[ZKH12, Kim12]
Triangle detection	[Bel12b]
Related graph problems, e.g., subgraph detection	[Zhu12, LMS11]
s-t connectivity, certain subgraph-detection & subgraph/not-a-minor problems	[BR12]
Graph collision	[GI12]
k-distinctness and 3-distinctness	[BL11, Bel12a]
Matrix rank	[Bel11]

super-polynomial quantum speedups

Other applications

Merkle puzzles [BHKKLS11]

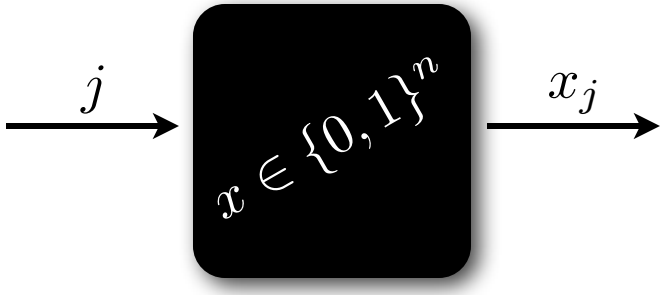
State conversion [AMRR11, ORR11, LMRSŠ11]

Direct-product theorem [LR11]

Query complexity

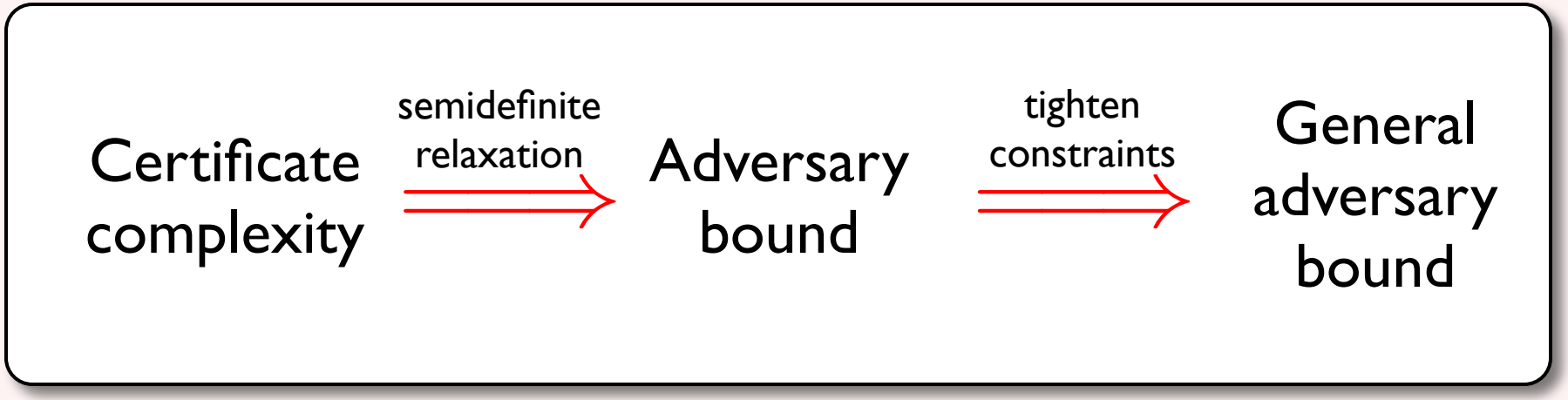
Goal: Evaluate function f on input *x*

Resource:



Algorithm:

The general adversary bound



A certificate for input x is a set of positions whose values fix f.

For f=OR:	Input	Minimal certificate
	00110	{3}
	00000	{1,2,3,4,5}

Certificate complexity = length of worst input's shortest certificate

(= nondeterministic query complexity)

A certificate for input x is a set of positions whose values fix f.

For f=OR:

Input	Minimal certificate
00110	{3}
00000	{1,2,3,4,5}

Certificate complexity $=$

$$\min_{\{\vec{p}_x \in \{0,1\}^n\}} \max_x \sum_j p_x[j]$$

$$\text{s.t.} \sum_{j:x_j \neq y_j} p_x[j] p_y[j] \geq 1 \quad \text{if } f(x) \neq f(y)$$

worst input's certificate length

minimize over certificates for all inputs x

(for total functions)

Certificate complexity

semidefinite relaxation

Adversary bound

tighten constraints

General adversary bound

worst input's certificate length

minimize over certificates for all inputs x

$$= \min_{\{\vec{p}_x \in \{0,1\}^n\}} \max_x \sum_j p_x[j]$$
$$\text{s.t.} \sum_{j:x_j \neq y_j} p_x[j]p_y[j] \geq 1 \quad \text{if } f(x) \neq f(y)$$

$$= \min_{\{\vec{p}_x \in \mathbb{R}^n\}} \max_x \sum_j p_x[j]^2$$
$$\text{s.t.} \sum_{j:x_j \neq y_j} p_x[j]p_y[j] \geq 1 \quad \text{if } f(x) \neq f(y)$$

$$= \min_{\{p_x[j] \in \mathbb{R}^m\}} \max_x \sum_j p_x[j]^2$$
$$\text{s.t.} \sum_{j:x_j \neq y_j} p_x[j]p_y[j] = 1 \quad \text{if } f(x) \neq f(y)$$

Certificate complexity

Adversary bound

- BBBV '97
- Ambainis '00
- Høyer, Neerbek, Shi '02
- Ambainis '03
- Barnum, Saks, Szegedy '03
- Laplante, Magniez '03
- Zhang '03
- Barnum, Saks '04

General adversary bound

- Høyer, Lee, Špalek '06

Theorem: Quantum query complexity $\left(f\right) = \Theta\left(\text{General adversary bound}\left(f\right)\right)$

Rest of this talk:

~~Proof~~

Applications!

Schur-Hadamard operator norm

[Schur 1911]

Filtered norm

[LMRŠS 2011]

$$\gamma_2\left(\text{matrix } A\right) = \min_{\{\vec{v}_x\}} \max_x \|\vec{v}_x\|^2 \quad \text{s.t. } A_{x,y} = \vec{v}_x \cdot \vec{v}_y$$

using short vectors

factor A as (row vectors)*(col vectors)

$$\gamma_2\left(\begin{matrix} A \text{ given} \\ \text{matrices } Z_j \end{matrix}\right)$$

= cost to factor A when Z_j matrices are given at cost 1

(used in classical & quantum *communication* complexity)

