

# Introduction and Overview (Preskills Notes)

John von Neumann

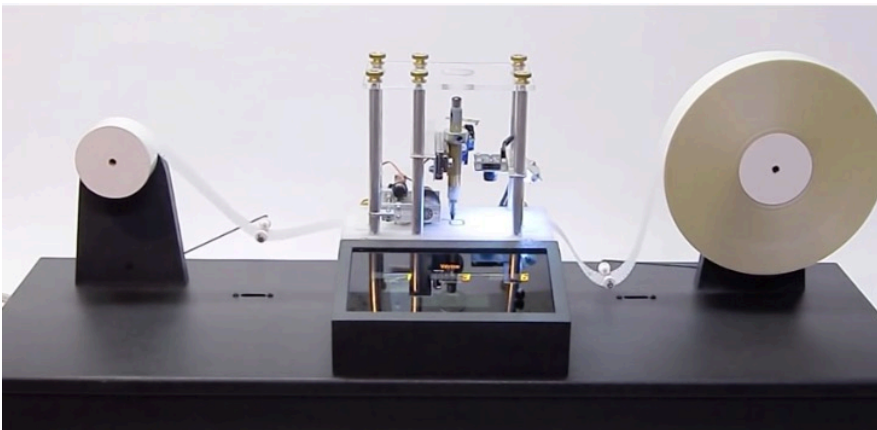
Physics of Information: Alan Turing

Notions: What is a computation ?  
What is computable

Formulation of Computer Science  
that is **Device Independent**



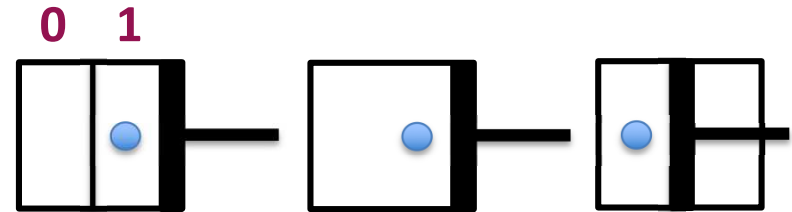
1937 Turing Machine:



<https://www.youtube.com/watch?v=E3keLeMwfHY>

Landaur: **Information is Physical!**

Example: **Erasure = Dissipation**



Entropy:  $\Delta S_{gas} = -k \ln 2$

Work:  $W = kT \ln 2 = 0.96 \times 10^{-23} \frac{J}{K} \cdot 300K$   
 $\sim 3 \times 10^{-21} J \sim 0.02 eV$

Is there a way around it ?

**Reversible Computation!**

But we need a different gate set !

# Introduction and Overview (Preskills Notes)

John von Neumann

Alan Turing

## Wikipedia:

A Turing Machine (TM) is a mathematical model of computation describing an abstract machine that manipulates symbols on a strip of paper according to a table of rules.

The TM operates on an infinite tape divided into cells, each of which can hold a symbol drawn from a finite set.

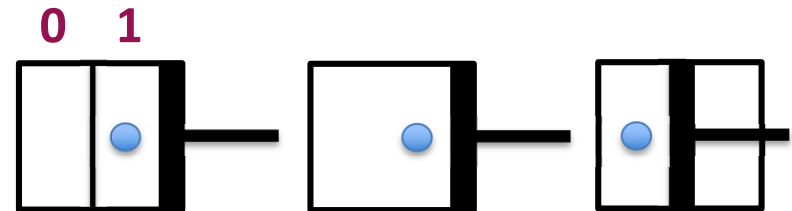
At each step the head reads the symbol in the cell. Then, based on the symbol and the TM's present state, the machine writes a symbol in the cell, and moves the head one step to the left or the right, or halts the computation.

## Church – Turing Thesis:

Everything that is computable can be computed on a Turing Machine with at most polynomial overhead.

Landaur: **Information is Physical!**

Example: **Erasure = Dissipation**



Entropy:  $\Delta S_{\text{gas}} = -k \ln 2$

Work:  $W = kT \ln 2 = 0.96 \times 10^{-23} \frac{\text{J}}{\text{K}} \cdot 300 \text{K}$   
 $\sim 3 \times 10^{-21} \text{J} \sim 0.02 \text{eV}$

Is there a way around it ?

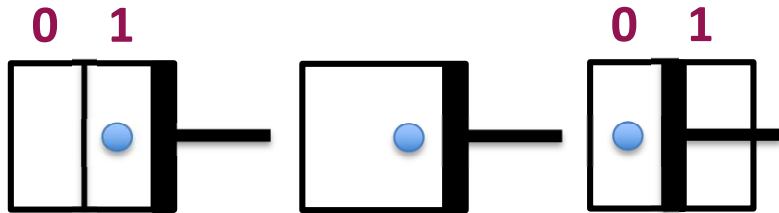
**Reversible Computation!**

But we need a different gate set!

# Introduction and Overview (Preskills Notes)

**Landaur:** Information is Physical!

**Example:** Erasure = Dissipation



**Entropy:**  $\Delta S_{gas} = -k \ln 2$

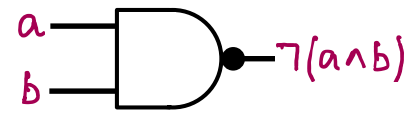
**Work:**  $W = kT \ln 2 = 0.96 \times 10^{-23} \frac{J}{K} \cdot 300K$   
 $\sim 3 \times 10^{-21} J \sim 0.02 eV$

Is there a way around it ?

**Reversible Computation!**

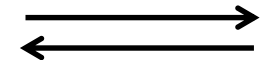
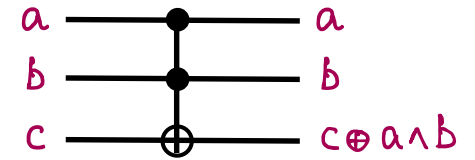
But we need a different gate set!

**NAND Gate:**



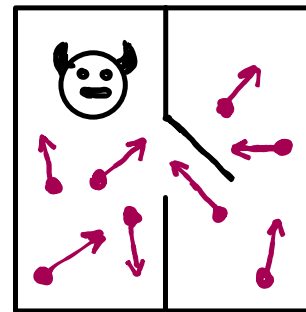
irreversible

**Toffoli Gate:**



reversible

**Maxwells Demon:**



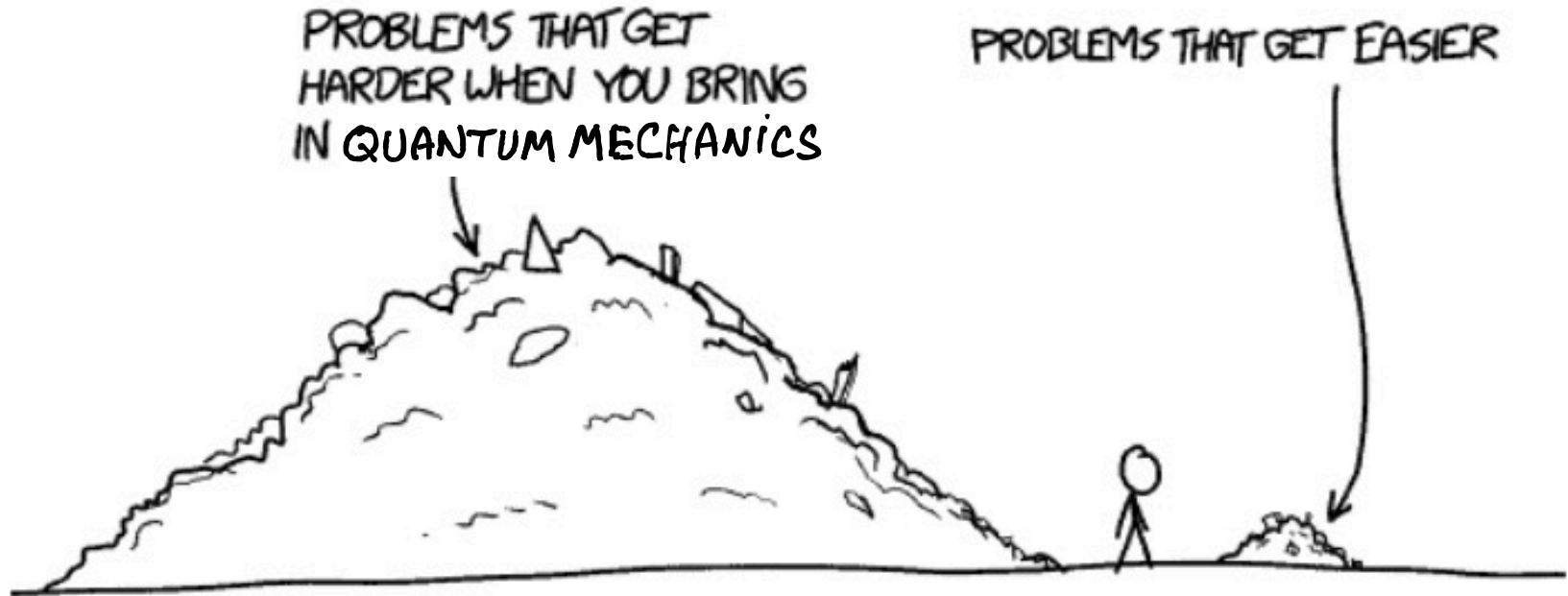
**Information is Physical!**

**Quantum Information**

**Carl Caves:** Quantum States are states of knowledge

**Physics is Information!**

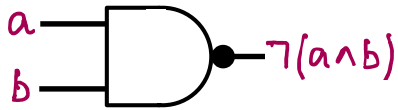
# Introduction and Overview (Preskills Notes)



Source: xkcd.com

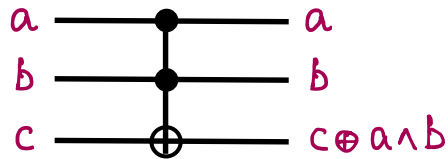
# Introduction and Overview (Preskills Notes)

NAND Gate:



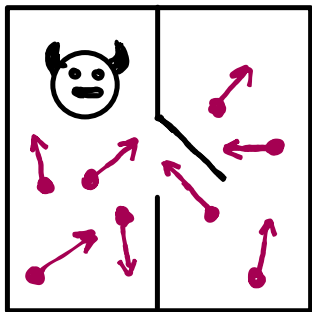
irreversible

Toffoli Gate:



reversible

Maxwells Demon:



Information is Physical!