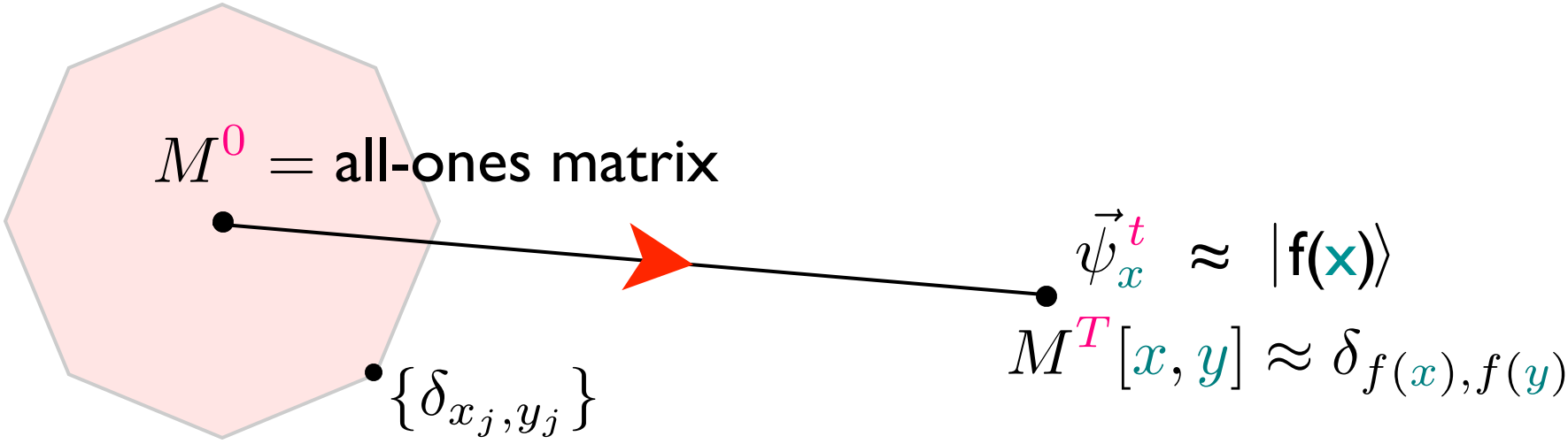
$$\text{General adversary bound}\left(f\right) = \gamma_2\left(\begin{matrix} \{\delta_{f(x),f(y)}\}_{x,y} - \text{(all-ones matrix)} \\ \text{given } \{\delta_{x_j,y_j}\}_{x,y} - \text{(all-ones matrix)} \end{matrix}\right)$$

General adversary bound $\left(f \right) = \gamma_2 \left(\begin{array}{l} \{\delta_{f(x),f(y)}\}_{x,y} - \text{(all-ones matrix)} \\ \text{given} \quad \{\delta_{x_j,y_j}\}_{x,y} - \text{(all-ones matrix)} \end{array} \right)$

Interpretation

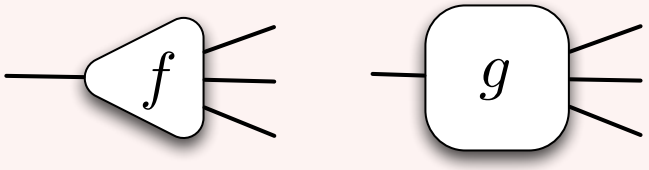
$\vec{\psi}_x^t$ = state of algorithm on input *x* at time *t*

$M^t[x,y] = \vec{\psi}_x^t \cdot \vec{\psi}_y^t$



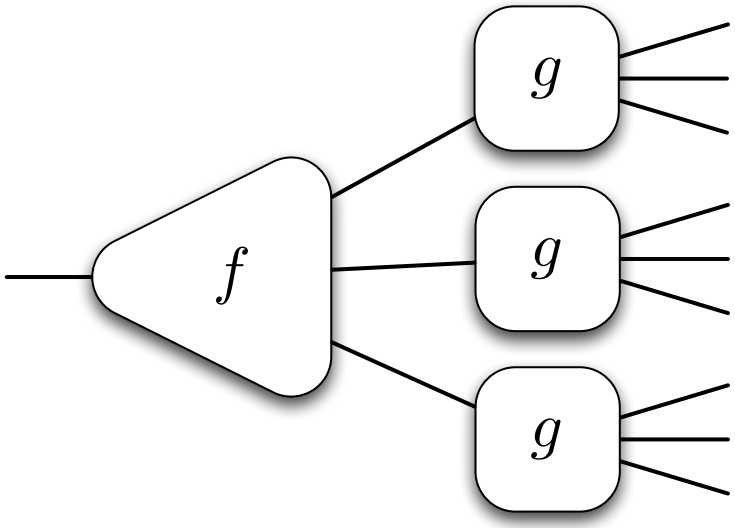
Given

Functions f, g



& optimal algorithms for each

Composed function



Is the composed algorithm optimal?

Deterministic: ✓

Nondeterministic: ✗

Randomized: ✗

$$R(f \circ \vec{g}) = O\big(R(f)R(g)\log R(f)\big)$$

$$\gamma_2\big(A|\{Z_1,\dots,Z_n\}\big) = \min_{\{\vec{u}_{xj},\vec{v}_{yj}\}} \max\left\{ \max_{\text{rows } x} \sum_j \|\vec{u}_{xj}\|^2, \max_{\text{cols } y} \sum_j \|\vec{v}_{yj}\|^2 \right\}$$

$$\text{s.t. } A_{x,y} = \sum\nolimits_j (Z_j)_{x,y} \langle \vec{u}_{xj}|\vec{v}_{yj}\rangle \text{ for all } x,y$$

- $\gamma_2(\cdot|Z)$ is a norm
- $\gamma_2(A|\{\text{all-ones matrix}\}) = \gamma_2(A)$
- $\gamma_2(A|\{A\}) = 1$
- Monotonicity: $\gamma_2(A|Z \cup Z') \le \gamma_2(A|Z)$
- Linearity: $\gamma_2(sA|Z) = s\gamma_2(A|Z)$
  $\gamma_2(A|sZ) = \frac{1}{s}\gamma_2(A|Z)$
- Composition: $\gamma_2(A|Z) \le \gamma_2(A|Y) \max_j \gamma_2(Y_j|Z)$
- Tensor product: $\gamma_2(A \otimes B|Y \otimes Z) = \gamma_2(A|Y)\gamma_2(B|Z)$
- Direct sum: $\gamma_2(A \oplus B|\{Y_j \oplus Z_j\}) = \max\{\gamma_2(A|\{Y_j\}), \gamma_2(B|\{Z_j\})\}$
- Entry-wise filtering: $\gamma_2(A \circ X|Z \circ X) \le \gamma_2(A|Z) \le \gamma_2(A|\{Z_j \circ X\})\gamma_2(X)$
  $\gamma_2(A \circ X|Z) \le \gamma_2(A|Z)\gamma_2(X)$
- If Z_2 is a submatrix of Z_1: $\gamma_2(A|\{Z_1, Z_2\}) = \gamma_2(A|\{Z_1\})$
- If row and column supports disjoint: $\gamma_2(A|\{Z_1, Z_2\}) = \gamma_2(A|\{Z_1 + Z_2\})$

Is the composed algorithm optimal?