Quantum Information

Carl Caves: Quantum States are states of knowledge

Physics is Information!

## New properties of QM

Measurement:

$$[A, B] \neq 0 \Rightarrow \Delta A \Delta B \geq \frac{\hbar}{2} |\langle [A, B] \rangle|$$

Acquire Info  $\rightarrow$  Disturb system

Randomness:

Outcome fundamentally unpredictable

"Collapse" of wavefunction

Cannot determine state of a quantum system if initially unknown

Cannot Copy  
No cloning theorem

Entanglement:

Non-local correlations

pure state, entangled

$$|\psi\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$$

$$\rho = \frac{1}{2} (|00\rangle\langle 00| + |11\rangle\langle 11|)$$

mixed state, not entangled

# Introduction and Overview (Preskills Notes)

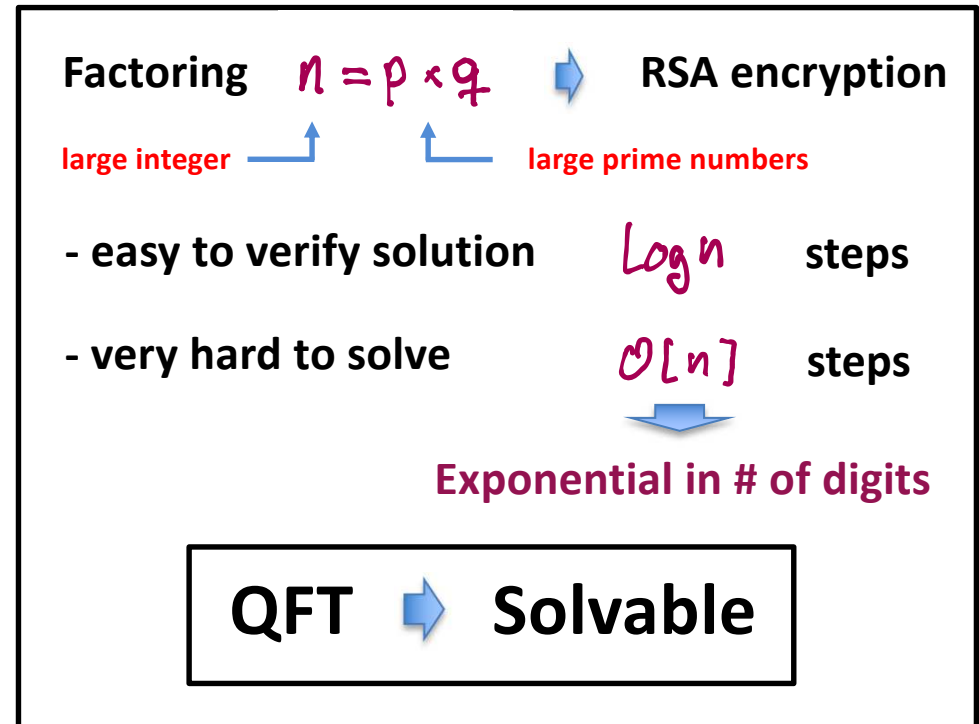
## Quantum Computing

Does QM impact Computation?

Peter Shor (1994): YES!  $\rightarrow$  Quantum Fourier Transform  
 $\downarrow$   
Factoring !

DFT on $N$ bits	$\mathcal{O}[(2^N)^2]$	steps
FFT on “	$\mathcal{O}[N2^N]$	“
QFT on “	$\mathcal{O}[N \log N]$	“

## Efficient Factoring





Preskill Ch. 1, p. 5-6  $T \propto e^{1.9(\log n)^{1/3}} e^{(\log \log n)^{2/3}}$   
Best Classical Algorithm

# Introduction and Overview (Preskills Notes)

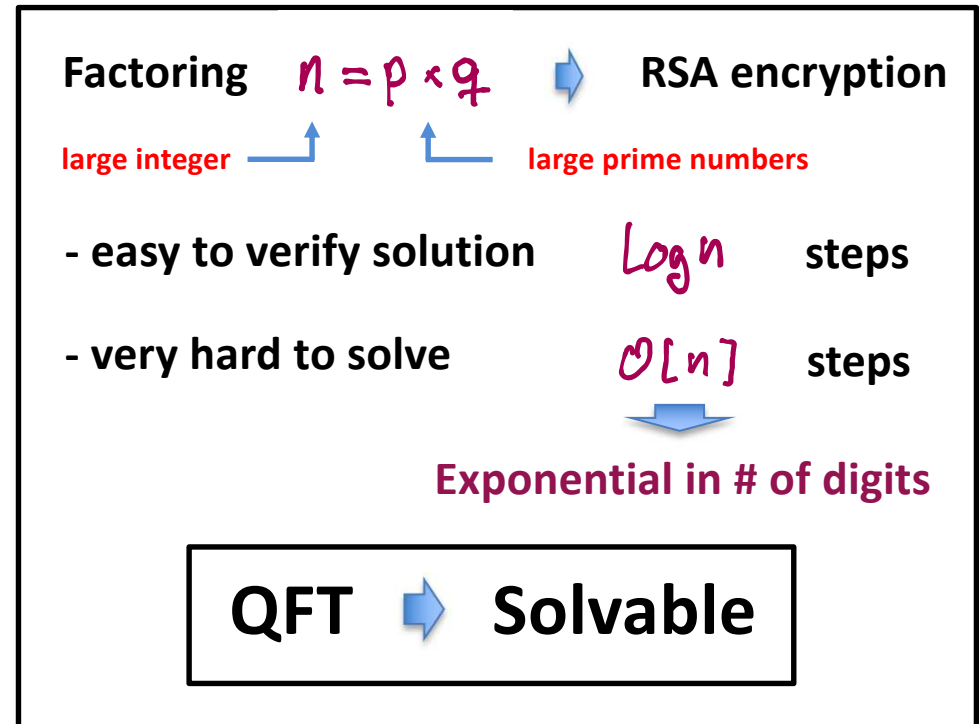
## Quantum Computing

Does QM impact Computation?

Peter Shor (1994): YES!  Quantum Fourier Transform  
 Factoring !

DFT on $N$ bits	$\mathcal{O}[(2^N)^2]$	steps
FFT on “	$\mathcal{O}[N2^N]$	“
QFT on “	$\mathcal{O}[N \log N]$	“

## Efficient Factoring



Preskill Ch. 1, p. 5-6  $T \propto e^{1.9(\log n)^{1/3}} e^{(\log \log n)^{2/3}}$

(1998) 130 digits in 1 month



400 digits in  $10^{10}$  years



(2022) 24 yrs = 16 Moores Law doublings

$2^{16} = 65,536$   400 digits  $\sim$  150kYrs

# Introduction and Overview (Preskills Notes)

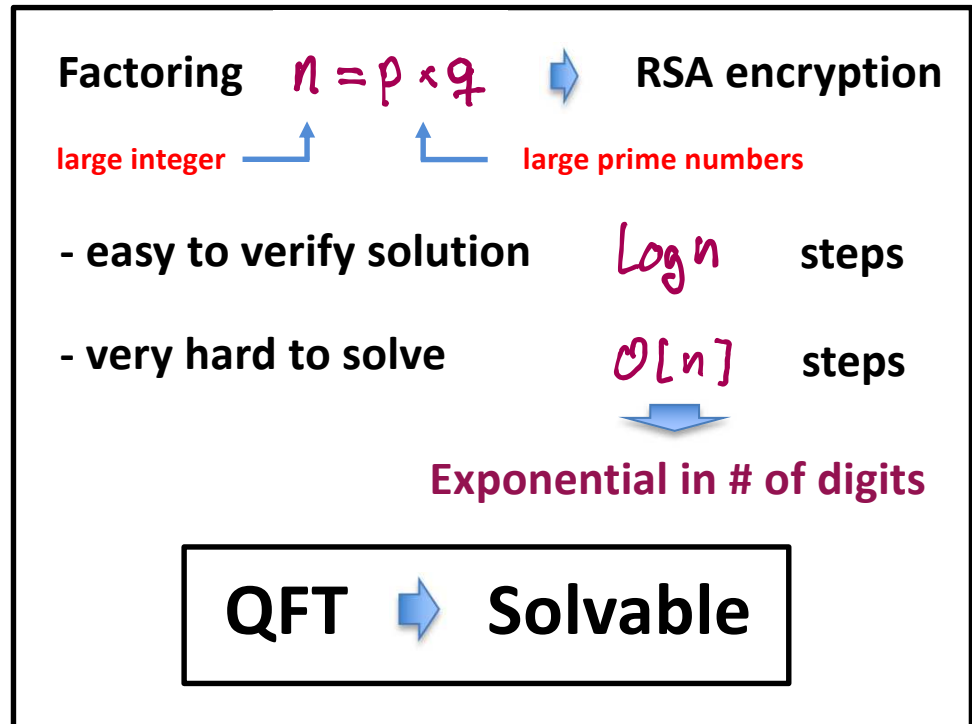
## Quantum Computing

Does QM impact Computation?

Peter Shor (1994): YES!  Quantum Fourier Transform  
 Factoring !

DFT on $N$ bits	$\mathcal{O}[(2^N)^2]$	steps
FFT on “	$\mathcal{O}[N2^N]$	“
QFT on “	$\mathcal{O}[N \log N]$	“

## Efficient Factoring



Preskill Ch. 1, p. 5-6  $T \propto e^{1.9(\log n)^{1/3}} e^{(\log \log n)^{2/3}}$

(1998) 130 digits in 1 month



400 digits in  $10^{10}$  years

(2022) 24 yrs = 16 Moores Law doublings

$2^{16} = 65,536$   400 digits  $\sim$  150kYrs



# Introduction and Overview (Preskills Notes)

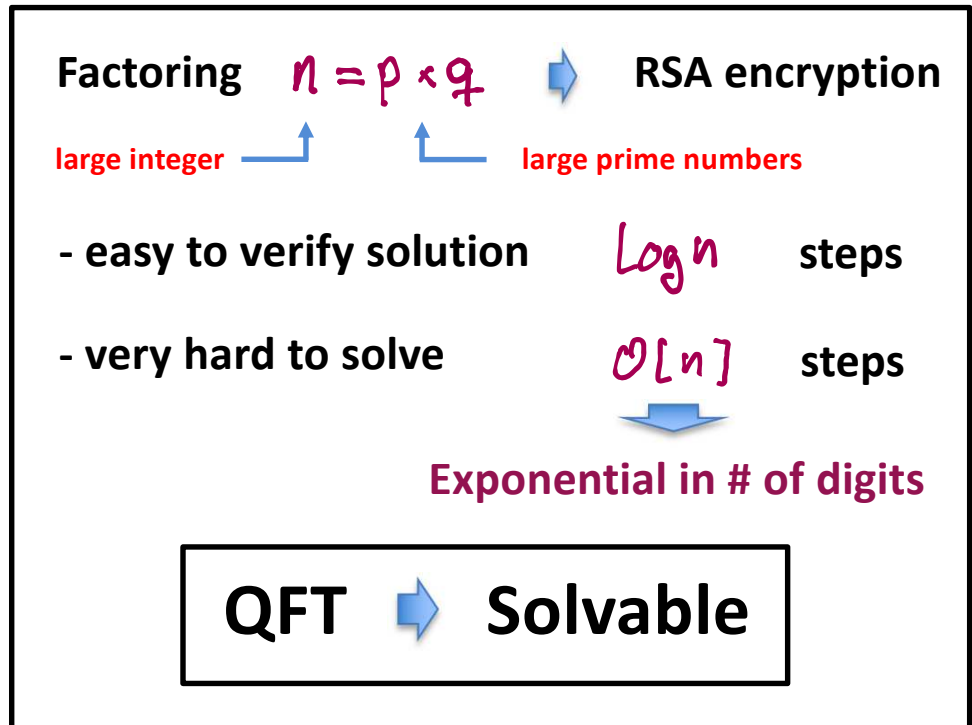
## Quantum Computing

Does QM impact Computation?

Peter Shor (1994): YES!  $\rightarrow$  Quantum Fourier Transform  $\rightarrow$  Factoring !

DFT on $N$ bits	$\mathcal{O}[(2^N)^2]$	steps
FFT on “	$\mathcal{O}[N2^N]$	“
QFT on “	$\mathcal{O}[N \log N]$	“

## Efficient Factoring



Preskill Ch. 1, p. 5-6  $T \propto e^{1.9(\log n)^{1/3}} e^{(\log \log n)^{2/3}}$

(1998) 130 digits/month

$\rightarrow$  400 digits/  $10^{10}$  years Polynomial in # of digits

Shors algorithm:  $\mathcal{O}[(\log n)^3]$   $\leftarrow$

130 digits/mo.  $\rightarrow$  400 digits/3 yrs if Quantum

# Introduction and Overview (Preskills Notes)

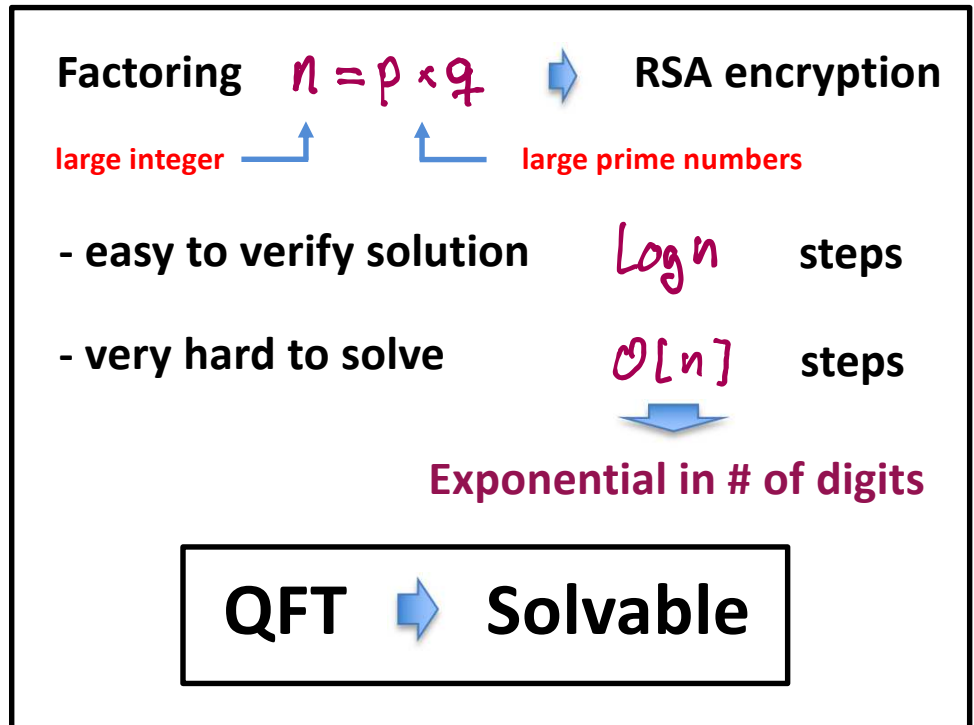
## Quantum Computing

Does QM impact Computation?

Peter Shor (1994): YES!  $\rightarrow$  Quantum Fourier Transform  
 $\downarrow$   
**Factoring !**

DFT on $N$ bits	$\mathcal{O}[(2^N)^2]$	steps
FFT on “	$\mathcal{O}[N2^N]$	“
QFT on “	$\mathcal{O}[N \log N]$	“

## Efficient Factoring



Preskill Ch. 1, p. 5-6  $T \propto e^{1.9(\log n)^{1/3}} e^{(\log \log n)^{2/3}}$

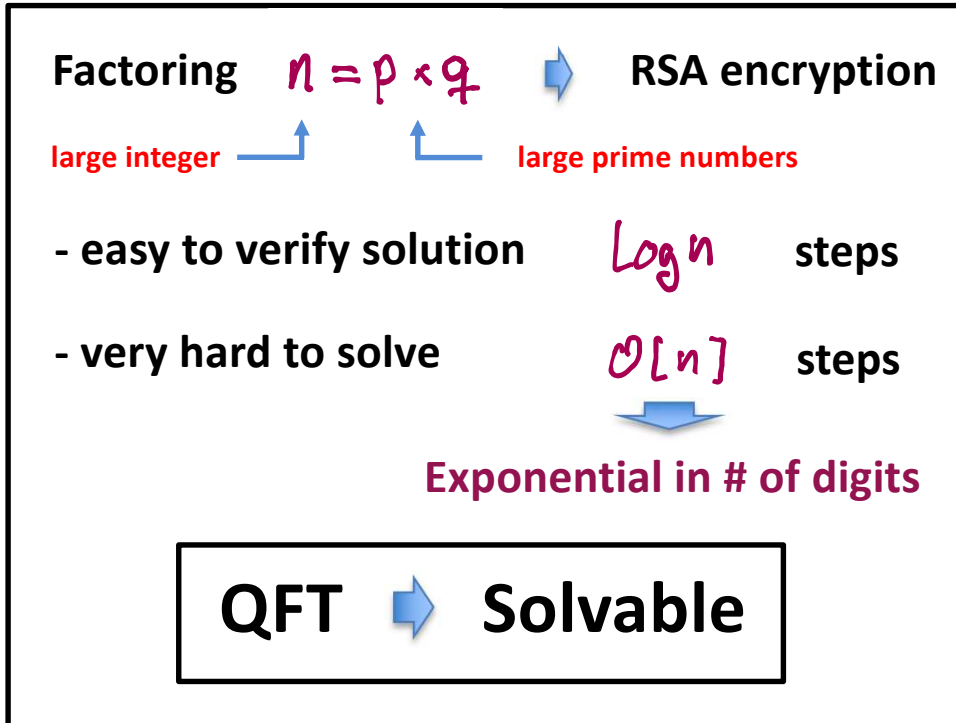
(1998) 130 digits/month

$\downarrow$   
 400 digits/  $10^{10}$  years Polynomial in # of digits

Shors algorithm:  $\mathcal{O}[(\log n)^3]$   $\leftarrow$

130 digits/mo.  $\rightarrow$  400 digits/3 yrs if Quantum

## Efficient Factoring



Preskill Ch. 1, p. 5-6  $T \propto e^{1.9(\log n)^{1/3}} e^{(\log \log n)^{2/3}}$