Deterministic: ✓

Nondeterministic: ✗

Randomized: ✗

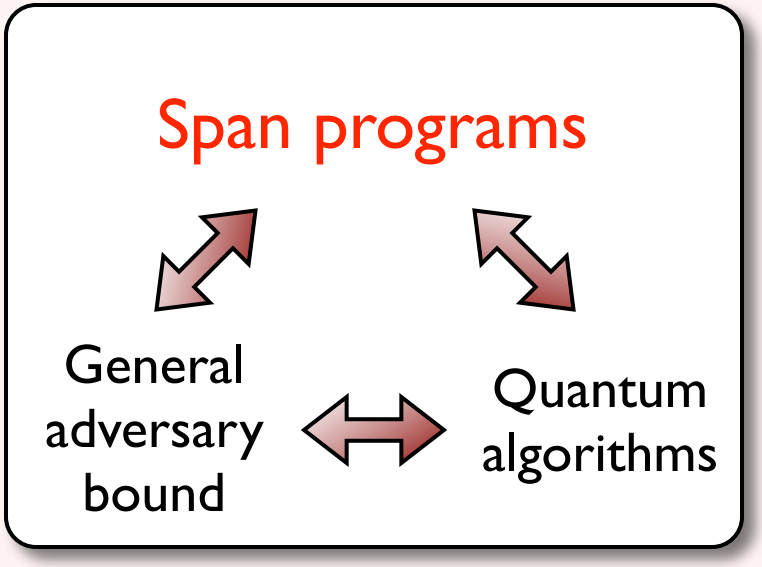
Quantum: ✓

$$Q(f \circ \vec{g}) = \Theta(Q(f)Q(g))$$

with a new notion of composition
… compose SDP solutions

$$\mathrm{Adv}(f \circ \vec{g}) = \mathrm{Adv}(f)\mathrm{Adv}(g)$$

Characterizes query complexity
for read-once formulas


Span programs
General adversary bound
Quantum algorithms

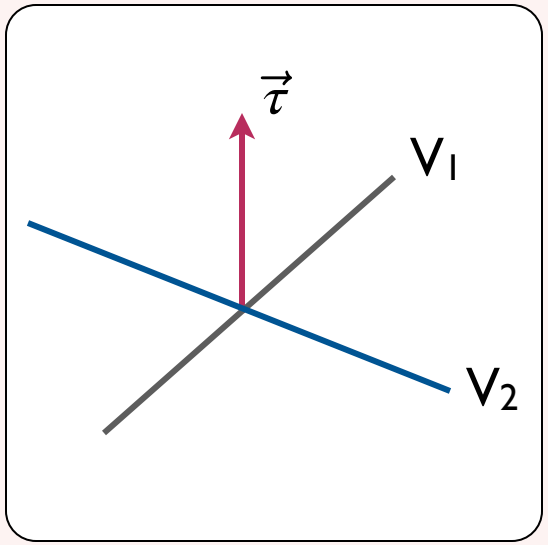
Span program

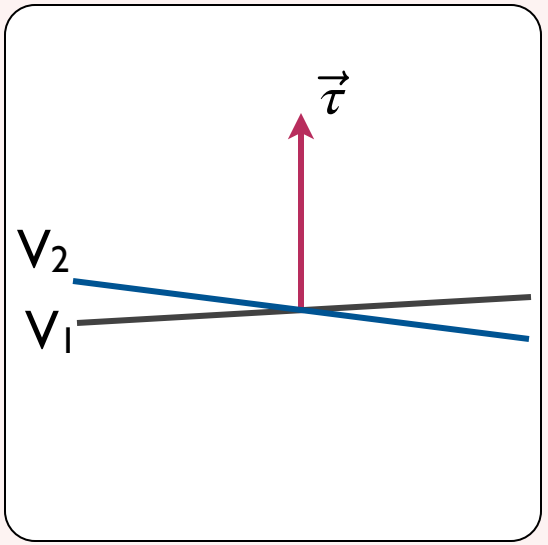
[Karchmer, Wigderson '93]

$$P = \left(\text{target vector } \vec{\tau}, \text{subspaces } V_1, \ldots, V_n \right) : \{0,1\}^n \to \{0,1\}$$

$$P(x) = \begin{cases} 1 & \text{if } \vec{\tau} \in \text{Span}(\{V_j : x_j = 1\}) \\ 0 & \text{otherwise} \end{cases}$$

Examples





st-connectivity span program

Input: Graph G

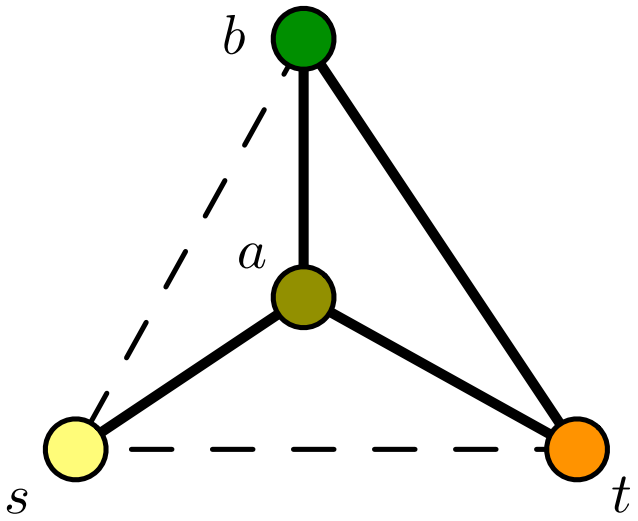
given by adjacency matrix, an $\binom{n}{2}$ bit string

Problem: Is there a path from *s* to *t*?

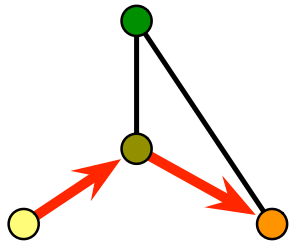
target vector $\vec{\tau} = |t\rangle - |s\rangle = \begin{matrix} s \\ \\ \\ \\ t \end{matrix}\begin{pmatrix} -1 \\ 0 \\ \vdots \\ 0 \\ +1 \end{pmatrix} \in \mathbf{R}^n$

input vector for edge (u,v): $|u\rangle - |v\rangle$

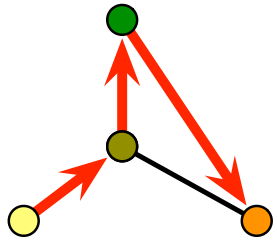
"Witnesses" to P(G)=1 ⇔ balanced s-t flows



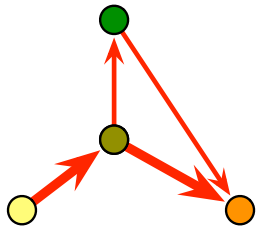
$$\begin{array}{r} |a\rangle - |s\rangle \\ +\ |t\rangle - |a\rangle \\ \hline |t\rangle - |s\rangle \end{array}$$ ✓



$$\begin{array}{r} |a\rangle - |s\rangle \\ |b\rangle - |a\rangle \\ +\ |t\rangle - |b\rangle \\ \hline |t\rangle - |s\rangle \end{array}$$ ✓



$$\begin{array}{r} |a\rangle - |s\rangle \\ \frac{1}{3}(|b\rangle - |a\rangle) \\ \frac{1}{3}(|t\rangle - |b\rangle) \\ +\ \frac{2}{3}(|t\rangle - |a\rangle) \\ \hline |t\rangle - |s\rangle \end{array}$$ ✓

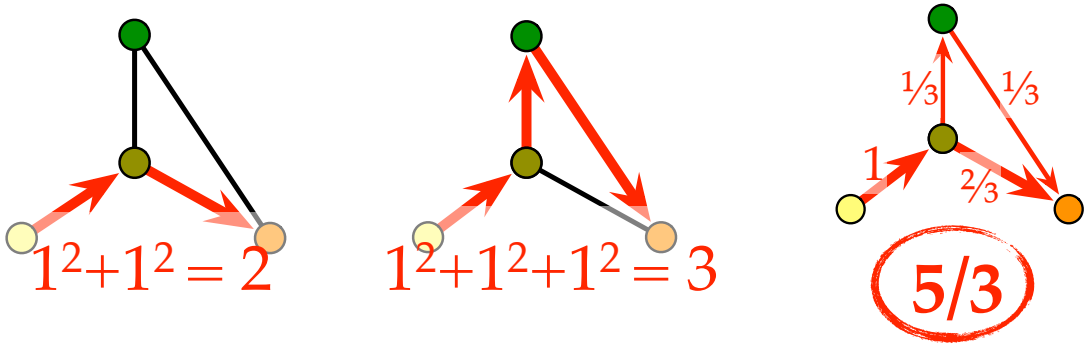


Witness size(P) = $\sqrt{\max_{G:\,P(G)=1} \text{wsize}(P,G) \cdot \max_{G:\,P(G)=0} \text{wsize}(P,G)}$

Case P(G)=1

wsize = Energy of electrical flow

= $R_{st}(G) \le \text{distance}(s,t) \le n$



Case P(G)=0

wsize = Cut size

$\le$ # possible edges = $\binom{n}{2}$

Witness size(P) $= \sqrt{\max_{G:\,P(G)=1} \text{wsize}(P,G) \cdot \max_{G:\,P(G)=0} \text{wsize}(P,G)}$

Case P(G)=1

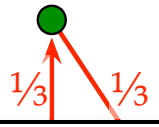
wsize = Energy of electrical flow

= $R_{st}(G) \leq$ distance(s,t) $\leq n$

Case P(G)=0

wsize = Cut size

$\leq$ # possible edges = $\binom{n}{2}$



Theorem: General adversary bound $\left(f \right) =$ minimum witness size of any span program for f

New *st*-connectivity quantum query algorithm

$O(\sqrt{(M R_{st})}) = O(n\sqrt{d})$ queries,

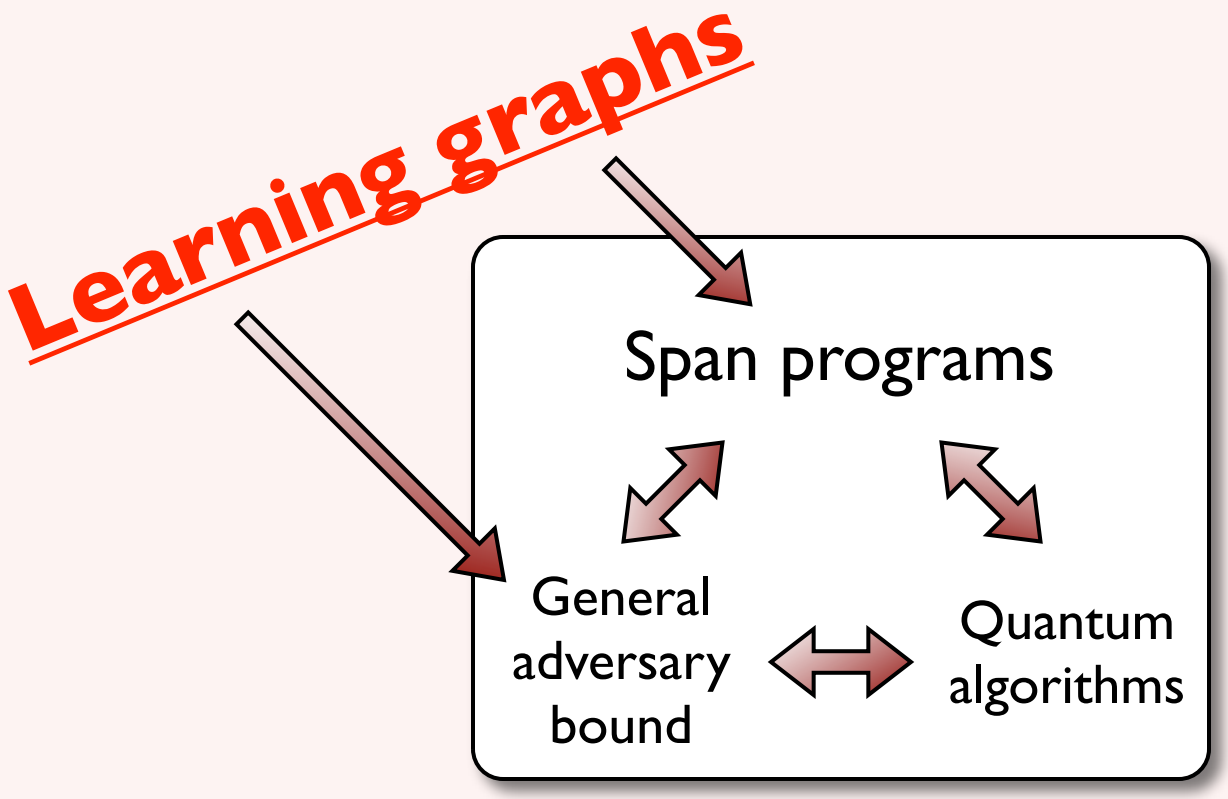
M = # *possible* edges

if s and t are promised either to be within distance d, or be disconnected

Log space

- space = # qubits = log(# input vectors)
- shorter than st-path
- qubits expensive

Efficiently implementable


Learning graphs
Span programs
General adversary bound
Quantum algorithms

General adversary bound

$$= \min_{\{p_x[j] \in \mathbb{R}^m\}} \max_x \sum_j \|p_x[j]\|^2$$

$$\text{s.t.} \sum_{j: x_j \neq y_j} p_x[j] \cdot p_y[j] = 1 \text{ if } f(x) \neq f(y)$$