(1998) 130 digits/month

$\Downarrow$   
400 digits/  $10^{10}$  years

Polynomial  
in # of digits

Shors algorithm:  $\mathcal{O}[(\log n)^3]$

130 digits/mo.  $\rightarrow$  400 digits/3 yrs if Quantum

## Quantum Complexity

## Efficient Factoring

Factoring  $n = p \times q$   $\rightarrow$  RSA encryption

large integer  $\rightarrow$   $\leftarrow$  large prime numbers

- easy to verify solution  $\log n$  steps
- very hard to solve  $\mathcal{O}[n]$  steps

$\downarrow$

Exponential in # of digits

**QFT  $\rightarrow$  Solvable**

Preskill Ch. 1, p. 5-6  $T_{oc} \approx e^{1.9(\log n)^{1/3}} e^{(\log \log n)^{2/3}}$

(1998) 130 digits/month

400 digits/  $10^{10}$  years

Polynomial  
in # of digits

Shors algorithm:  $\mathcal{O}[(\log n)^3]$

130 digits/mo.  $\rightarrow$  400 digits/3 yrs if Quantum

## Quantum Complexity

Benioff, Feynman  $\rightarrow$  Simulating QM is hard

Simulating Physics with computers (1982)

qubit  $\begin{matrix} \text{---} & |1\rangle \\ \text{---} & |0\rangle \end{matrix}$   $| \psi \rangle = a|0\rangle + b|1\rangle$

$N$  qubits  $\rightarrow$   $|01101011\dots 010\rangle = |x\rangle \in \mathcal{E}$

$N$  bit integer =  $x$ : logical basis state  $\uparrow$

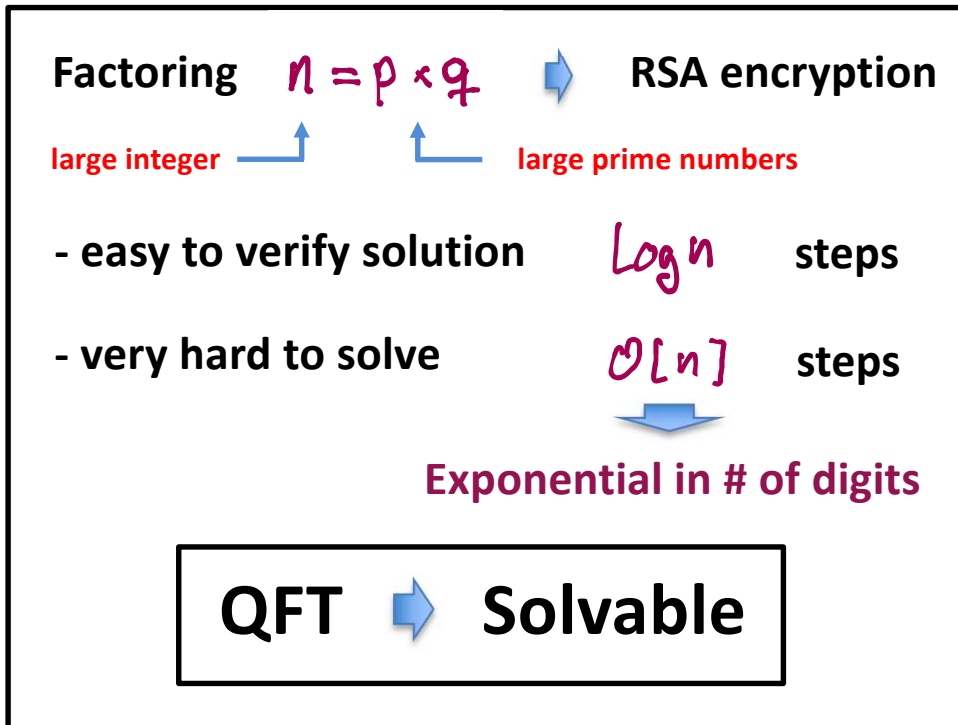
$2^N$  - dimensional Hilbert Space

General State:

$$|\psi\rangle = \frac{1}{2^{N/2}} \sum_{x=0}^{2^N-1} a_x |x\rangle$$

# Introduction and Overview (Preskills Notes)

## Efficient Factoring



Preskill Ch. 1, p. 5-6  $T_{oc} \approx e^{1.9(\log n)^{1/3}} e^{(\log \log n)^{2/3}}$

(1998) 130 digits/month

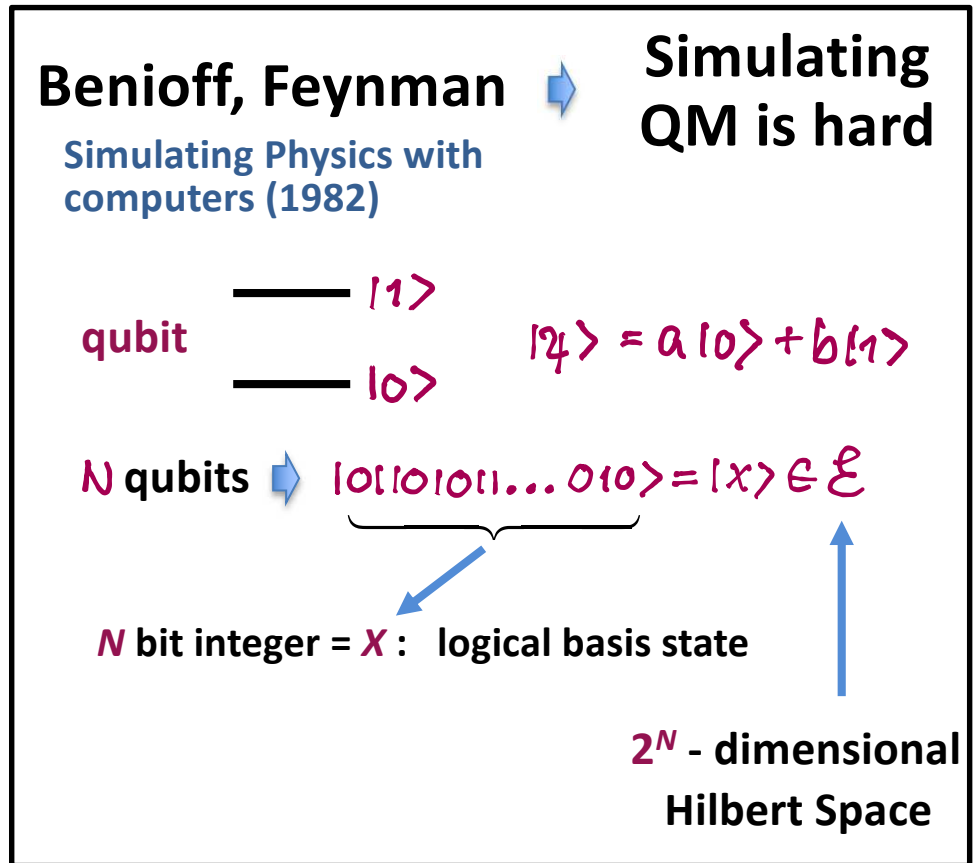
400 digits/  $10^{10}$  years

Polynomial in # of digits

Shors algorithm:  $\mathcal{O}[(\log n)^3]$

130 digits/mo.  $\rightarrow$  400 digits/3 yrs if Quantum

## Quantum Complexity



General State:

$$|\psi\rangle = \frac{1}{2^{N/2}} \sum_{x=0}^{2^N-1} a_x |x\rangle$$

# Introduction and Overview (Preskills Notes)

## Quantum Complexity

Benioff, Feynman



Simulating Physics with computers (1982)

Simulating QM is hard

qubit  $|1\rangle$   
 $|0\rangle$   $|\psi\rangle = a|0\rangle + b|1\rangle$

$N$  qubits  $\Rightarrow |01101011\dots 010\rangle = |x\rangle \in \mathcal{E}$

$N$  bit integer =  $x$ : logical basis state

$2^N$  - dimensional Hilbert Space

General State:

$$|\psi\rangle = \frac{1}{2^{N/2}} \sum_{x=0}^{2^N-1} a_x |x\rangle$$

# Introduction and Overview (Preskills Notes)

## Quantum Complexity

Benioff, Feynman  $\rightarrow$  Simulating QM is hard

Simulating Physics with computers (1982)

qubit  $|1\rangle$   
 $|0\rangle$   $|\psi\rangle = a|0\rangle + b|1\rangle$

$N$  qubits  $\rightarrow |01101011\dots 010\rangle = |x\rangle \in \mathcal{E}$

$N$  bit integer =  $x$ : logical basis state

$2^N$  - dimensional Hilbert Space

General State:

$$|\psi\rangle = \frac{1}{2^{N/2}} \sum_{x=0}^{2^N-1} a_x |x\rangle$$

## Simulating Physics with Computers

Richard P. Feynman

Department of Physics, California Institute of Technology, Pasadena, California 91107

Received May 7, 1981

### 1. INTRODUCTION

On the program it says this is a keynote speech—and I don't know what a keynote speech is. I do not intend in any way to suggest what should be in this meeting as a keynote of the subjects or anything like that. I have my own things to say and to talk about and there's no implication that anybody needs to talk about the same thing or anything like it. So what I want to talk about is what Mike Dertouzos suggested that nobody would talk about. I want to talk about the problem of simulating physics with computers and I mean that in a specific way which I am going to explain. The reason for doing this is something that I learned about from Ed Fredkin, and my entire interest in the subject has been inspired by him. It has to do with learning something about the possibilities of computers, and also something about possibilities in physics. If we suppose that we know all the physical laws perfectly, of course we don't have to pay any attention to computers. It's interesting anyway to entertain oneself with the idea that we've got something to learn about physical laws; and if I take a relaxed view here (after all I'm here and not at home) I'll admit that we don't understand everything.