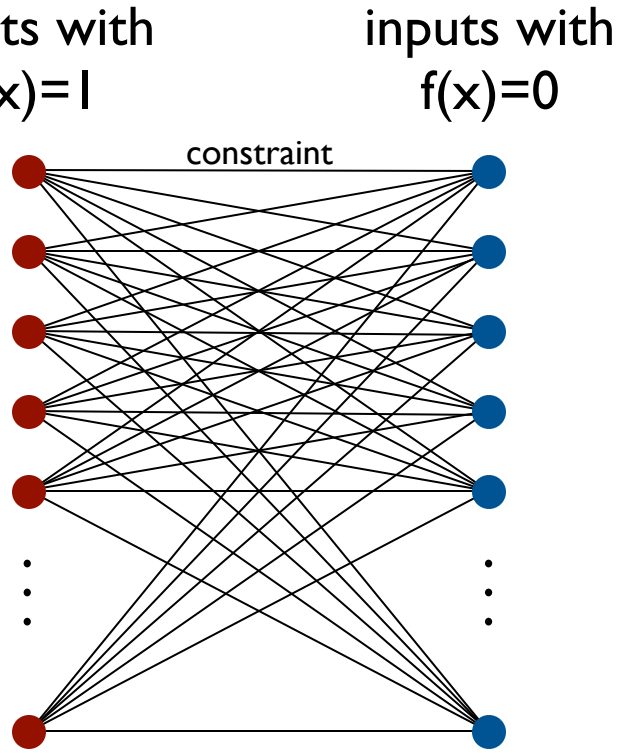
Solving this SDP is hard!

1. Exponentially large
2. Exact *equality* constraints

Idea:

st-connectivity solution based on flows & cuts

flow · cut = net flow across cut = 1

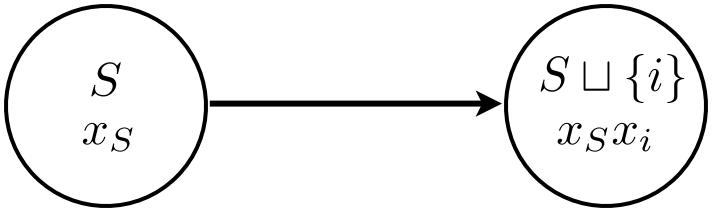


Learning graphs: Reductions to *st*-connectivity

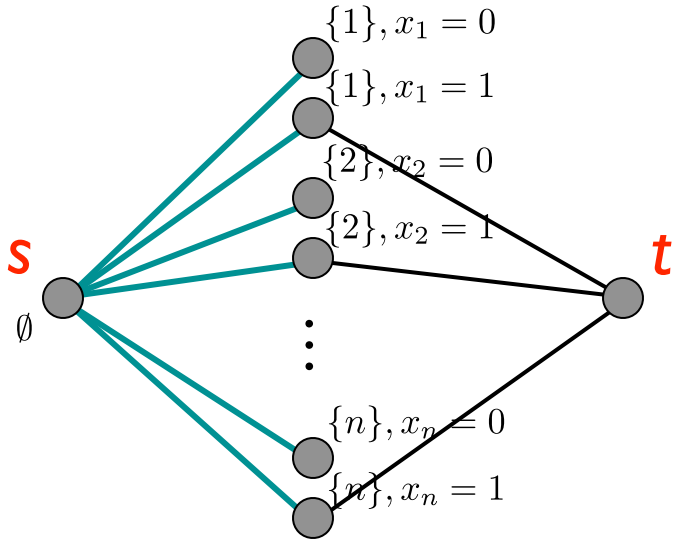
- Vertices: $\left\{(S, x_S) : \begin{array}{l} \text{subset } S \subseteq [n] \\ \text{partial assignment } x_S \in [k]^S \end{array}\right\}$

  source = $\varnothing$ sinks = vertices with a certificate for f

- Edges:

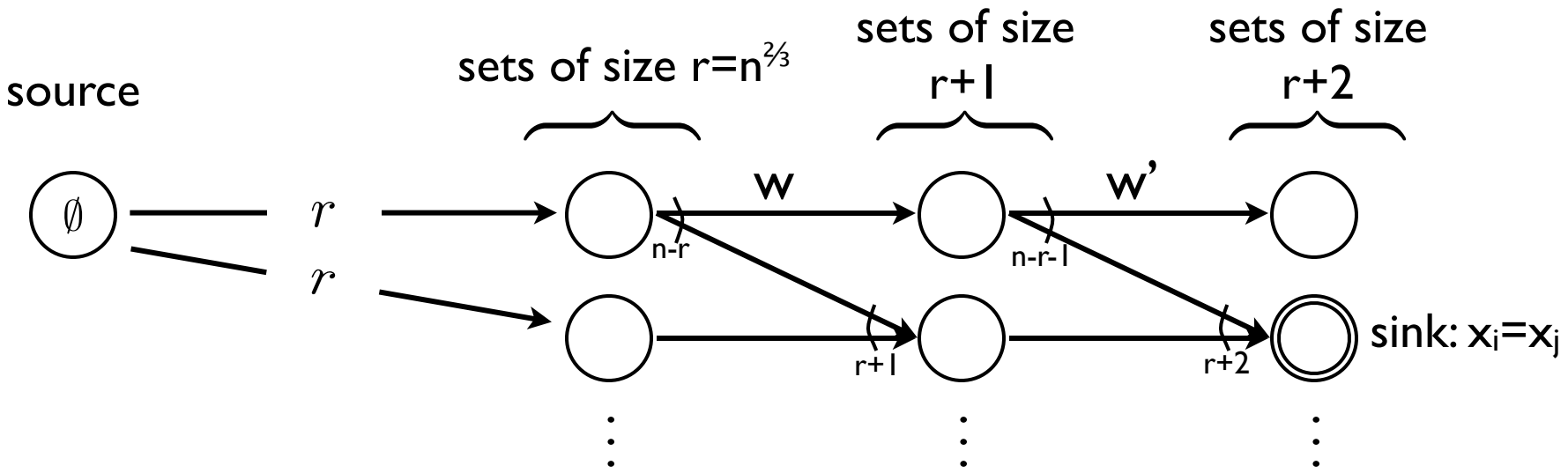


Example: Grover search

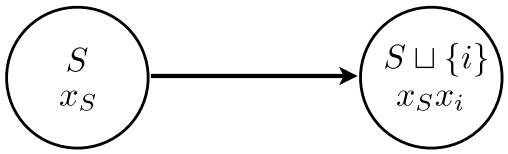


Element Distinctness

Given $x \in [k]^n$, is there a pair with $x_i = x_j$?