The first question is, What kind of computer are we going to use to simulate physics? Computer theory has been developed to a point where it realizes that it doesn't make any difference; when you get to a *universal computer*, it doesn't matter how it's manufactured, how it's actually made. Therefore my question is, Can physics be simulated by a universal computer? I would like to have the elements of this computer *locally interconnected*, and therefore sort of think about cellular automata as an example (but I don't want to force it). But I do want something involved with the

# Introduction and Overview (Preskills Notes)

## Quantum Complexity

Benioff, Feynman

Simulating Physics with computers (1982)



Simulating QM is hard

qubit  $|1\rangle$   
 $|0\rangle$   $|\psi\rangle = a|0\rangle + b|1\rangle$

$N$  qubits  $\rightarrow |01101011\dots 010\rangle = |x\rangle \in \mathcal{E}$

$N$  bit integer =  $x$ : logical basis state

$2^N$  - dimensional Hilbert Space

General State:

$$|\psi\rangle = \frac{1}{2^{N/2}} \sum_{x=0}^{2^N-1} a_x |x\rangle$$

## Simulating Physics with Computers

Richard P. Feynman

Department of Physics, California Institute of Technology, Pasadena, California 91107

Received May 7, 1981

### 1. INTRODUCTION

On the program it says this is a keynote speech—and I don't know what a keynote speech is. I do not intend in any way to suggest what should be in this meeting as a keynote of the subjects or anything like that. I have my own ideas about what should be in this meeting, but I don't want to talk about them. The speaker, Fredkin, has to also say something about the physics of computation. We've given a view here of what we understand about computation.

**1<sup>st</sup> Reading Assignment**

Read as much as you feel like, but don't get bogged down in the details.

Focus on Chapters 4 & 5 and summarize in a few bullet points, no more than a half page, 12pt font, single line spacing, 1 inch margins

The first question is, What kind of computer are we going to use to simulate physics? Computer theory has been developed to a point where it realizes that it doesn't make any difference; when you get to a *universal computer*, it doesn't matter how it's manufactured, how it's actually made. Therefore my question is, Can physics be simulated by a universal computer? I would like to have the elements of this computer *locally interconnected*, and therefore sort of think about cellular automata as an example (but I don't want to force it). But I do want something involved with the



# Introduction and Overview (Preskills Notes)

## Quantum Complexity

Benioff, Feynman  $\rightarrow$  Simulating QM is hard

Simulating Physics with computers (1982)

qubit  $|1\rangle$   
qubit  $|0\rangle$   $|z\rangle = a|0\rangle + b|1\rangle$

$N$  qubits  $\rightarrow |10101011\dots 010\rangle = |x\rangle \in \mathcal{E}$

$N$  bit integer =  $x$ : logical basis state

$2^N$  - dimensional Hilbert Space

## Quantum Computation

Computation is a map

$$|\Phi_{in}\rangle \rightarrow U|\psi_{in}\rangle = |\Phi_{out}\rangle$$

input

"program"

output

access via measurement

Schrödinger Evolution

$$i\hbar \frac{dU}{dt} = HU$$

given enough memory & time we can find  $U$

# Introduction and Overview (Preskills Notes)

## Quantum Computation

Computation is a map

$$|\Phi_{in}\rangle \xrightarrow{U} |\Phi_{out}\rangle = |\Phi_{out}\rangle$$

↑                    ↑                    ↑  
input            "program"            output  
access via  
measurement

Schrödinger Evolution

$$i\hbar \frac{dU}{dt} = HU$$

given enough memory  
& time we can find  $U$

## Quantum Computation

- \* A classical computer can simulate a QC
- \* Notion of computability unchanged

Simulation is hard :

$N$  qubits     $2^N$  prob. amp.'s

$N = 100 \rightarrow 2^{100} \sim 10^{30}$  p.a.'s

$N = 300 \rightarrow 2^{300} \sim 10^{90}$  p.a.'s

$10^{90} \gg$  # of particles in the  
visible universe

Jeff Kimble:    Hilbert Space is a  
mighty big place

# Introduction and Overview (Preskills Notes)

## Quantum Computation

- \* A classical computer can simulate a QC
- \* Notion of computability unchanged

Simulation is hard :

$N$  qubits  $2^N$  prob. amp.'s

$N = 100 \rightarrow 2^{100} \sim 10^{30}$  p.a.'s

$N = 300 \rightarrow 2^{300} \sim 10^{90}$  p.a.'s

$10^{90} \gg$  # of particles in the visible universe

Jeff Kimble: Hilbert Space is a mighty big place

- \* Is it possible for a classical computer to efficiently simulate QM ?
- \* Use probabilistic local algorithm (the most general kind)

John Bell

Bell's Theorem:

No local probabilistic theory can reproduce all of QM

# Introduction and Overview (Preskills Notes)

- \* Is it possible for a classical computer to efficiently simulate QM ?
- \* Use probabilistic local algorithm (the most general kind)

**John Bell**

**Bell's Theorem:**

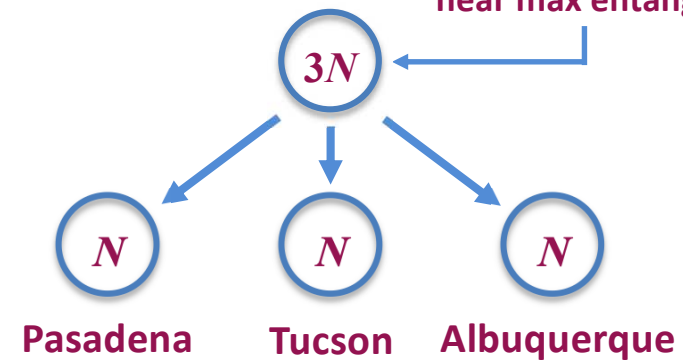
**No local probabilistic theory can reproduce all of QM**

## Non-Local Correlations

Key to Quantum Information

Thought experiment:

Generic Pure State in  $2^{3N}$  dimensional space near max entangled



Local state close to random  $\rho \sim \frac{1}{2^N} \mathbb{I}$

Shannon Info  $S = -\sum_{i=1}^{2^N} p_i \log p_i$

Von Neuman Info  $S = -\text{Tr}[\rho \log \rho]$

entropy  $\uparrow$

max value of  $S$   
 $\sim N - 2^{-(N+1)}$

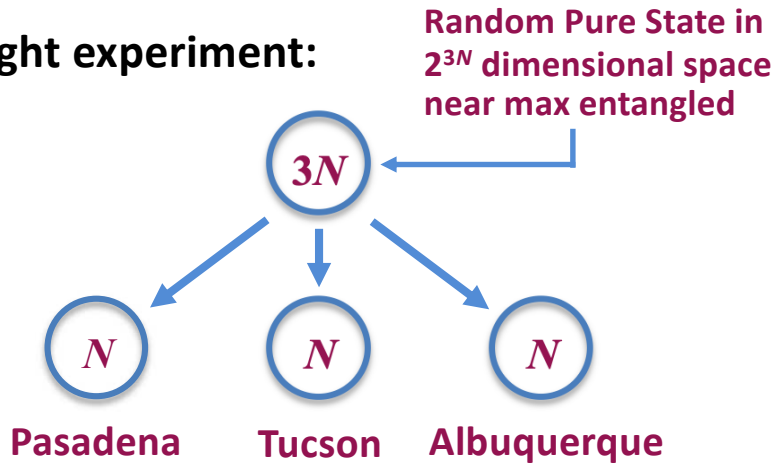
**Almost all info in  $N$  - body state in Non-Local correlations**

# Introduction and Overview (Preskills Notes)

## Non-Local Correlations

Key to Quantum Information

Thought experiment:



Local state close to random

$$S \sim \frac{1}{2^N} \mathbb{I}$$

Shannon Info

$$S = -\sum_{i=1}^{2^N} p_i \log p_i$$

max value of  $S$   
 $\sim N - 2^{-(N+1)}$

Von Neuman Info

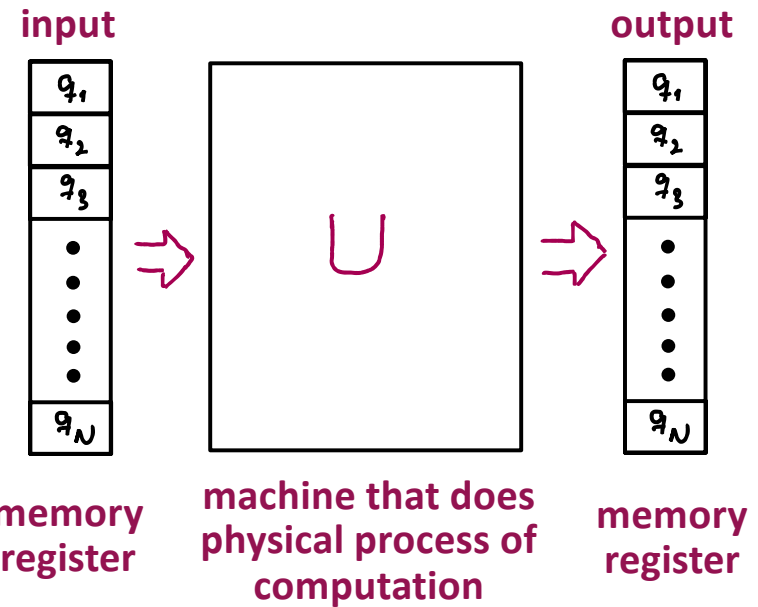
$$S = -\text{Tr}[\rho \log \rho]$$

entropy

Almost all info in  $N$  - body state in Non-Local correlations

OK – Plausible QM can do more  
 Where does the QC's power come from?

## Visualization of Computation



Classical: Register is in one of the logical states

$$x = q_1 q_2 q_3 \dots q_N$$

binary #

Reversible transformation

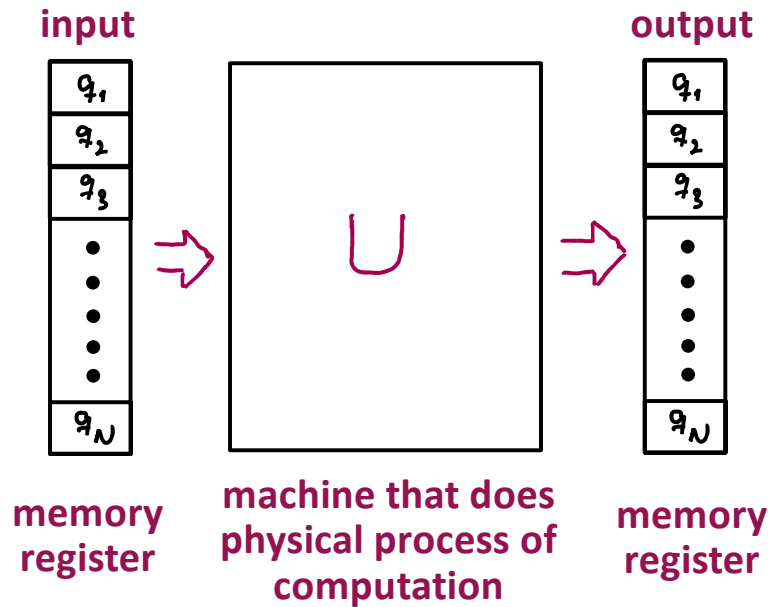
$$U: x \rightarrow y$$

# Introduction and Overview (Preskills Notes)

## OK – Plausible QM can do more

Where does the QC's power come from?

### Visualization of Computation



Classical: Register is in one of the logical states

$$x = \underbrace{q_1 q_2 q_3 \dots q_N}_{\text{binary \#}}$$

Reversible transformation

$$U: x \rightarrow y$$

Quantum: Register can be in any coherent superposition of logical states  $|x\rangle$

Unitary transformation  $U: |x\rangle \rightarrow |y\rangle$

Maps basis to basis  $U: \{|x\rangle\} \rightarrow \{|y\rangle\}$

### Quantum Parallelism

$$|\psi_{in}\rangle \rightarrow \sum_x a_x |x\rangle \rightarrow |\psi_{out}\rangle = U|\psi_{in}\rangle = \sum_x a_x |y\rangle = \sum_x b_x |x\rangle$$

Machine processes  $2^N$  inputs “in parallel” !

Beware: measurement collapses Q. Register into a single basis state at random

We get one random result out of  $2^N$

# Introduction and Overview (Preskills Notes)

Quantum: Register can be in any coherent superposition of logical states  $|x\rangle$

Unitary transformation  $U: |x\rangle \rightarrow |y\rangle$

Maps basis to basis  $U: \{|x\rangle\} \rightarrow \{|y\rangle\}$

## Quantum Parallelism

$$|\psi_{in}\rangle \rightarrow \sum_x a_x |x\rangle \rightarrow$$

Quantum Sampling Problem

$$\rightarrow |\psi_{out}\rangle = U|\psi_{in}\rangle = \sum_x a_x |y\rangle = \sum_x b_x |x\rangle$$

Machine processes  $2^N$  inputs “in parallel” !

Beware: measurement collapses Q. Register into a single basis state at random

We get one random result out of  $2^N$

Quantum Algorithms look for global properties of functions – symmetry, periodicity, etc.

- \* Classical -> requires many function evaluations
- \* Quantum -> design **U** so measurement gives answer with high probability
- \*  $\exists$  classes of problems (**sampling problems**) which are classically hard but quantum “easy”

↑  
Google “Quantum Supremacy”

Expert insight into current research

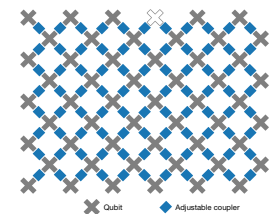
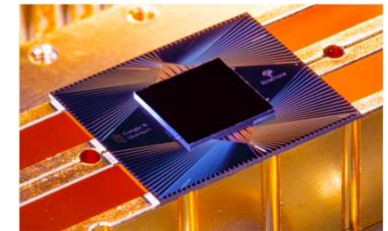
## News & views

Quantum information

### Quantum computing takes flight

William D. Oliver

A programmable quantum computer has been reported to outperform the most powerful conventional computers in a specific task – a milestone in computing comparable in importance to the Wright brothers’ first flights. See p.505



# Introduction and Overview (Preskills Notes)

Quantum Algorithms look for global properties of functions – symmetry, periodicity, etc.

- \* Classical  $\rightarrow$  requires many function evaluations
  - \* Quantum  $\rightarrow$  design  $U$  so measurement gives answer with high probability
  - \*  $\exists$  classes of problems (sampling problems) which are classically hard but quantum “easy”
- ↑
- Google “Quantum Supremacy”

Expert insight into current research

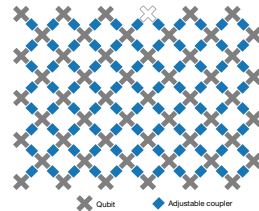
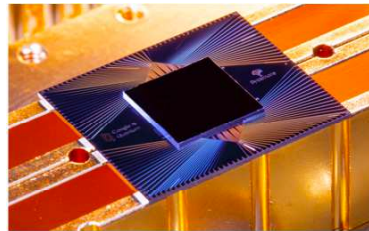
## News & views

Quantum information

### Quantum computing takes flight

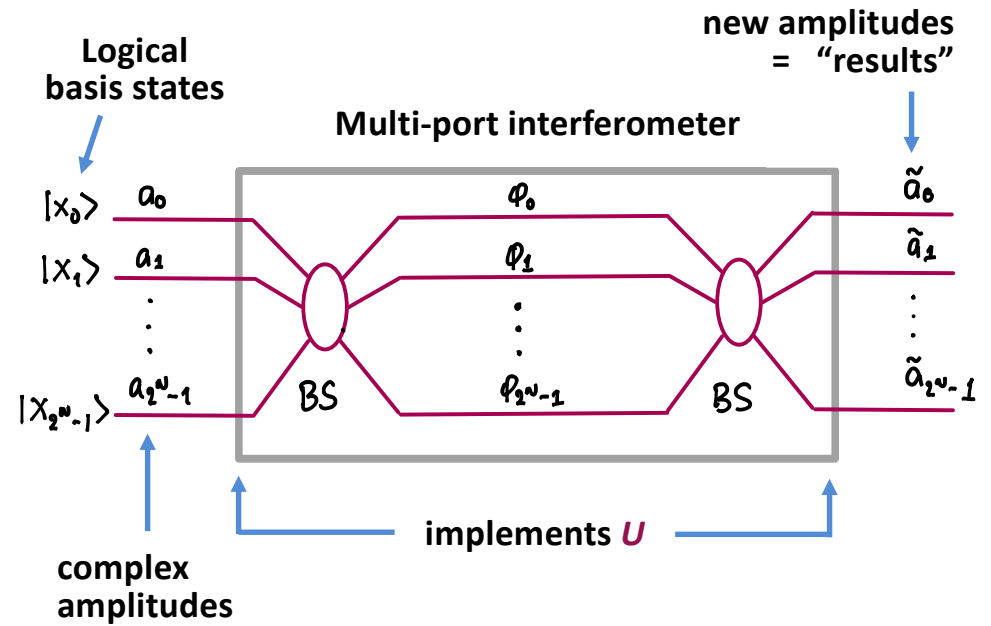
William D. Oliver

A programmable quantum computer has been reported to outperform the most powerful conventional computers in a specific task – a milestone in computing comparable in importance to the Wright brothers’ first flights. See p.505



What might be inside the machine ?

Wave interference w/classical fields ?