Edges



True case: source ~ sink

flow energy = $\frac{r}{\binom{n-2}{r}} + \frac{1/w}{\binom{n-2}{r}} + \frac{1/w'}{\binom{n-2}{r}}$

False case: source ≁ sink

cut cost = $\binom{n}{r} r + \binom{n}{r} (n-r)w + \binom{n}{r+1} (n-r-1)w'$

New quantum algorithms

General formula evaluation	[Rei11b]
Almost-balanced formulas	[RŠ12, Rei11c]
AND-OR formulas ("game trees")	[Rei11a]
Large AND-OR formulas with inputs satisfying certain promises	[ZKH12, Kim12]
Triangle detection	[Bel12b] ☆
Related graph problems, e.g., subgraph detection	[Zhu12, LMS11] ☆
s-t connectivity, certain subgraph-detection & subgraph/not-a-minor problems	[BR12]
Graph collision	[GI12]
k-distinctness and 3-distinctness	[BL11, Bel12a] ☆
Matrix rank	[Bel11]

super-polynomial quantum speedups

☆=Learning graphs

Other applications

Merkle puzzles [BHKKLS11]

State conversion [AMRR11, ORR11, LMRSŠ11]

Direct-product theorem [LR11]

Quantum computing at USC

Roadblocks for a quantum computer

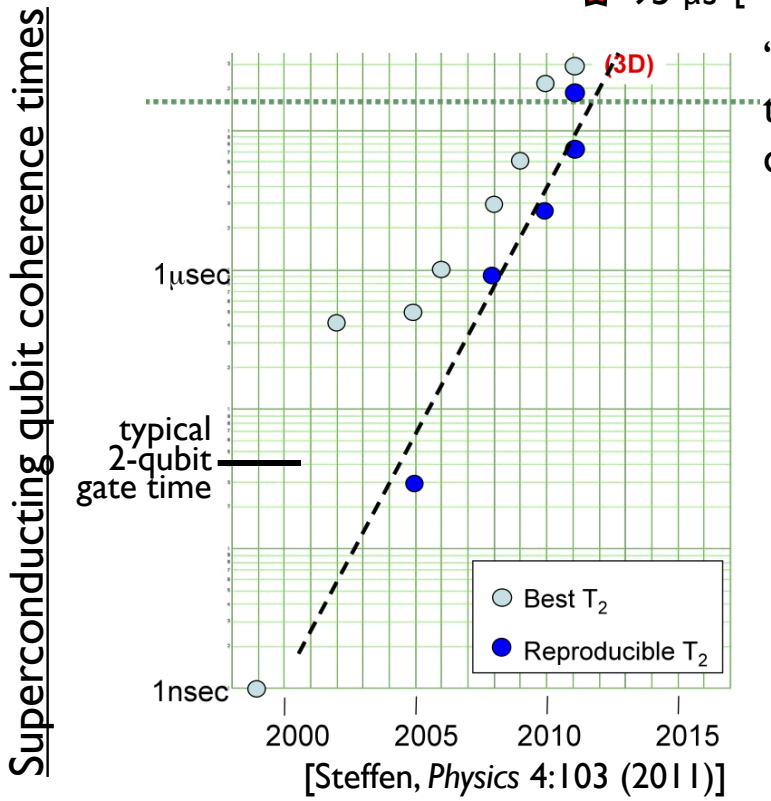
Noise

Architecture

Scaling

Noise

★ 95 μs [Rigetti et al., 2012]



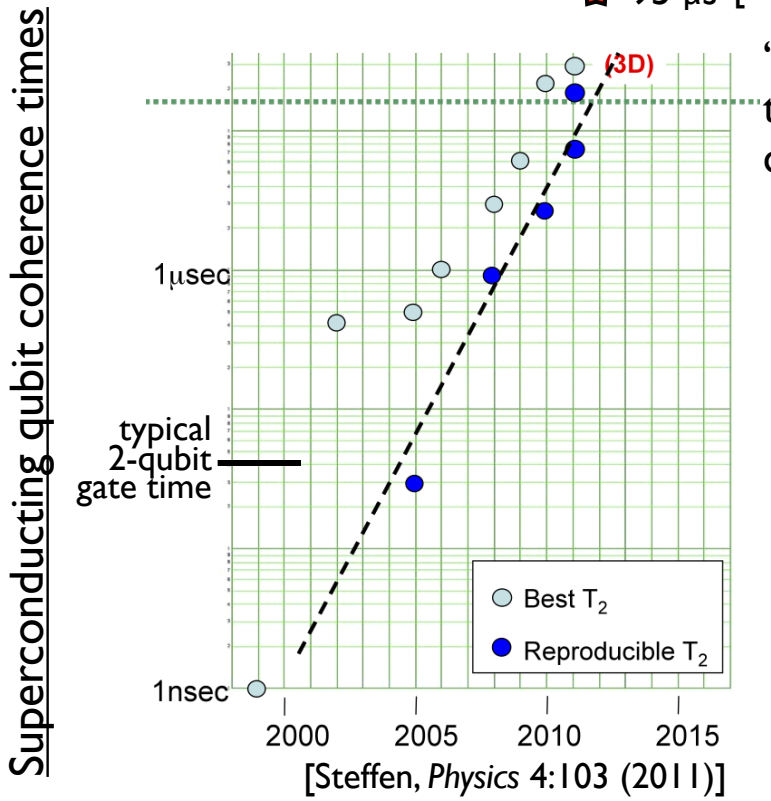
"This technology [is] a strong candidate for the immediate construction of prototype quantum processors with 10-1000 qubits."

Architecture

Scaling

Noise

★ 95 μs [Rigetti et al., 2012]



"This technology [is] a strong candidate for the immediate construction of prototype quantum processors with 10-1000 qubits."

Architecture

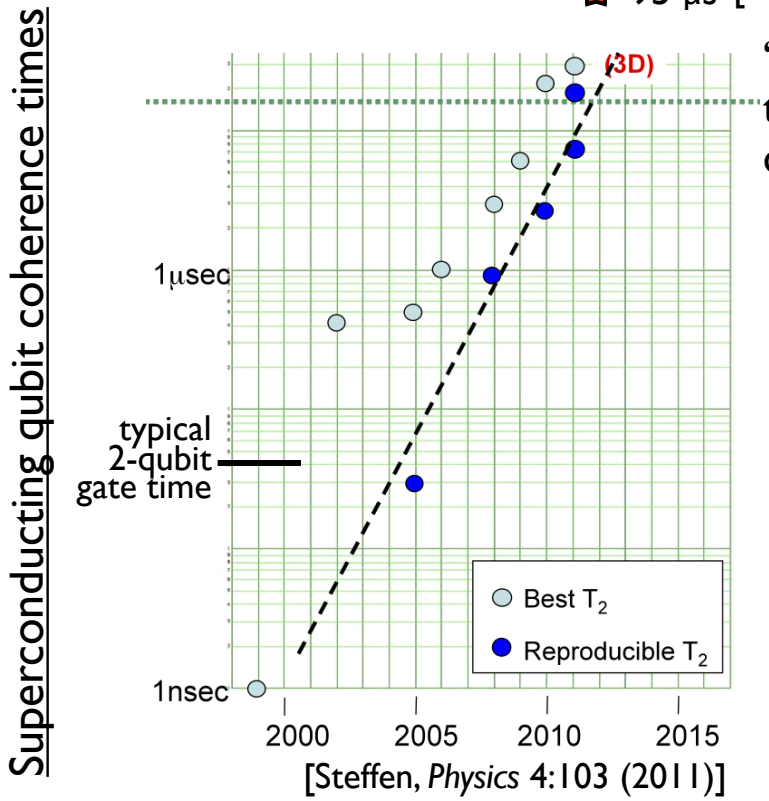
"Aggressive Optimization and Resource Estimation of Next-Generation Quantum Computing Systems"