Note:  $N$  - qubit register  $\rightarrow 2^N$  “paths”

**Beware of Resource Scaling !**



# Introduction and Overview (Preskills Notes)

Quantum Algorithms look for global properties of functions – **symmetry, periodicity, etc.**

- \* Classical → requires many function evaluations
- \* Quantum → design **U** so measurement gives answer with high probability
- \*  $\exists$  classes of problems (**sampling problems**) which are classically hard but quantum “easy”  
Google “Quantum Supremacy”

Expert insight into current research

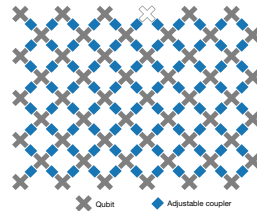
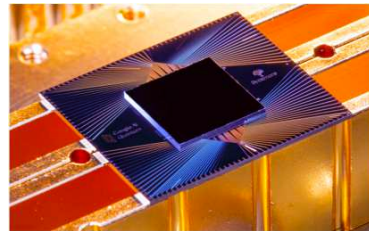
## News & views

Quantum information

### Quantum computing takes flight

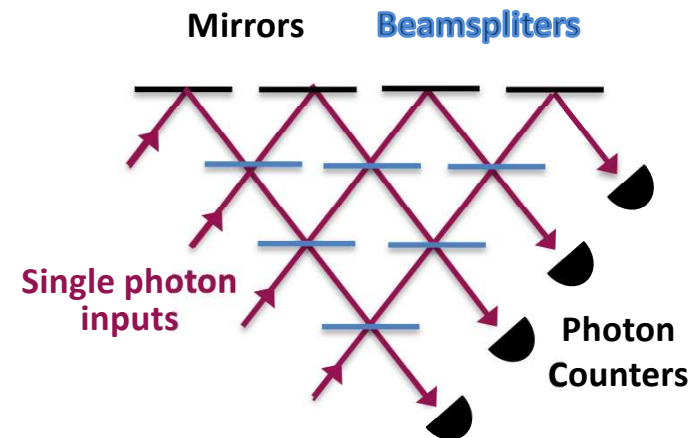
William D. Oliver

A programmable quantum computer has been reported to outperform the most powerful conventional computers in a specific task – a milestone in computing comparable in importance to the Wright brothers’ first flights. See p.505



## Boson Sampling

An example from Optics/Photonics Setup



Beware: Boson behavior at Beamsplitters hard to predict photon statistics across outputs.

Exponential in #'s of Beamsplitters

# Introduction and Overview (Preskills Notes)

Quantum Algorithms look for global properties of functions – **symmetry, periodicity, etc.**

\* Classical  $\rightarrow$  requires many function evaluations

\* Quantum  $\rightarrow$  design **U** so measurement gives answer with high probability

\*  $\exists$  classes of problems (**sampling problems**) which are classically hard but quantum “easy”

Google “Quantum Supremacy”

Expert insight into current research

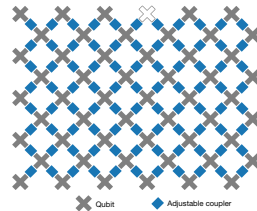
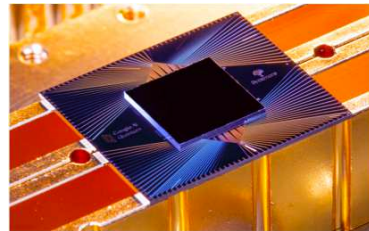
## News & views

Quantum information

## Quantum computing takes flight

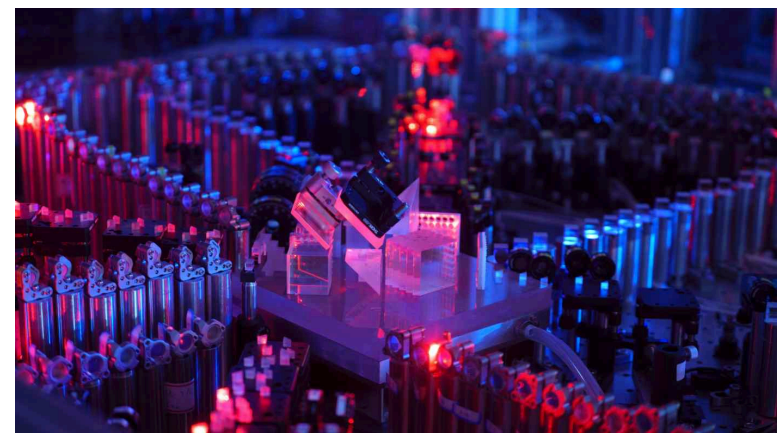
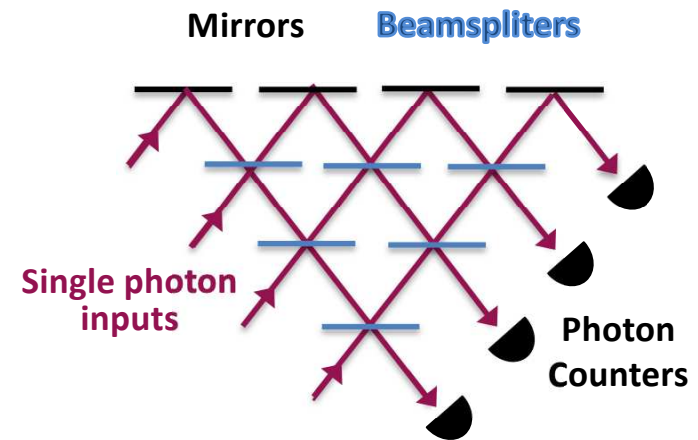
William D. Oliver

A programmable quantum computer has been reported to outperform the most powerful conventional computers in a specific task – a milestone in computing comparable in importance to the Wright brothers’ first flights. See p.505



## Boson Sampling

An example from Optics/Photonics Setup



An optical quantum computer developed by a team of Chinese researchers including those from the University of Science and Technology of China. (courtesy of Han-Sen Zhong of the research group)

# Introduction and Overview (Preskills Notes)

Quantum Algorithms look for global properties of functions – **symmetry, periodicity, etc.**

- \* Classical  $\rightarrow$  requires many function evaluations
- \* Quantum  $\rightarrow$  design **U** so measurement gives answer with high probability
- \*  $\exists$  classes of problems (**sampling problems**) which are classically hard but quantum “easy”  
Google “Quantum Supremacy”

Expert insight into current research

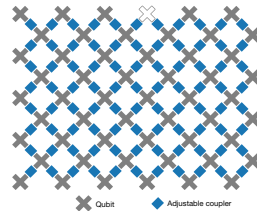
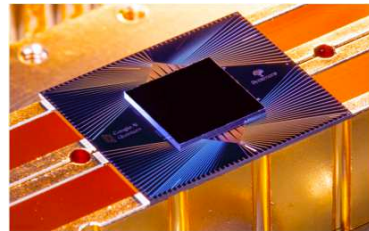
## News & views

Quantum information

## Quantum computing takes flight

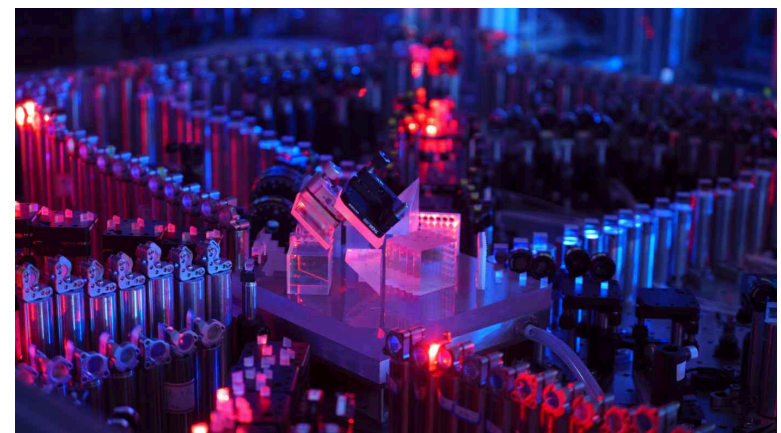
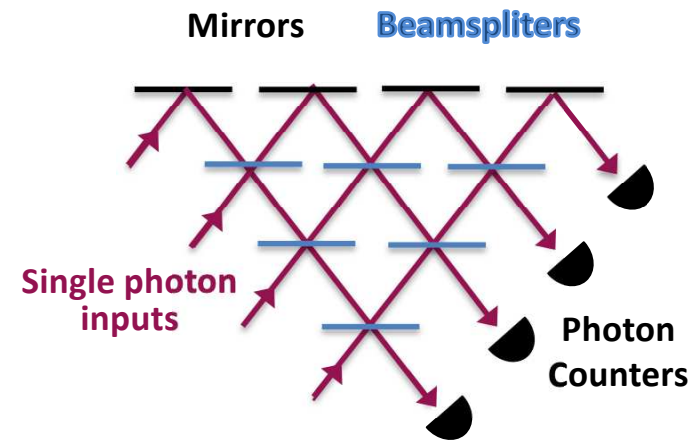
William D. Oliver

A programmable quantum computer has been reported to outperform the most powerful conventional computers in a specific task – a milestone in computing comparable in importance to the Wright brothers' first flights. See p.505



## Boson Sampling

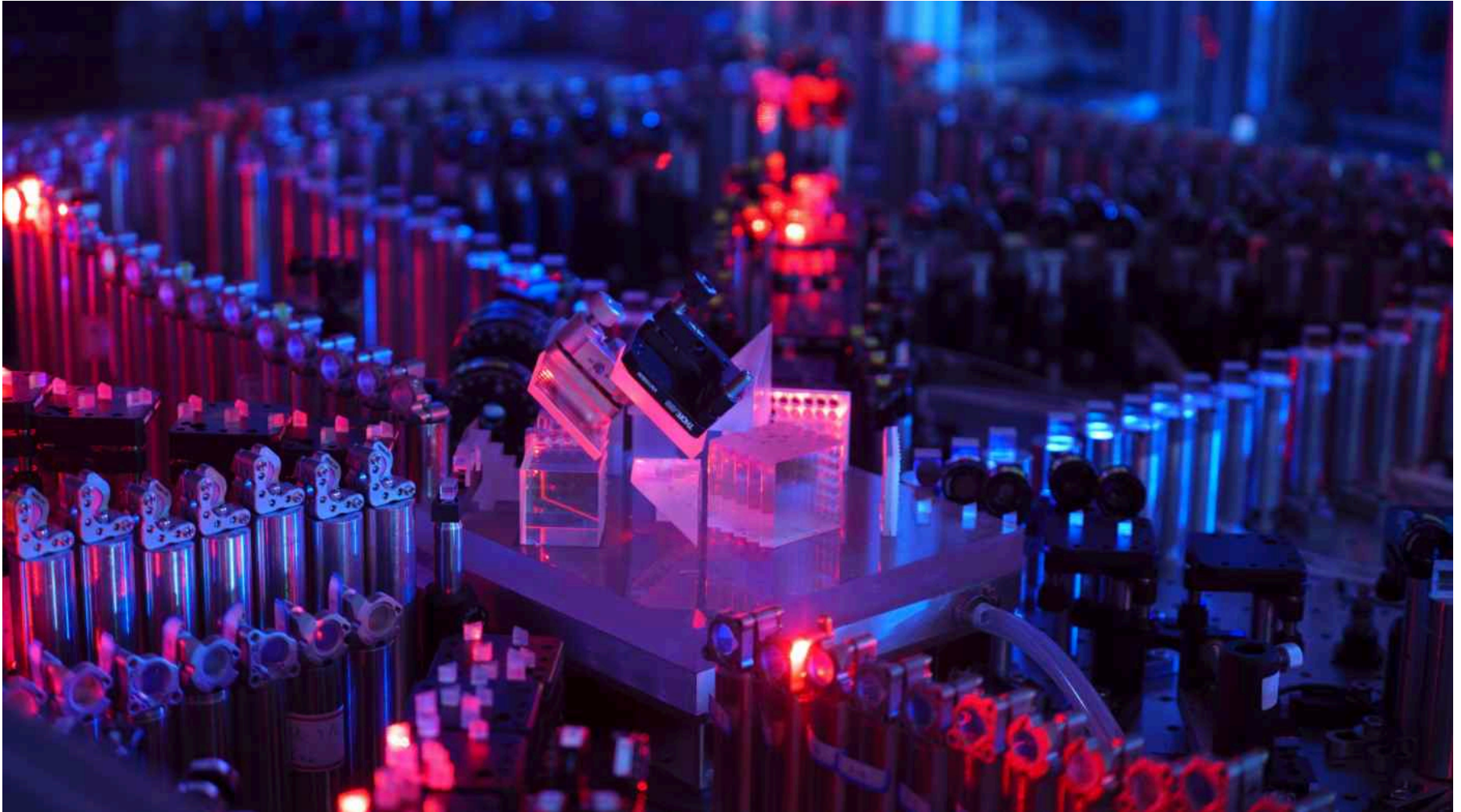
An example from Optics/Photonics Setup



An optical quantum computer developed by a team of Chinese researchers including those from the University of Science and Technology of China. (courtesy of Han-Sen Zhong of the research group)

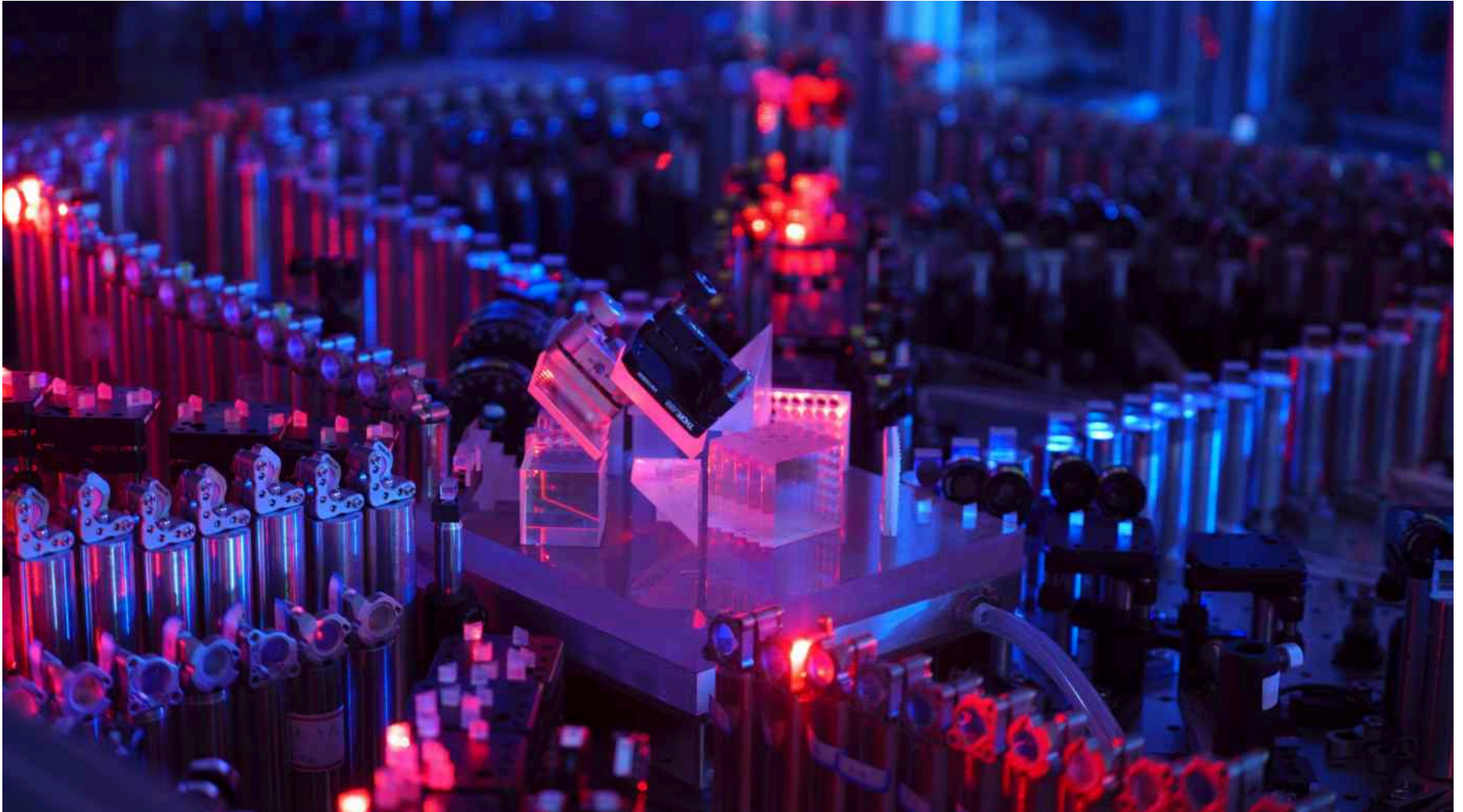


# Introduction and Overview (Preskills Notes)



An optical quantum computer developed by a team of Chinese researchers including those from the University of Science and Technology of China. (courtesy of Han-Sen Zhong of the research group)

# Introduction and Overview (Preskills Notes)



An optical quantum computer developed by a team of Chinese researchers including those from the University of Science and Technology of China. (courtesy of Han-Sen Zhong of the research group)



# Introduction and Overview (Preskills Notes)

Quantum Algorithms look for global properties of functions – **symmetry, periodicity, etc.**

\* Classical → requires many function evaluations

\* Quantum → design **U** so measurement gives answer with high probability

\* ∃ classes of problems (**sampling problems**) which are classically hard but quantum “easy”

Google “Quantum Supremacy”

Expert insight into current research

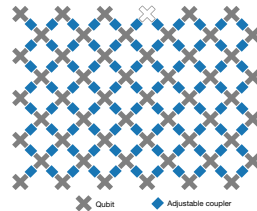
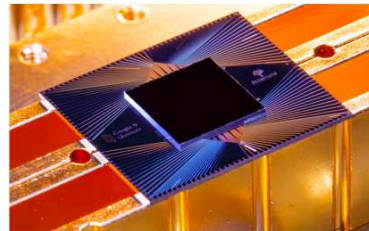
## News & views

Quantum information

## Quantum computing takes flight

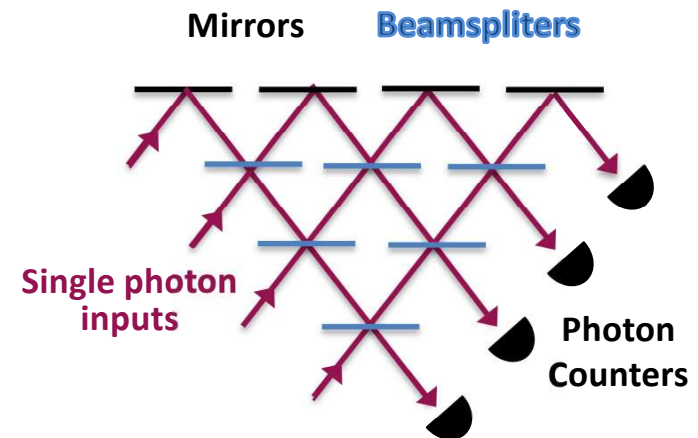
William D. Oliver

A programmable quantum computer has been reported to outperform the most powerful conventional computers in a specific task – a milestone in computing comparable in importance to the Wright brothers’ first flights. See p.505



## Boson Sampling

An example from Optics/Photonics Setup



Imagine aligning that thing...!