– USC, Berkeley, UCSB, Caltech, Princeton, Alcatel-Lucent, SC Solutions

Scaling

Noise

★ 95 μs [Rigetti et al., 2012]



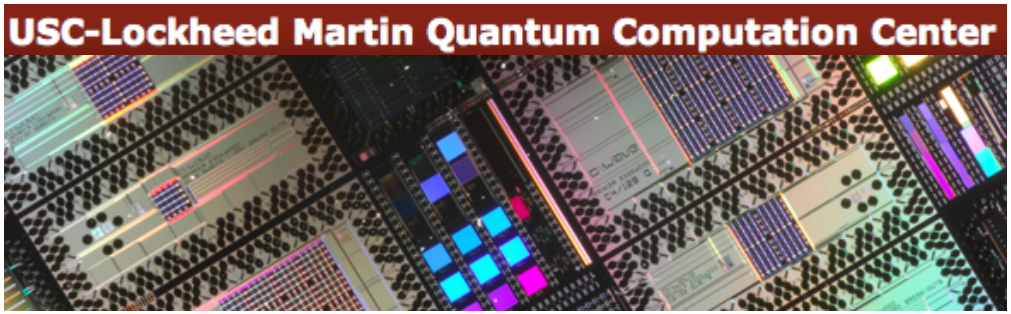
"This technology [is] a strong candidate for the immediate construction of prototype quantum processors with 10-1000 qubits."

Architecture

"Aggressive Optimization and Resource Estimation of Next-Generation Quantum Computing Systems"

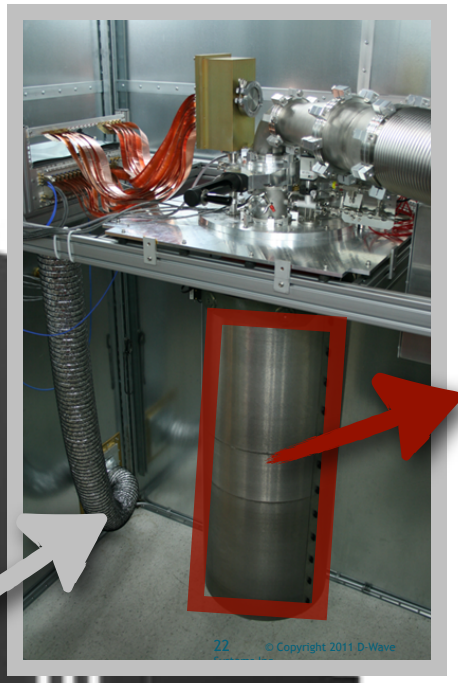
– USC, Berkeley, UCSB, Caltech, Princeton, Alcatel-Lucent, SC Solutions

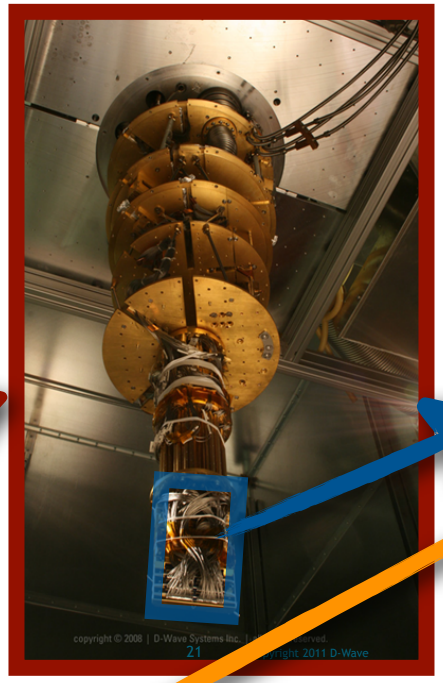
Scaling

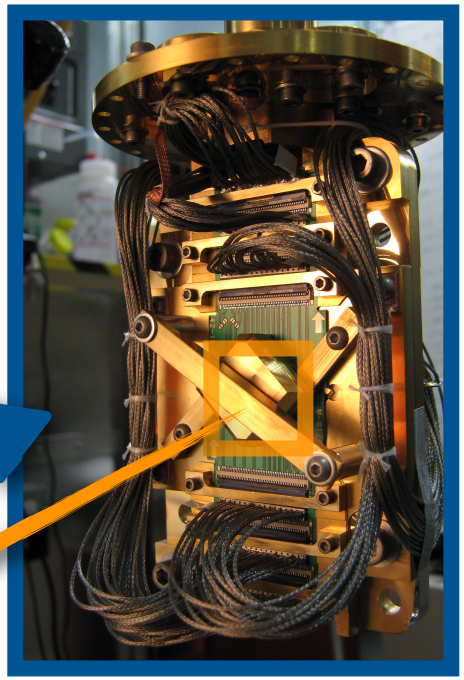


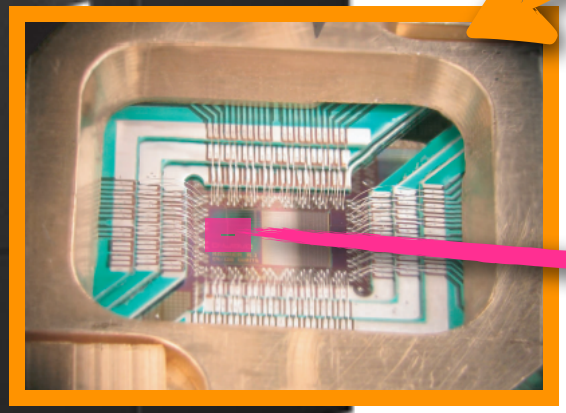
D-Wave One = 128-qubit adiabatic quantum computation device

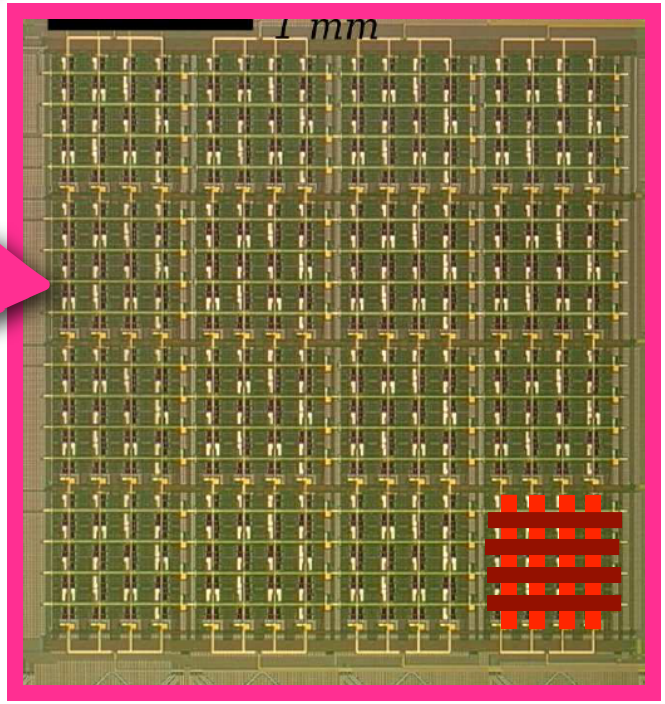

D-Wave

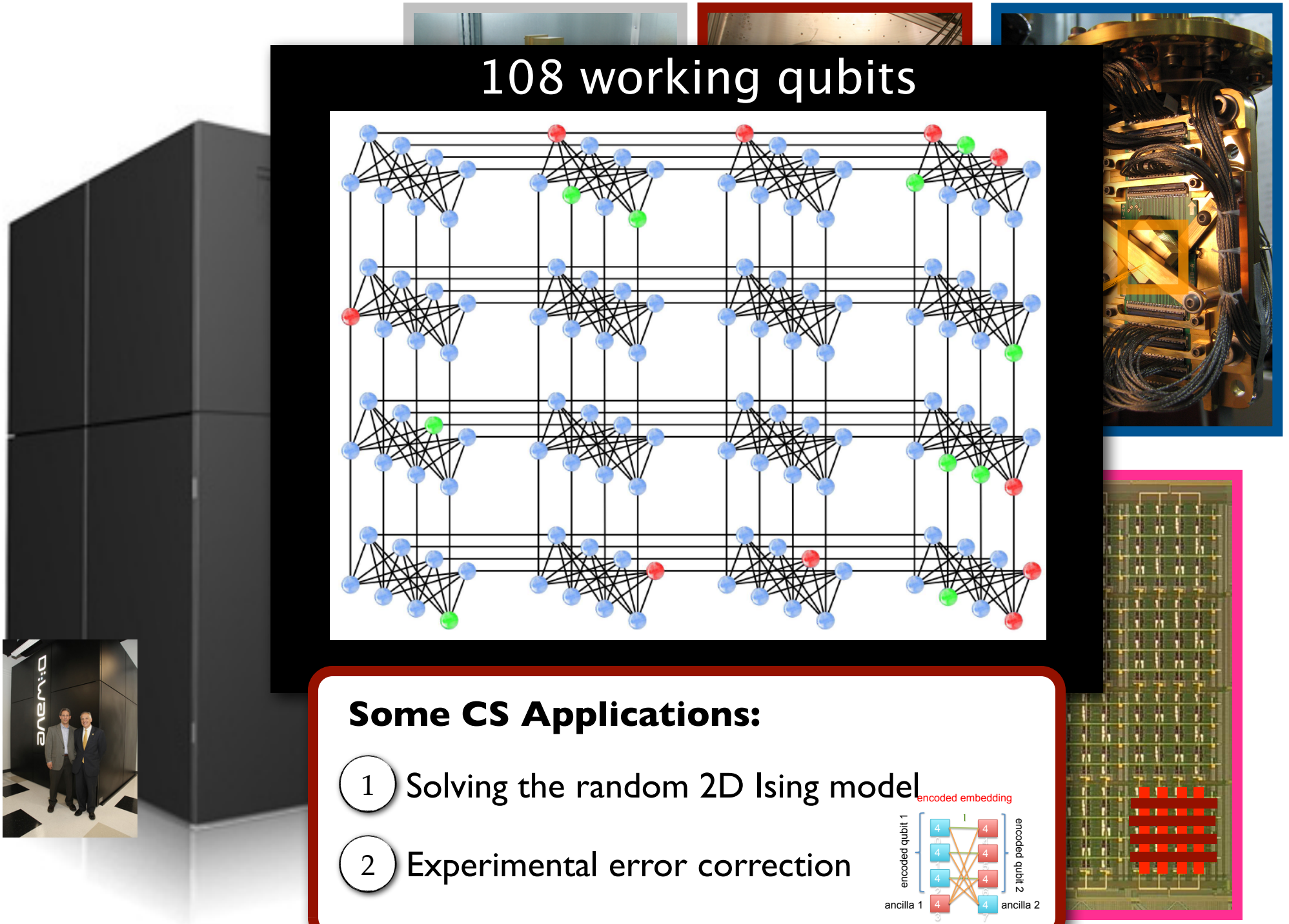

22
© Copyright 2011 D-Wave


21






1 mm


108 working qubits
Some CS Applications:
1 Solving the random 2D Ising model
encoded embedding
encoded qubit 1
encoded qubit 2
ancilla 1
ancilla 2
2 Experimental error correction
D-Wave

USC

Center for Quantum Information Science and Technology (CQIST)

USC-Lockheed Martin Quantum Computation Center

EE

+ more

Physics

Courses

- EE 520: Intro. Quantum Information Processing (Brun)
- EE 539: Engineering Quantum Mechanics (Levi)
- EE 587: Nonlinear & Adaptive Control (Jonckheere)
- EE 599: Quantum Error Correction (Lidar)
- EE 599: Adiabatic Quantum Computing (Boixo)
- EE 599: Quantum Algorithms (Reichardt)
- Phys 510: Computational Physics (Haas)
- Phys 720: Quantum Information Science & Many-Body Physics (Zanardi)
- Chem 599: Theory of Open Quantum Systems (Lidar)
- Chem 599: The Cutting Edge in Quantum Information Science (Lidar)

CS theorists: Shang-Hua Teng, David Kempe, Fei Sha, Shaddin Dughmi, Len Adleman, Dorit Hochbaum, Ming-Deh Huang, Michael Waterman, Hamid Nazerzadeh, Alejandro Toriello, Kristina Lerman, Milind Tambe, Yan Liu, Bhaskar Krishnamachari, Viktor Prasanna

Caltech:

iQIM INSTITUTE FOR QUANTUM INFORMATION AND MATTER

UCSB: