Irène Joliot-Curie
Laboratoire de Physique des 2 Infinis

Description of complex quantum systems with quantum computers

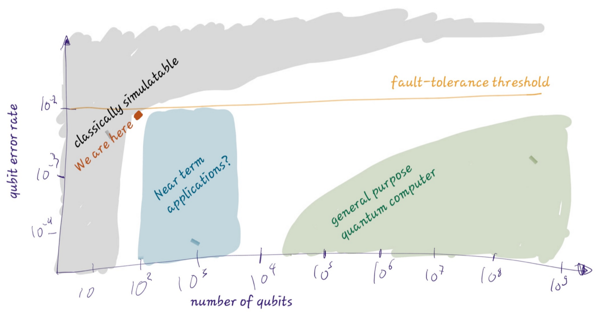
Denis Lacroix (IJCLab)

QC2I project

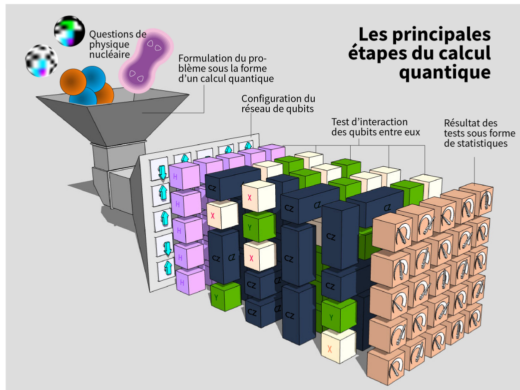
Depuis 80 ans, nos connaissances bâtissent de nouveaux mondes

Brief introduction to QC

Current status and opportunities

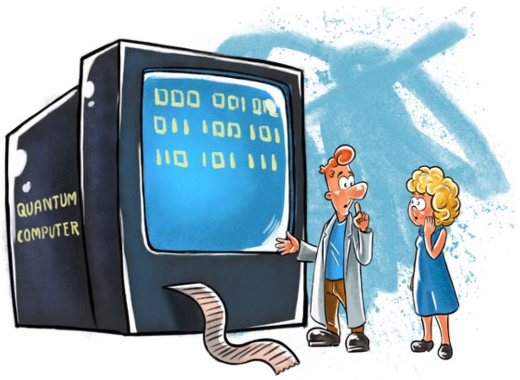


Discussion on ongoing projects in complex many-body systems



Short introduction to bit versus Qubits

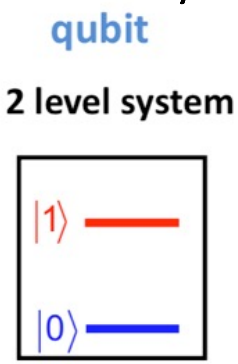
Classical computers
Works with bits

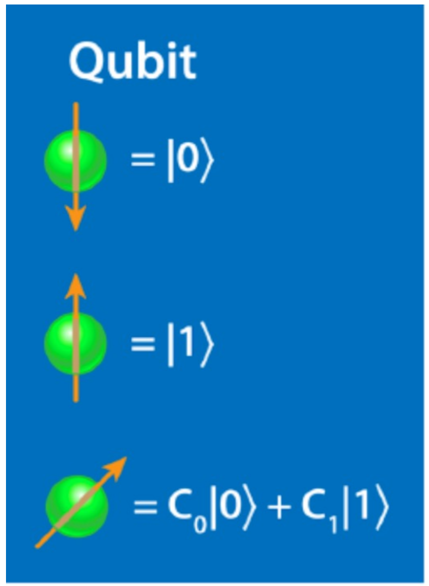


Bits are only 0 or 1

Quantum computers with
Quantum bits

Qubits can be seen
As two-level systems





A single Qubit can be any superposition of 0 and 1

Obvious advantage

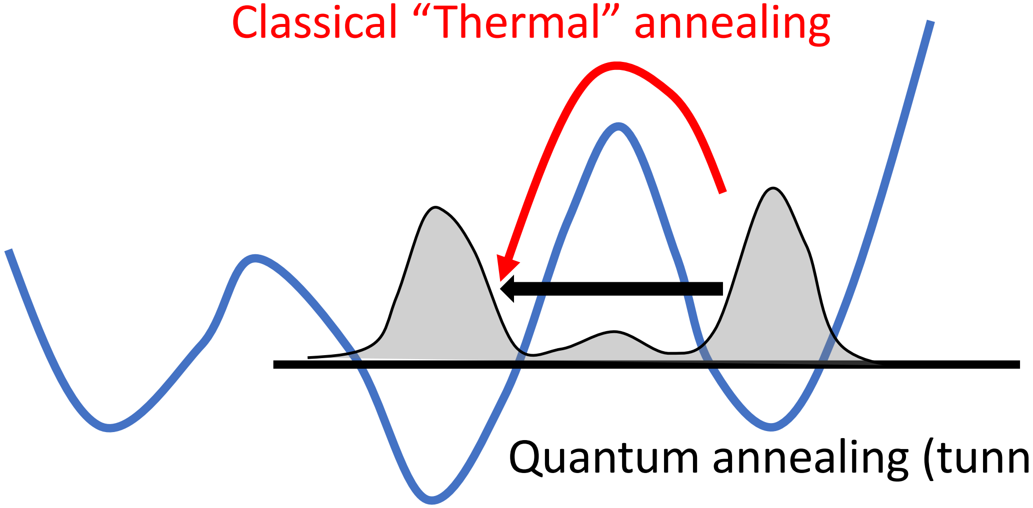
The coefficients can take any values that verifies $|c_0|^2 + |c_1|^2 = 1$

And with many
Qubits

New aspects can be used like quantum interference and entanglement

Illustration of quantum advantages

Quantum Tunneling and quantum annealing



Quantum entanglement

Assume two persons (Alice and Bob)  

The humor of A&B are encoded in the wave-function

A B → A B

$$|\Phi\rangle = \alpha|😀😀\rangle + \beta|😫😫\rangle$$

Suppose I measure Bob 😀 → 😀

😫 → 😫

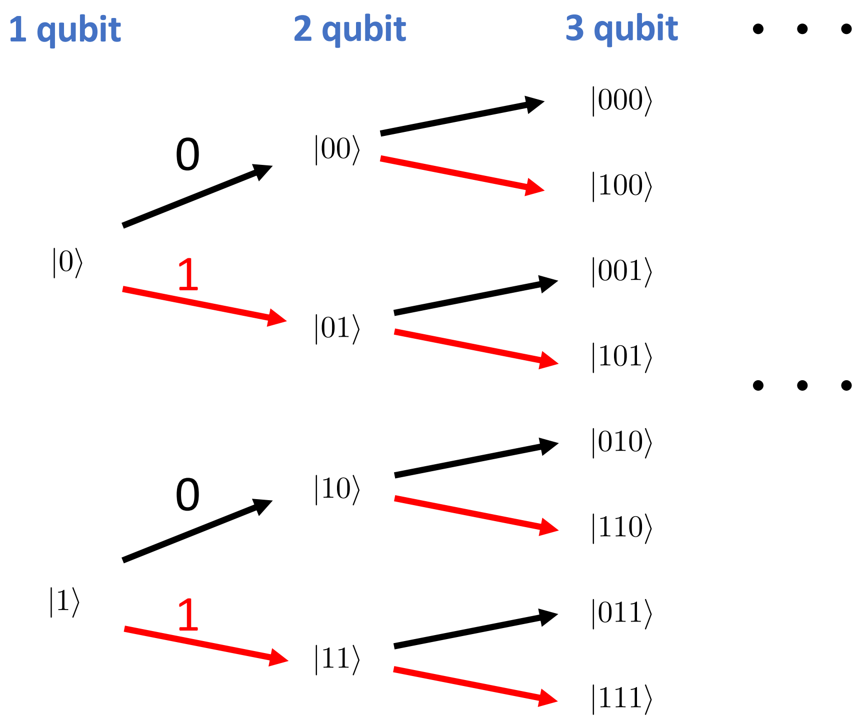
I can measure partial info and get the full info

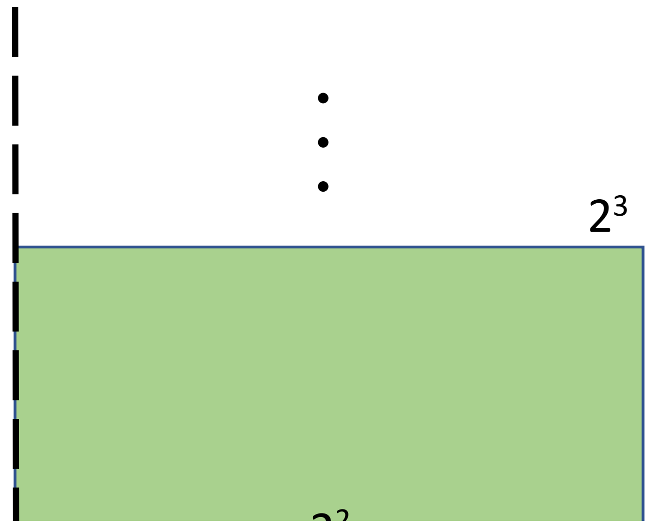
The info is destroyed after measurement

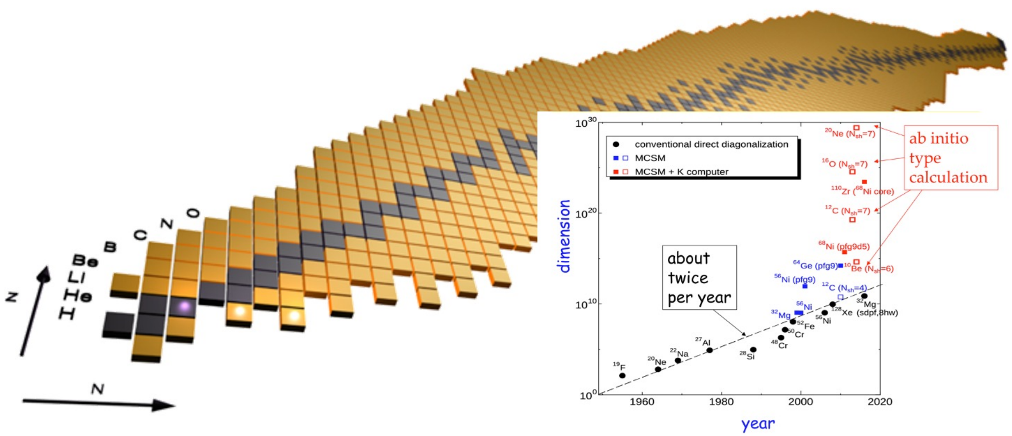
Illustration of quantum advantages

Systems described on qubits

$$|\Psi\rangle = \sum_{s_i=0,1} \Psi_{s_1,\cdots,s_N} |s_1,\cdots,s_n\rangle$$







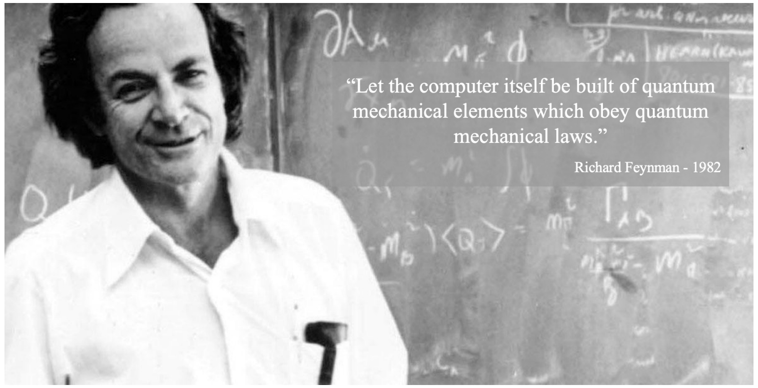
Quantum supremacy

$\sim 2^{50}$

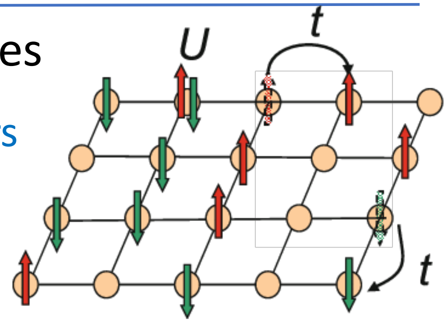
With 2^{300} (i.e. 300 qubits) the size is more than the number of particles In the universe. J. Preskill

What are the anticipated applications ?

Simulation of Quantum complex systems



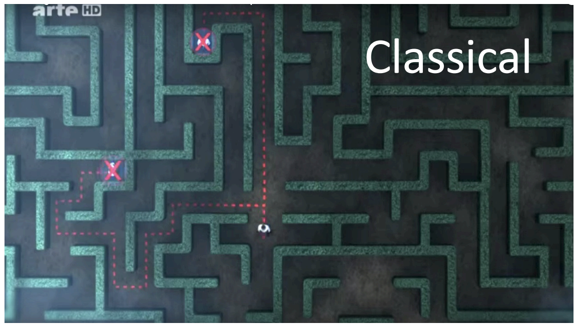
Ex: systems on lattices



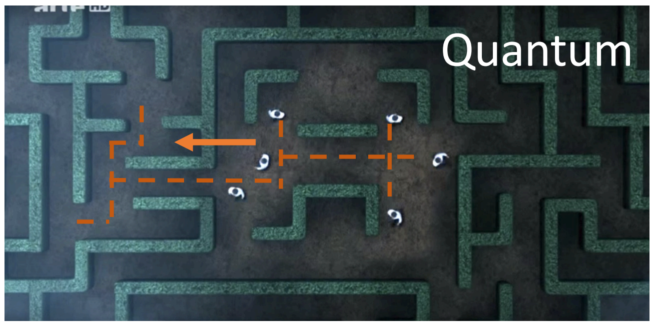
On classical computers
Can be solved exactly
For max 20 particles.

On quantum computers:
N sites means only N qubits

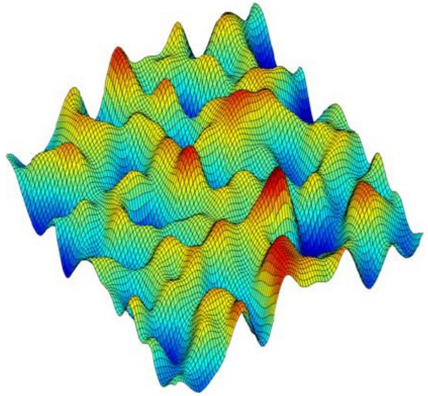
Quantum versus classical search



VS

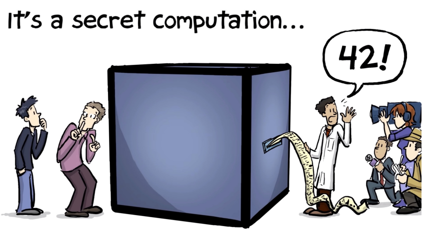


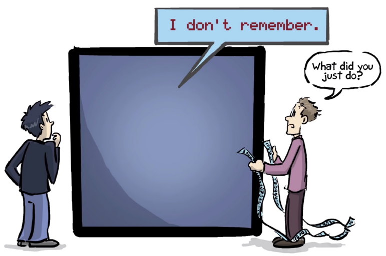
Credit: The Fabric of The Cosmos: Quantum Leap

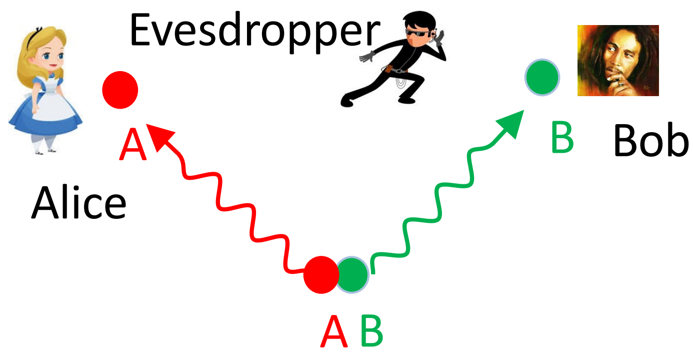


Exploring complex landscape:
molecules,
customers preferences (amazon), ...

Quantum secrets (cryptography, quantum key, ...)



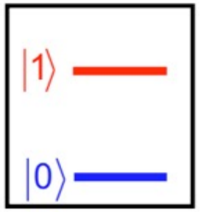




Minimal - Practical aspects of quantum computers

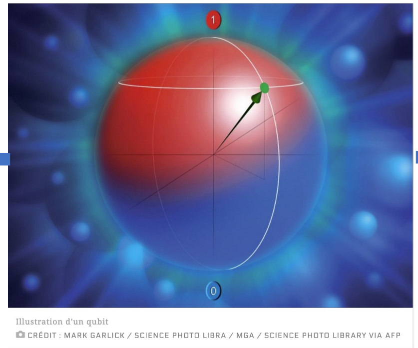
qubit

2 level system



Manipulate the Qubits (Make rotations)

$|0\rangle$ Initial state



Final state $c_0|0\rangle + c_1|1\rangle$

Measure the state

$|c_0|^2, \; |c_1|^2$

("destroy" the state)

With many Qubits

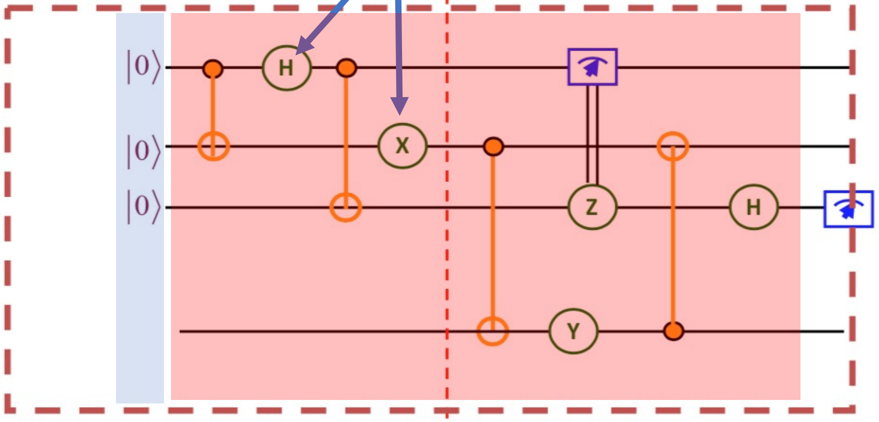
$N = 2^n$ computational basis states

$$\overbrace{|010001\ldots1\rangle}^{n} = |p\rangle$$

Initial state

Elementary operations

Quantum circuits

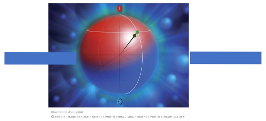


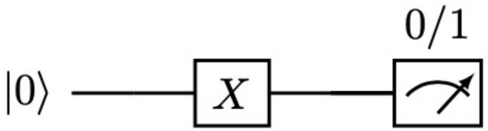
$$\sum_{i_k=0,1} a_{i_1 i_2 i_3 i_{4\ldots} i_{2^N}} |i_1, i_2, i_3 \ldots i_{2^N}\rangle$$

Gives the $|a|^2$

The quantum computing toolkit

Unary operations

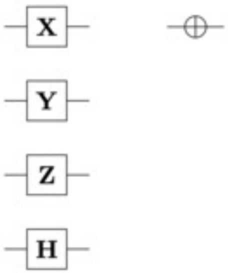




Rotations

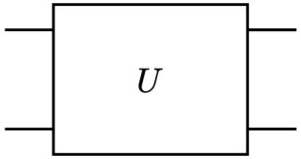
$$R_X(\varphi) = e^{-i\varphi X/2}$$

Standard examples

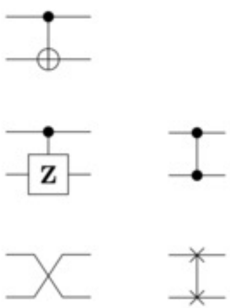


Pauli-X (X)		$\begin{matrix} & \|0\rangle & \|1\rangle & \\ & \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & & \begin{matrix}\|0\rangle \\ \|1\rangle\end{matrix} \end{matrix}$
Pauli-Y (Y)		$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z (Z)		$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Hadamard (H)		$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

Binary operations



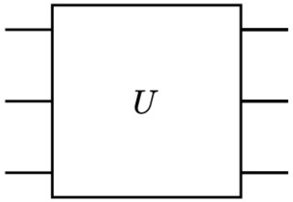
Standard examples



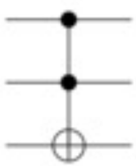
Controlled Not (CNOT, CX)		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$	$\|11\rangle \leftrightarrow \|10\rangle$
Controlled Z (CZ)		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$	$\|11\rangle \leftrightarrow -\|11\rangle$
SWAP		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix}\|00\rangle \\ \|01\rangle \\ \|10\rangle \\ \|11\rangle\end{matrix}$	$\|01\rangle \leftrightarrow \|10\rangle$

Ternary operations

Standard example



Toffoli (CCNOT, CCX, TOFF)



$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$|110\rangle \leftrightarrow |111\rangle$$

FROM DREAM TO REALITY

Quantum programming is "easy" but working really with quantum computers is difficult

-adapt to the technology.
-search of efficient algorithms on this technology.
-try to correct for nasty noise as much as possible.

This looks more like an experimental program than informatic or quantum theory.

QC is not unique

Digital QC

Qubit, Qutrit, qudits

Analog computing

Quantum computing today is firstly an experimental challenge

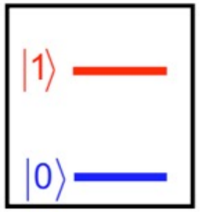
REALITY

Everything around want to destroy the ideal picture and the quantum coherence.

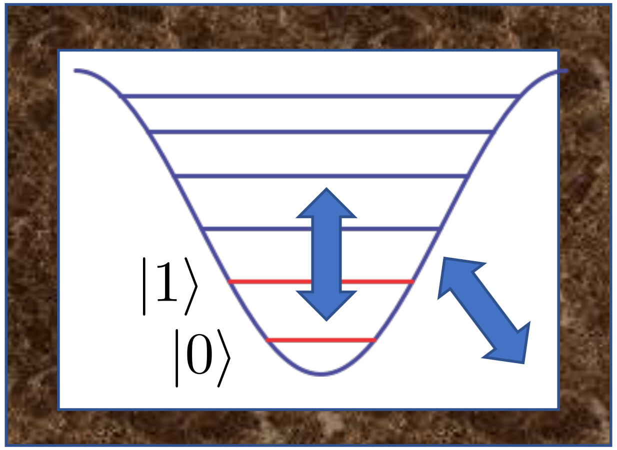
Leads to loss of information
And decoherence

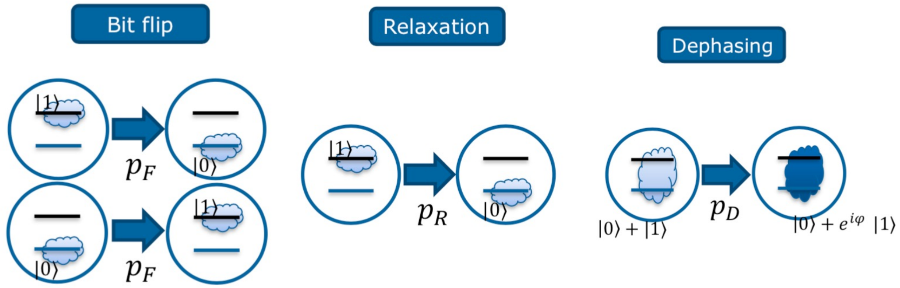
Ideal Qubits

2 level system



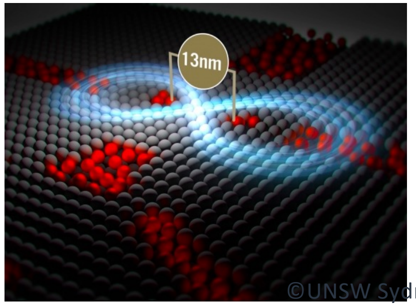
External Exp. Setup



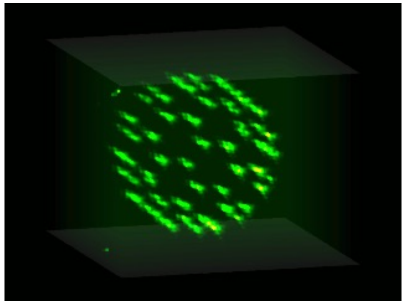


For multi-Qubits, also cross-talk (making operations on one qubit Impacts other qubits)

Working with quantum computers now means working in a noise environment short programs (before decoherence occurs)

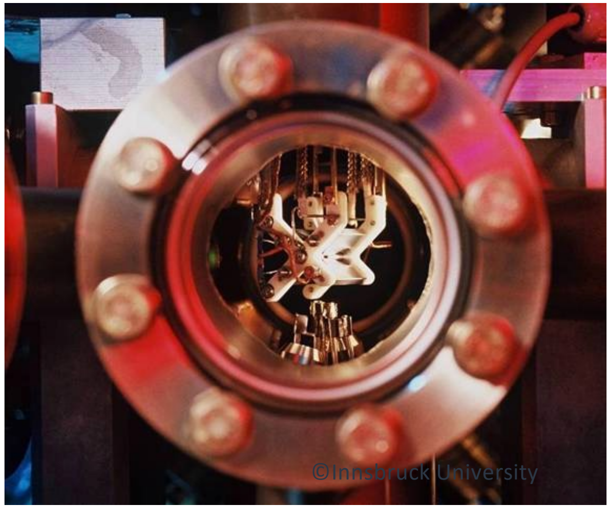


Silicon qubits

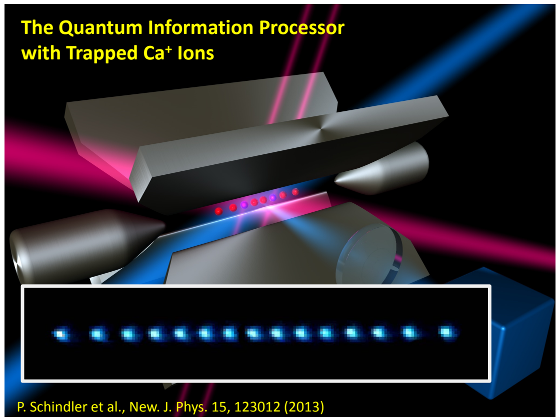


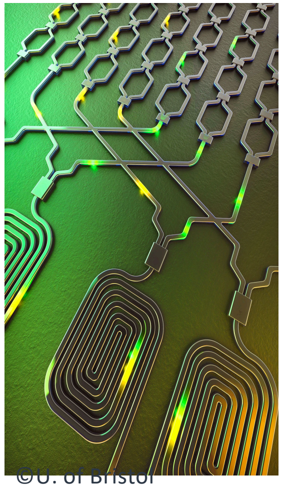
Neutral atoms

NMR

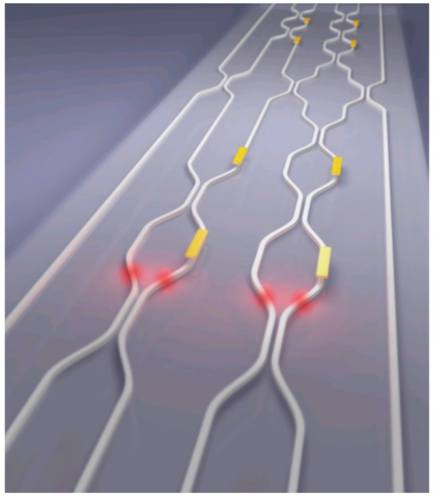


Trapped ions





Photons



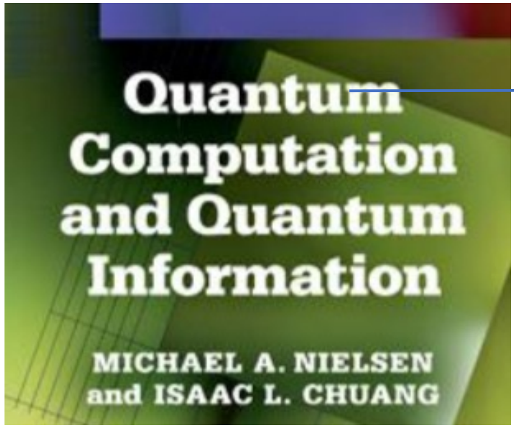
Superconducting qubits

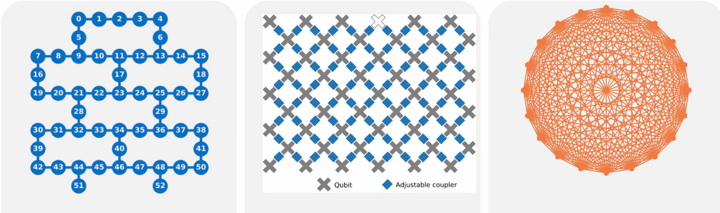
Platforms comparison

	Leading technologies in NISQ era[1]		Candidate technologies beyond NISQ		
Qubit type or technology	**Superconducting[2]**	**Trapped ion**	**Photonic**	**Silicon-based[3]**	**Topological[8]**
Description of qubit encoding	Two-level system of a superconducting circuit	Electron spin direction of ionized atoms in vacuum	Occupation of a waveguide pair of single photons	Nuclear or electron spin or charge of doped P atoms in Si	Majorana particles in a nanowire
Physical qubits[4,5]	IBM: 20, Rigetti: 19, Alibaba: 11, Google: 9	Lab environment: AQT[6]: 20, IonQ: 14	**6×3[9]**	**2**	target: 1 in 2018
Qubit lifetime	**~50–100 µs**	**~50 s**	**~150 µs**	**~1–10 s**	target ~100 s
Gate fidelity[7]	**~99.4%**	**~99.9%**	**~98%**	**~90%**	target ~99.9999%
Gate operation time	**~10–50 ns**	**~3-50 µs**	**~1 ns**	**~1–10 ns**	–
Connectivity	Nearest neighbors	All-to-all	To be demonstrated	Nearest neighbor	–
Scalability	No major road-blocks near-term	Scaling beyond one trap (>50 qb)	Single photon sources and detection	Novel technology potentially high scalability	?
Maturity or technology readiness level	TRL[10] 5	TRL 4	TRL 3	TRL 3	TRL 1
Key properties	Cryogenic operation Fast gating Silicon technology	Improves with cryogenic temperatures Long qubit lifetime Vacuum operation	Room temperature Fast gating Modular design	Cryogenic operation Fast gating Atomic-scale size	Estimated: Long lifetime High fidelities

Boston Consulting Group – Nov 2018

Where we are now



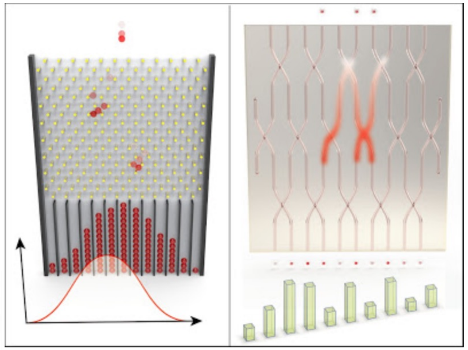


IBM
Rochester 53 qubits, october 2019
65 qubits, october 2020

Google
Sycamore 53 qubits, october 2019

IonQ
11 qubits, 2018
32 qubits, 2020

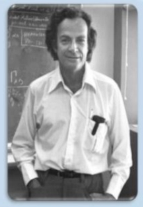
Quantum computational advantage using photons, Science 370 (2020)



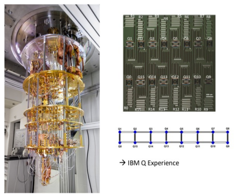
Simulating physics with computers-1982

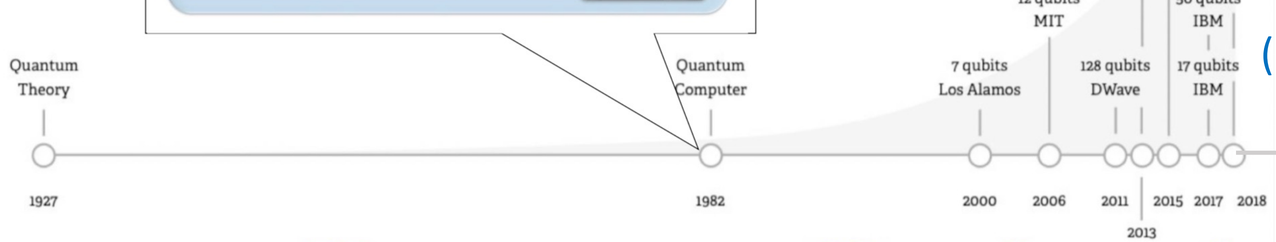
Richard P. Feynman (Nobel Prize in Physics 1965)

"Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy."



IBM QX5 (16 qubits)



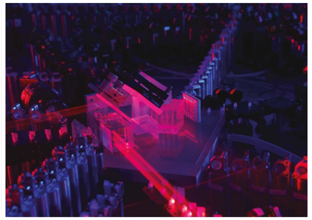


55 YEARS

18 YEARS

6 YEARS

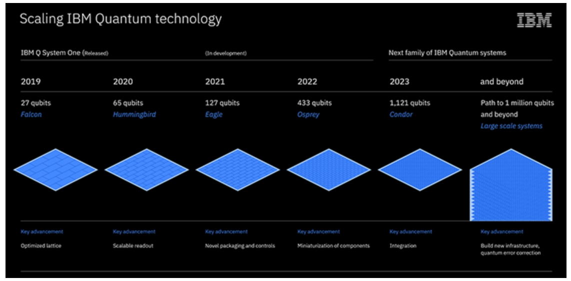
1 YEAR



This photonic computer did a task that would take a classical computer 2.5 billion years.

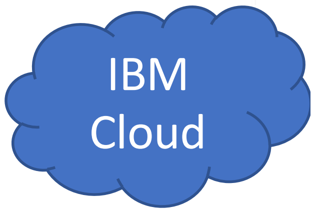
(2020)

(2021)



?

IonQ Gemini desk computer



Quantum supremacy using a programmable superconducting processor

Nature | Vol 574 | 24 OCTOBER 2019 | **505**

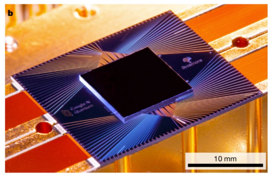
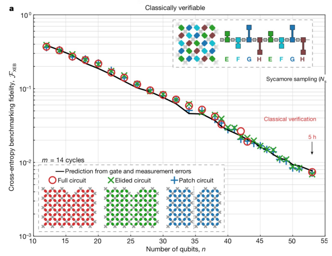
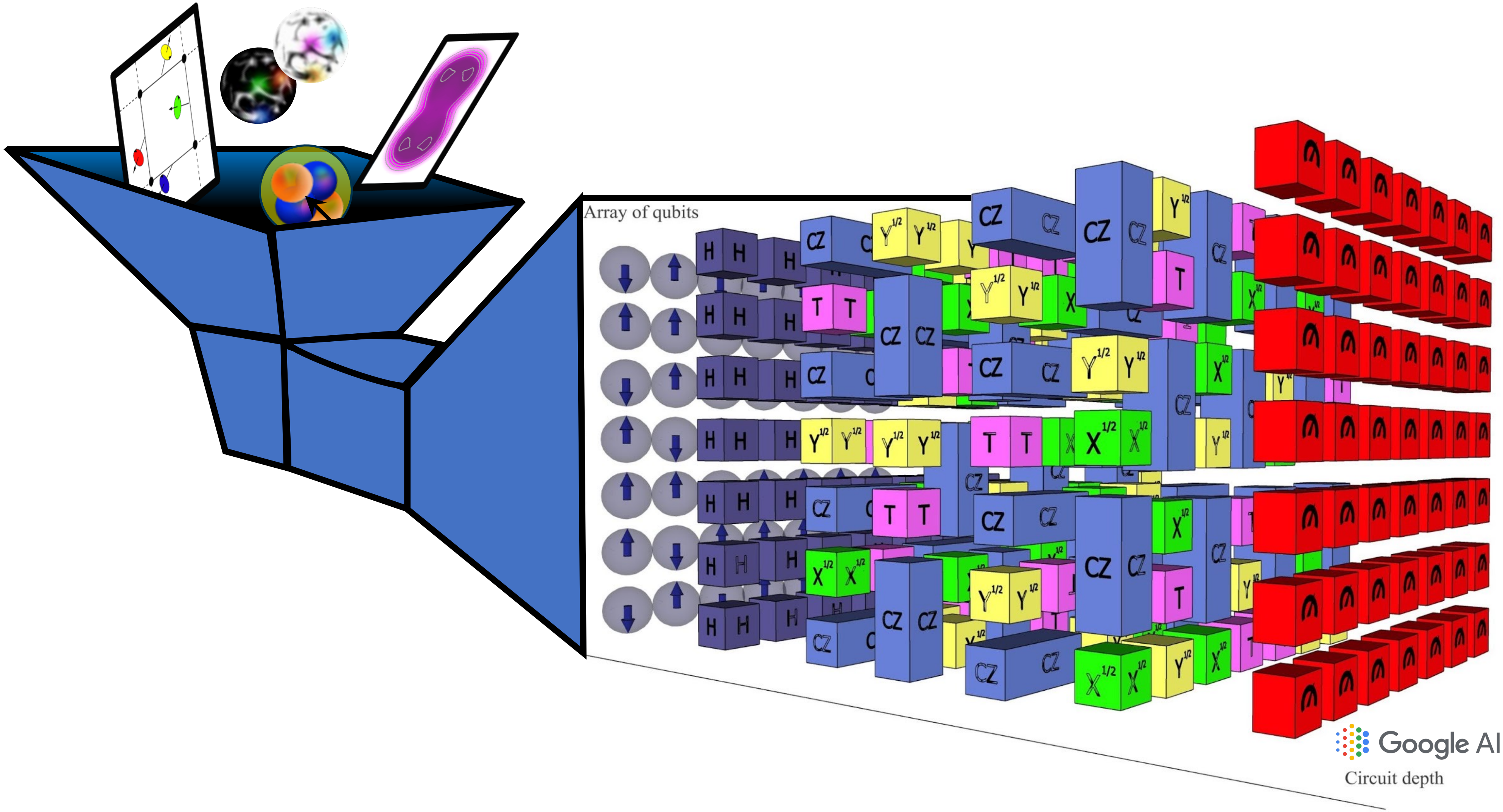


Fig. 1 | The Sycamore processor. a, Layout of processor, showing a rectangular array of 54 qubits (grey), each connected to its four nearest neighbours with couplers (blue). The inoperable qubit is outlined. b, Photograph of the Sycamore chip.

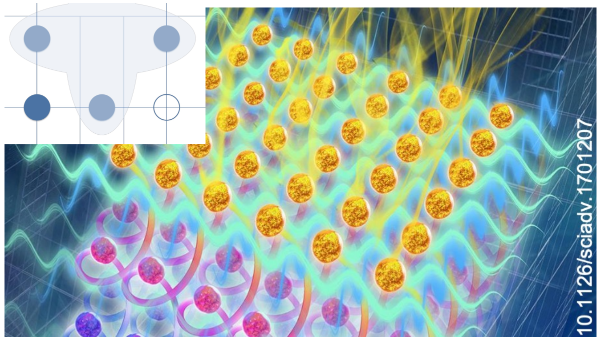


Coming back to the Nuclear physics case



Few initiated applications in the world in IN2P3 fields

Lattice gauge theories



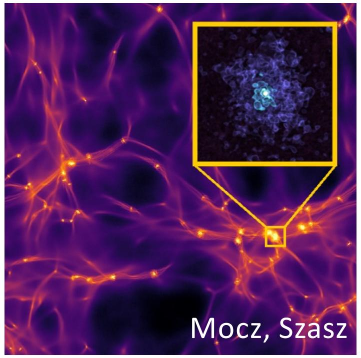
Zohar, Kolck, Savage, …

N-body problem

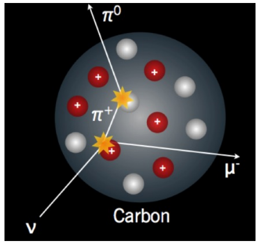
N-body nuclear systems

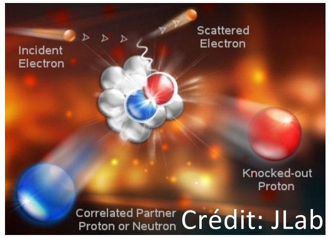
Dumitrescu, Hagen, Carlson, Papenbrock…

Dark matter



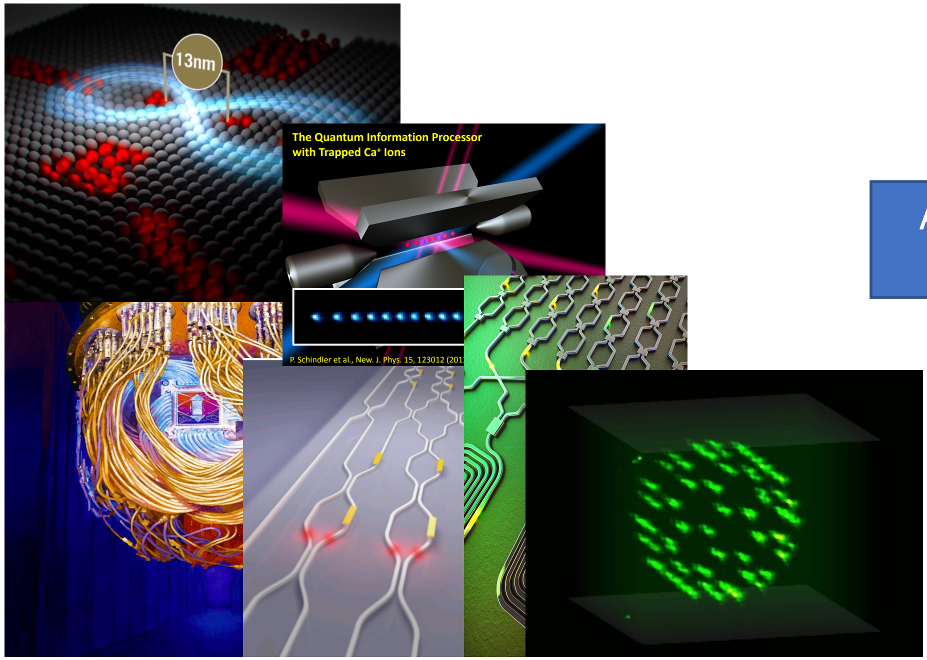
Dynamics: e, ν scattering



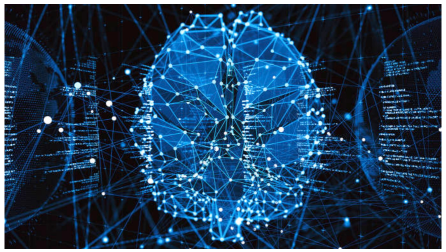


Roggero, Carlson, …

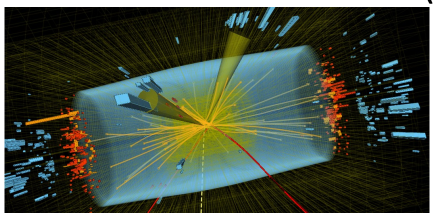
E. Zohar, J. I. Cirac, B. Reznik, Phys. Rev. Lett. **110**, 125304 (2013)
E. Zohar, J. I. Cirac, B. Reznik, Phys. Rev. A **88** 023617 (2013)
E. Zohar, J. I. Cirac, B. Reznik, Rep. Prog. Phys. **79**, 014401 (2016)
D. González Cuadra, **E. Zohar**, J. I. Cirac, New J. Phys. **19** 063038 (2017)



Applications to data mining (event classification)



CMS-detector (with LLR)



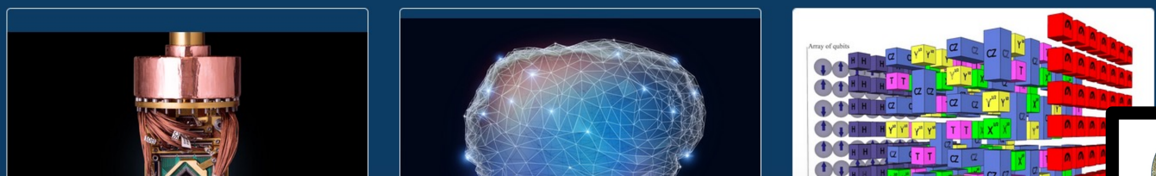
Some advertising

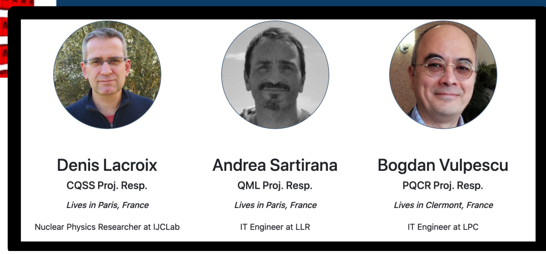
https://qc.pages.in2p3.fr/web/

QC2I is a computing project supported by IN2P3, the French national nuclear and particle physics institute. Its goal is to explore the possible applications of the emerging quantum computing technologies to particles and nuclear physics problems as well as astrophysics. The main tasks are:

- to identify, within IN2P3, scientists/engineers/technicians who are interested in using quantum technologies,
- to facilitate the access and training on quantum computers,
- to identify milestones applications for nuclear/particle physics and astrophysics,
- to design dedicated algorithms and proof of principle applications.

The project action has three main directions: **Prepare the Quantum Computing Revolution** (PQCR), **Quantum Machine Learning** (QML), **Complex Quantum Systems Simulation** (CQSS)



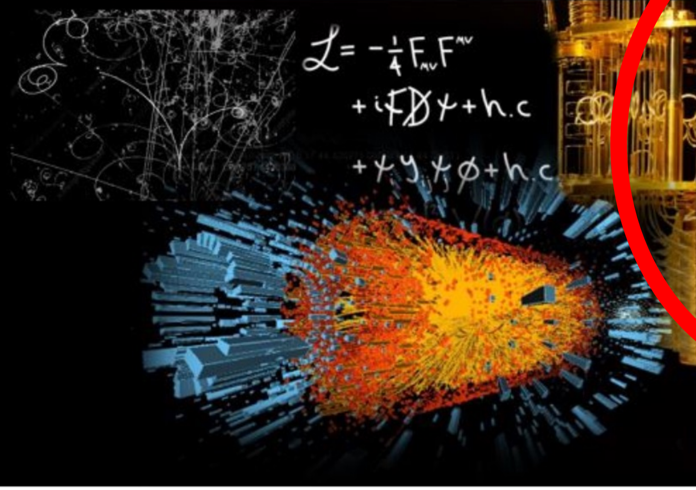


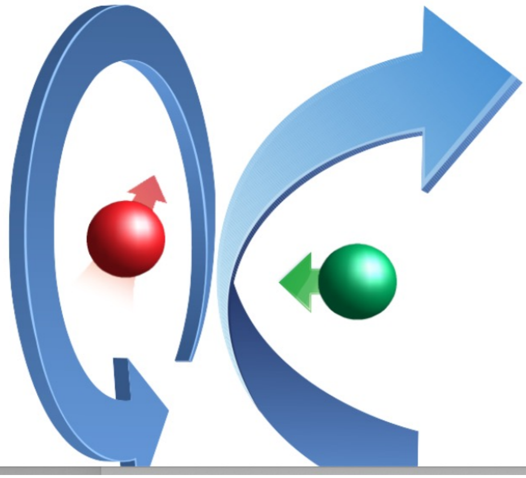
https://qc.pages.in2p3.fr/web/

QC2I | About | Members | Meetings | Talks | Collab-Tools | QC-Usefull

- QC-Events
- QC-Basics
- QC-Tools
- QC-Hardware
- QC-Web
- QC-Q&A

QC2I: Quantum Computing for the two Infinites





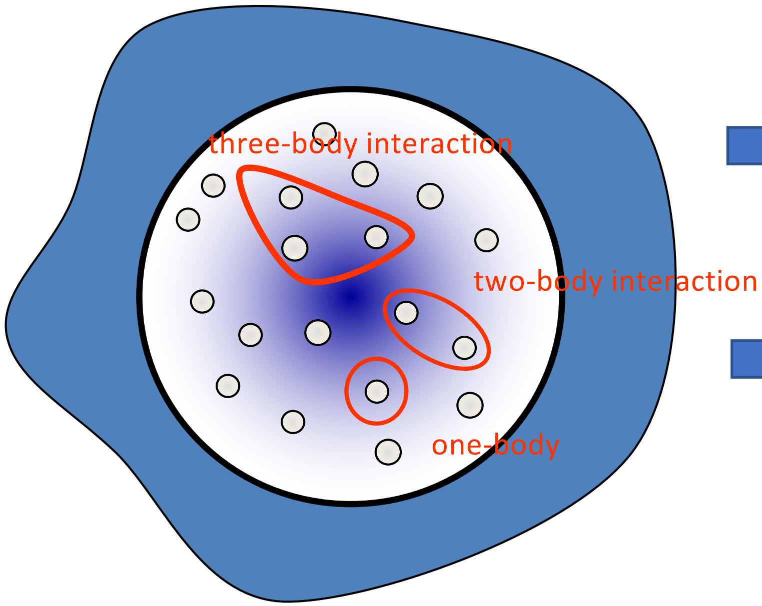
QC2I is a computing project supported by IN2P3, the French national nuclear and particle physics institute. Its goal is to explore the possible applications of the emerging quantum computing technologies to particles and nuclear physics problems as well as astrophysics. The main tasks are:

- to identify, within IN2P3, scientists/engineers/technicians who are interested in using quantum technologies,
- to facilitate the access and training on quantum computers,
- to identify milestones applications for nuclear/particle physics and astrophysics,
- to design dedicated algorithms and proof of principle applications.

The project action has three main directions: **Prepare the Quantum Computing Revolution** (PQCR), **Quantum Machine Learning** (QML), **Complex Quantum Systems Simulation**

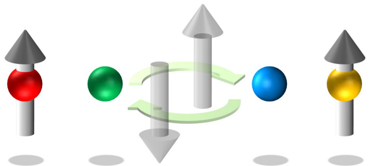
One example:

Simulation of complex quantum (interacting) systems

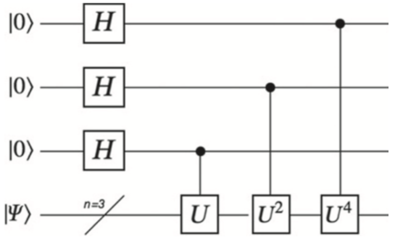


- If you have N one-body degrees of freedoms
  The Hilbert space has an exponential
  Scaling (~N!)
- Even today, only a limited area (small systems- few %) of the nuclear Chart can be calculated with most powerful Supercomputers.

Take a simple version of
your favorite many-body problem



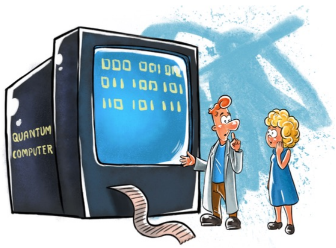
Map/formulate it as a problem
with Qubit



Use standard QC algorithms
or
Propose new QC algorithms

Test on a real
Quantum
platforms

It works sometimes !

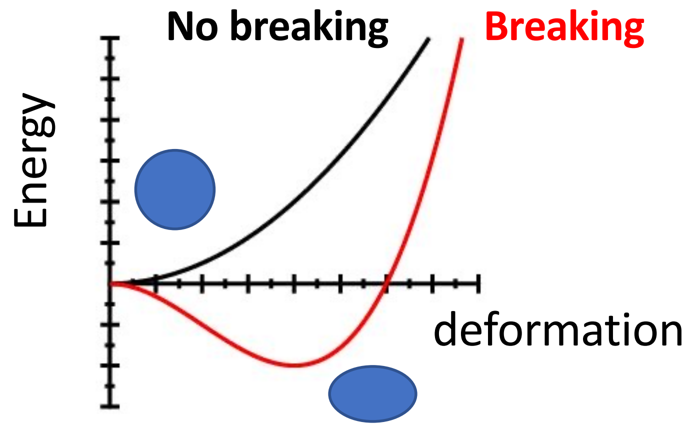


Test on a
QC emulator

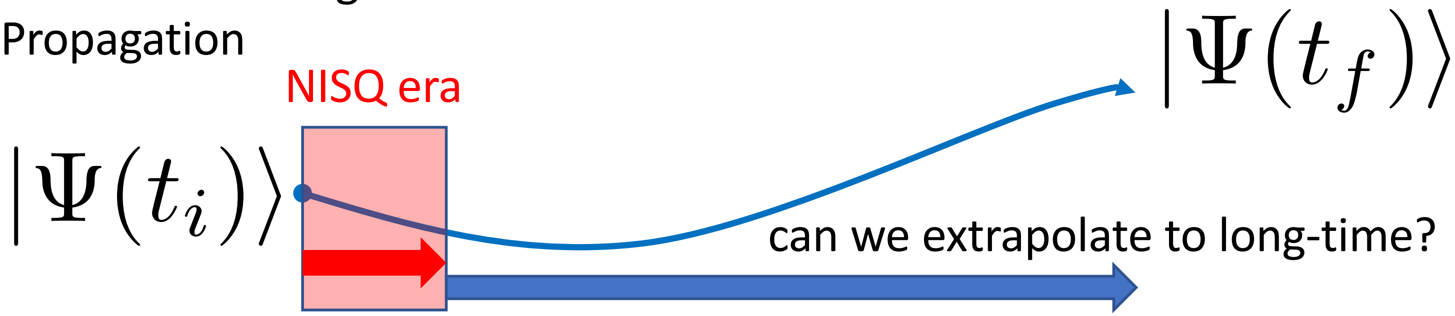
It works !

The recent applications we made (in many-body systems)

- Breaking symmetries and restoration of symmetries in many-body systems on quantum computers
  - Application to the counting of particle number (for superfluid systems)

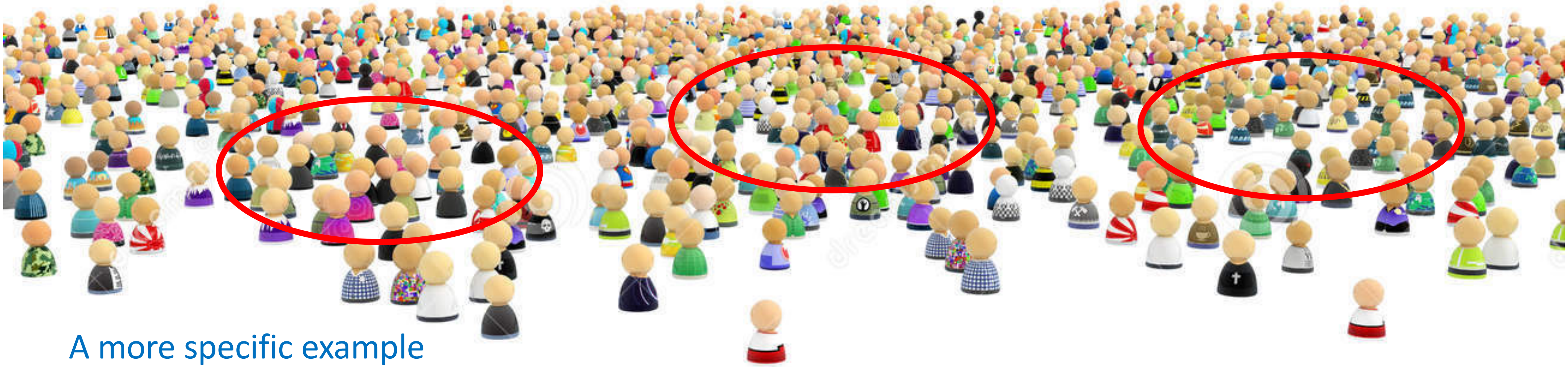


- Replacing bosons by pairs of fermions to probe quantum supremacy
  - Prediction of long time evolution from short-time Propagation

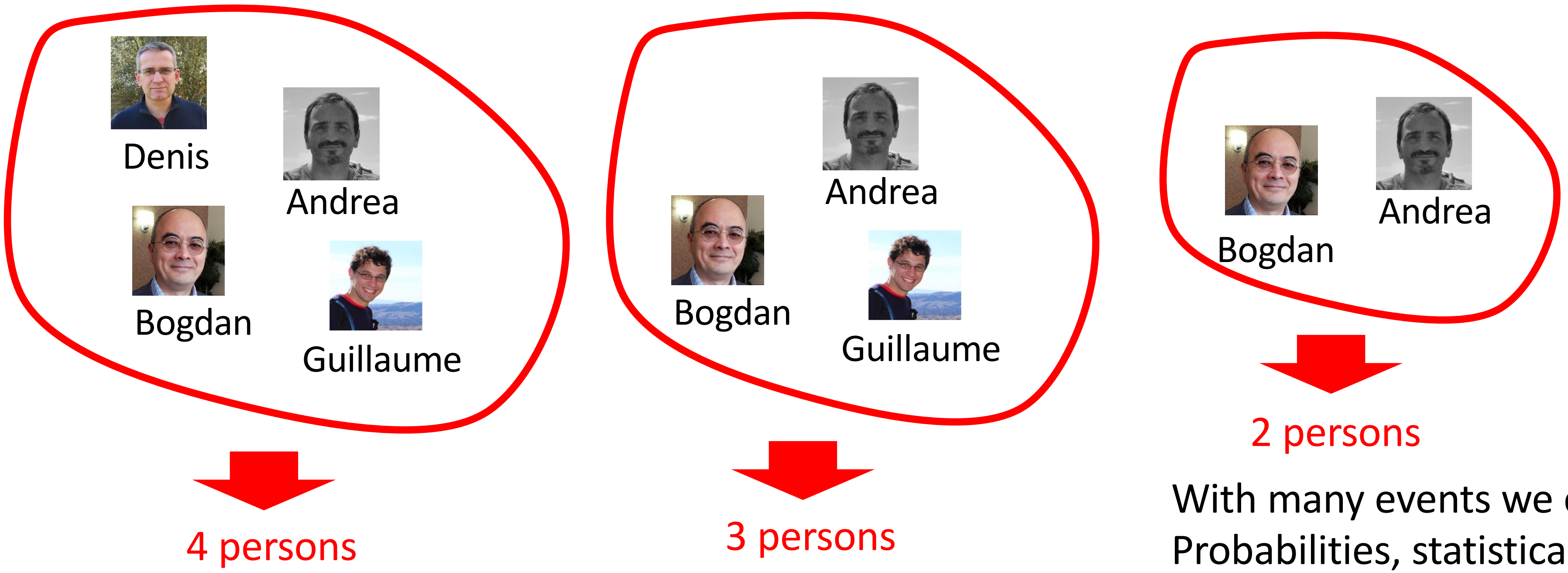


The counting statistic problem

I want to count people



A more specific example

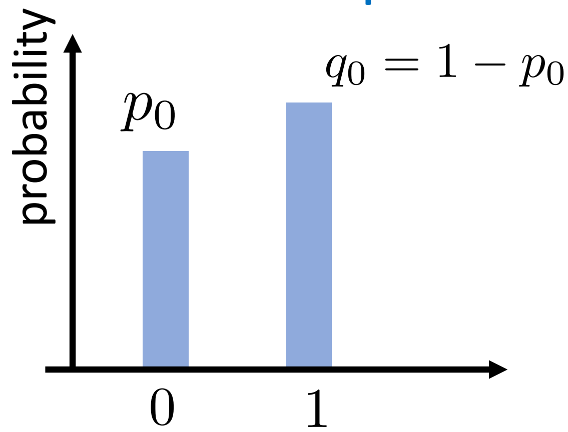


With many events we can do
Probabilities, statistical analysis

I assign a qubit to each person

$$\left| \right. $$$$\left. \right\rangle = \sqrt{p_0}|0\rangle + \sqrt{1-p_0}|1\rangle$$

Measuring the qubit gives the probability



Demystifying QC Illustration with qiskit

```
[1]: import numpy as np
     from qiskit import *
     %matplotlib inline
     import math

     from qiskit.visualization import plot_histogram
```

Creation of the circuit

```
[2]: nq=1
     nc=1
     qr = QuantumRegister(nq, 'q')   # qubit of interest + register qubits
     cr = ClassicalRegister(nc, 'c')    # classical register
     # name of the circuit
     mycircuit = QuantumCircuit(qr, cr)

     #make the rotation
     angle = 4*2*math.pi/12

     mycircuit.rx(angle,0)
     mycircuit.measure(0,0)

     #mycircuit.draw()
     print(mycircuit)
```

```
q_0: ┤ RX(2π/3) ├┤M├
c: 1/═════════════╩═
                  0
```

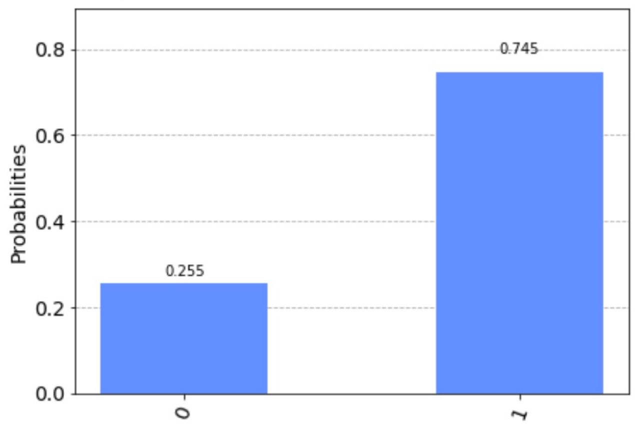
Running the circuit

```
[3]: # building our own normalized histo
     # Running the code !
     backend = Aer.get_backend('qasm_simulator')
     shots = 10000
     results = execute(mycircuit, backend=backend, shots=shots).result()
     answer = results.get_counts()

     print(answer)
     plot_histogram(answer)
```

```
{'0': 2546, '1': 7454}
```

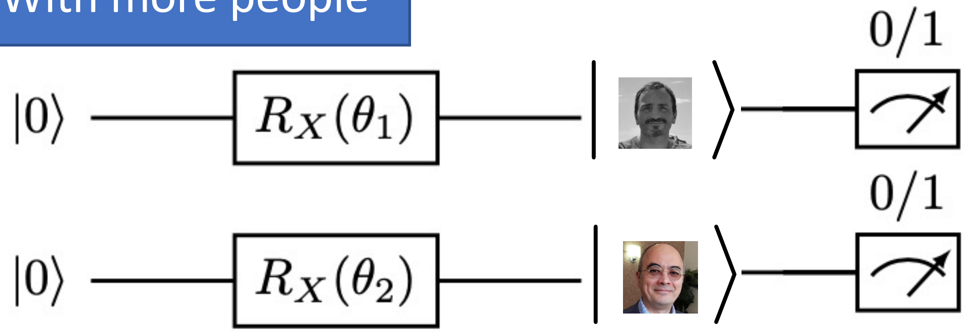
[3]:

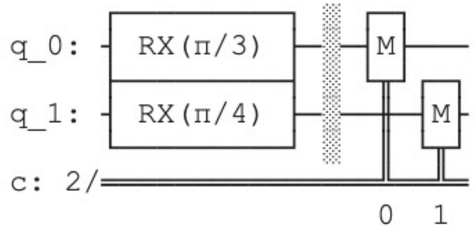


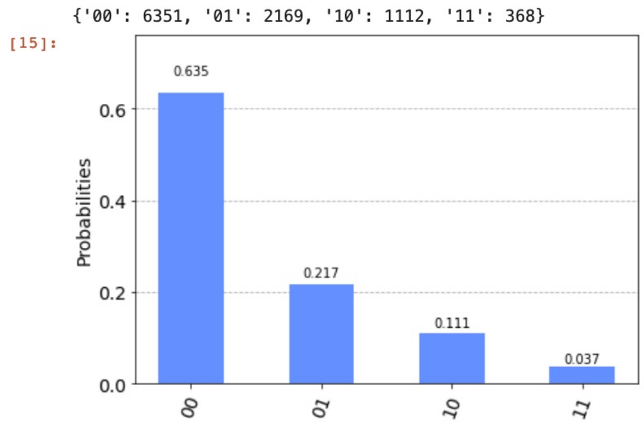
The counting statistic problem

In quantum systems

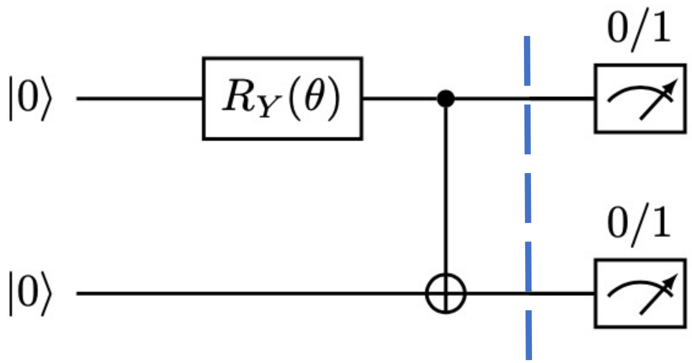
With more people

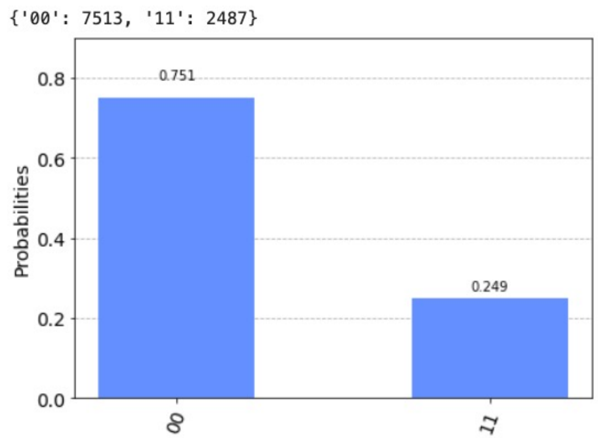






People can be entangled

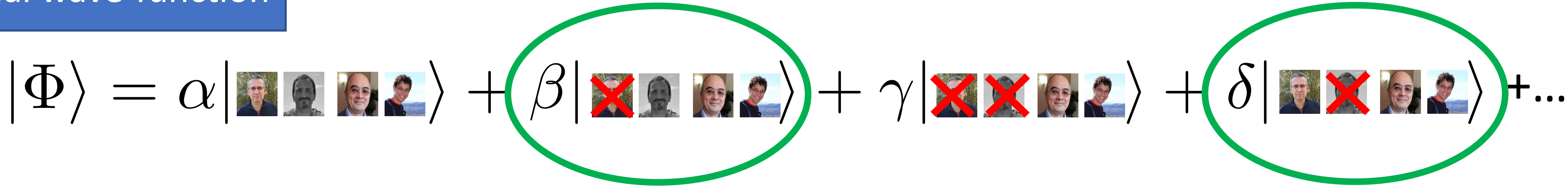


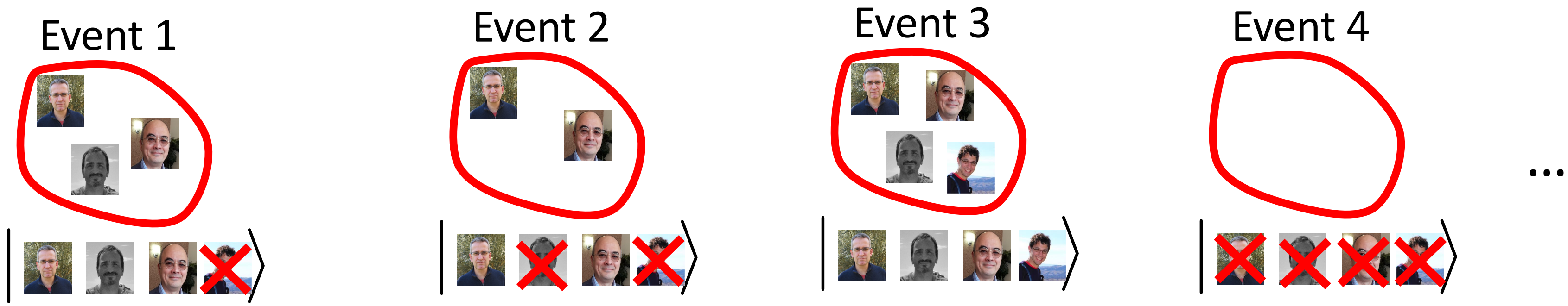


Here I created a Bell state

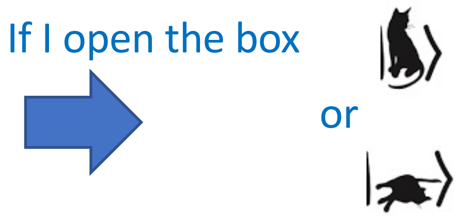
The counting statistic problem without destroying the wave-function

Initial wave-function



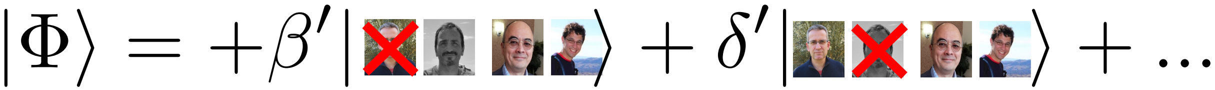


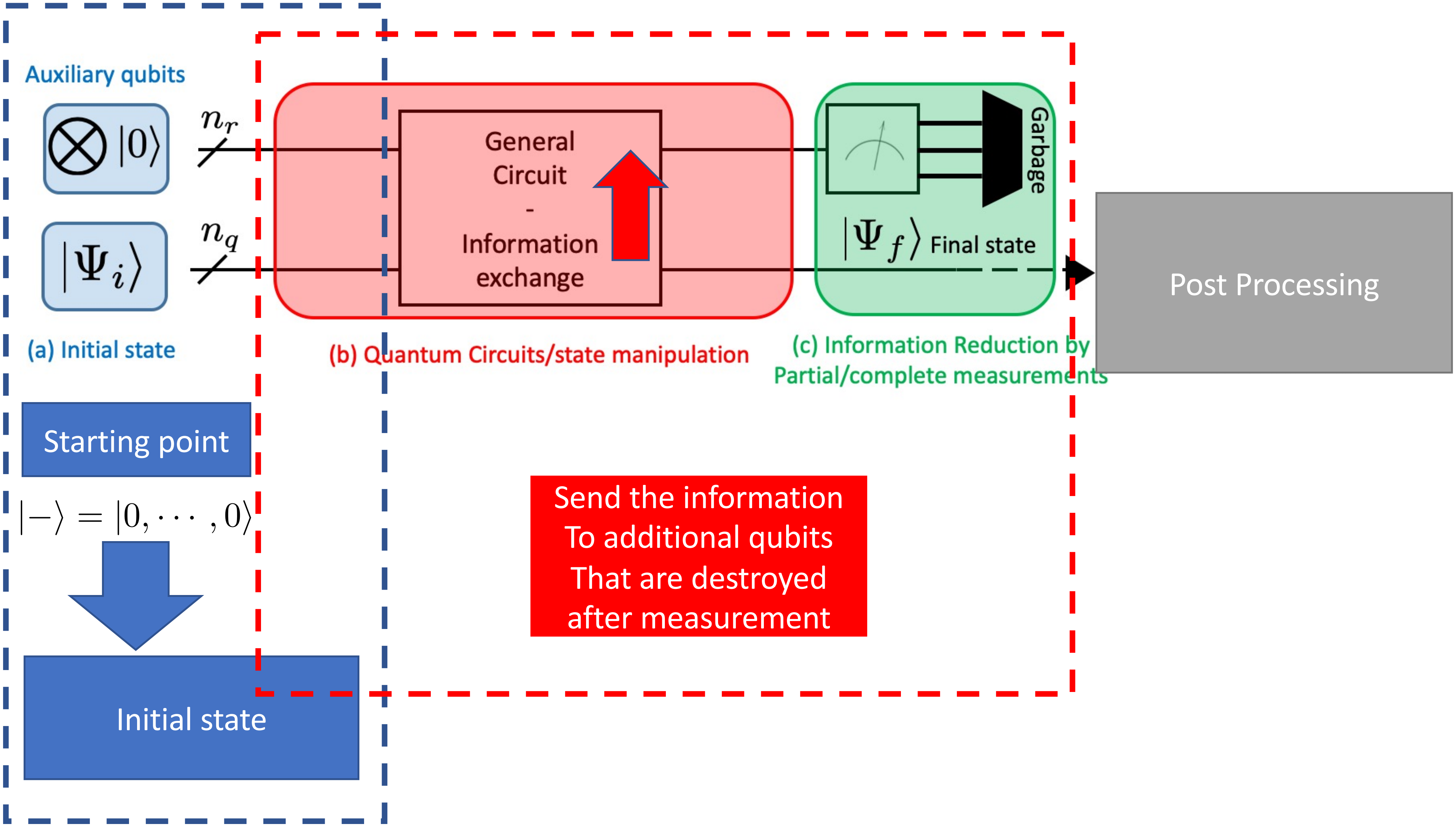
After the measurement the wave-function collapse to one of the state



A more difficult problem

I want to select the component with 3 persons without completely destroying it



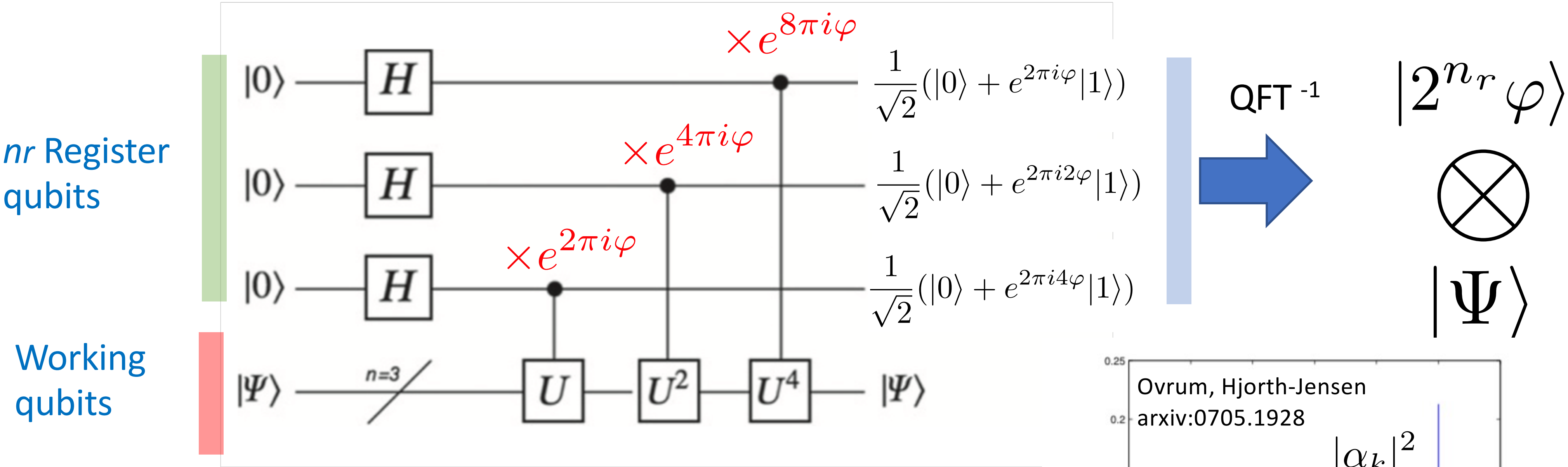

Auxiliary qubits
$\bigotimes |0\rangle$
n_r
$|\Psi_i\rangle$
n_q
(a) Initial state
Starting point
$|-\rangle = |0, \cdots, 0\rangle$
Initial state
General Circuit - Information exchange
(b) Quantum Circuits/state manipulation
Garbage
$|\Psi_f\rangle$ Final state
(c) Information Reduction by Partial/complete measurements
Post Processing
Send the information To additional qubits That are destroyed after measurement

The quantum-Phase estimate (QPE) algorithm

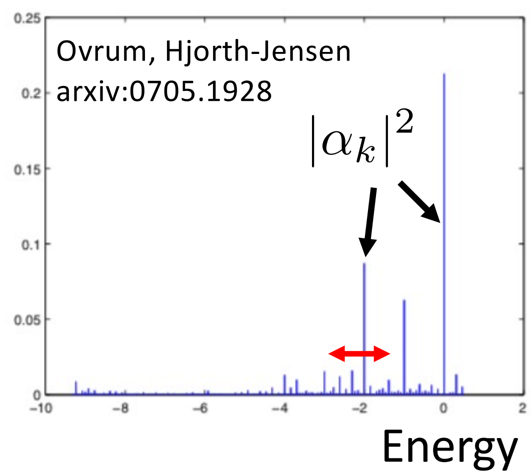
For eigenvalue problems

Assume a unitary operator U

Assume an eigenstate $|\Psi\rangle$ Such that $U|\Psi\rangle = e^{2\pi i\varphi}|\Psi\rangle$



General Case **Example**



$$|\Psi\rangle = \sum_k \alpha_k |\phi_k\rangle \xrightarrow{\text{QPE}} \sum_k \alpha_k |\theta_k 2^{n_r}\rangle \otimes |\phi_k\rangle$$

register eigenstate

FIG. 7: Pairing model simulated with 24 qubits, where 14 were simulation qubits, i.e. there are 14 available quantum levels, and 10 were work qubits. The correct eigenvalues are $0, -1, -2, -3, -4, -5, -6, -8, -9$. In this run we did not divide up the time interval to reduce the error in the Trotter approximation, i.e., $I = 1$.

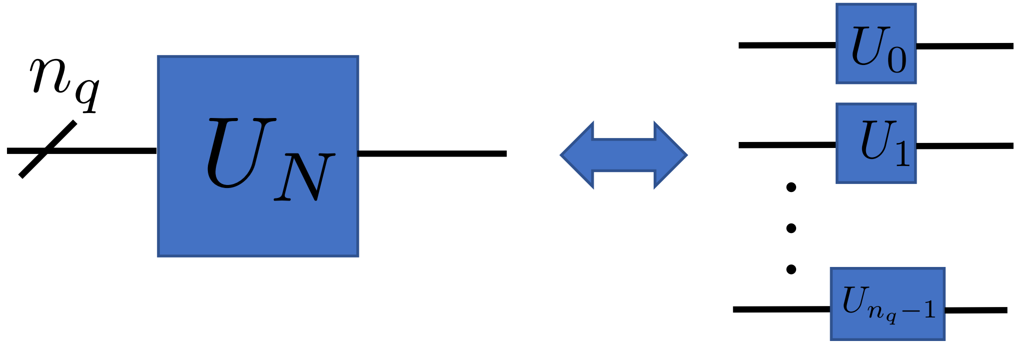
Simple Idea: take the phase proportional to the number of persons!

Practical details

$$U_N = \prod_j U_j$$

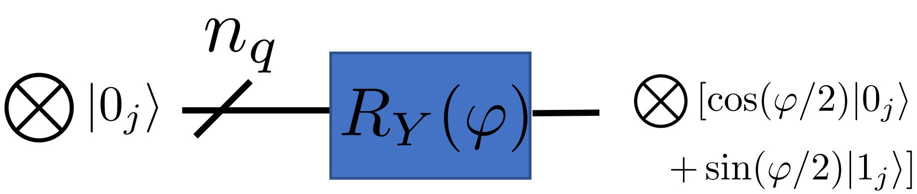
$$U_i = |0_i\rangle\langle 0_i| + \exp(i\pi/2^{n_0-1})|1_i\rangle\langle 1_i|$$

$$U_i = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/2^{n_0-1}} \end{bmatrix}$$

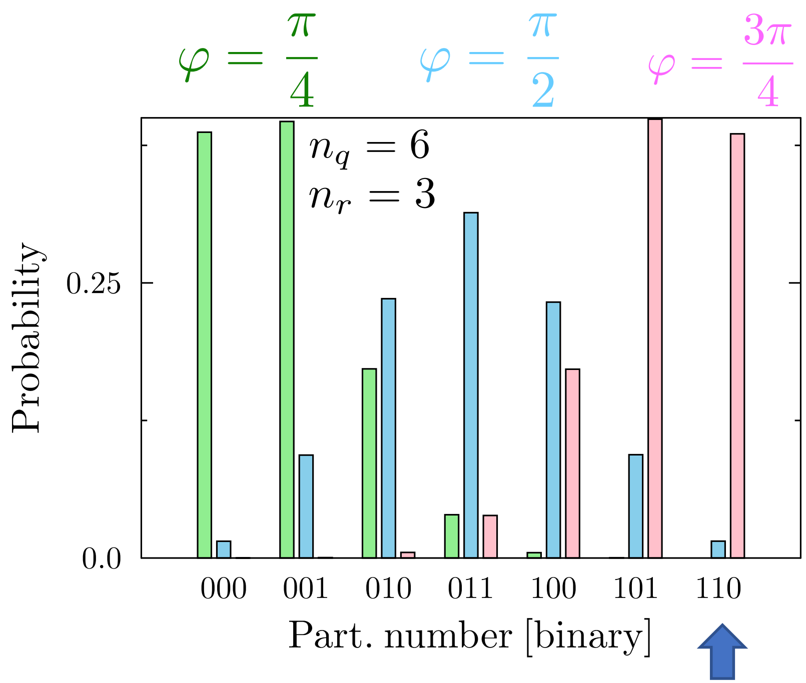


Example: Qubit counting statistics

Initial state



$$P(A) = C^A_{n_q} p^A (1-p)^{n_q - A}$$
$$p = \sin^2(\varphi/2)$$

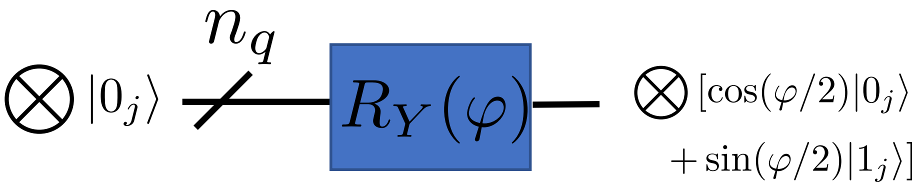


$$6/2^4 = 1/2 + 1/4 + 0/8 \equiv [110]$$

Calculation made with the IBM Qiskit python package

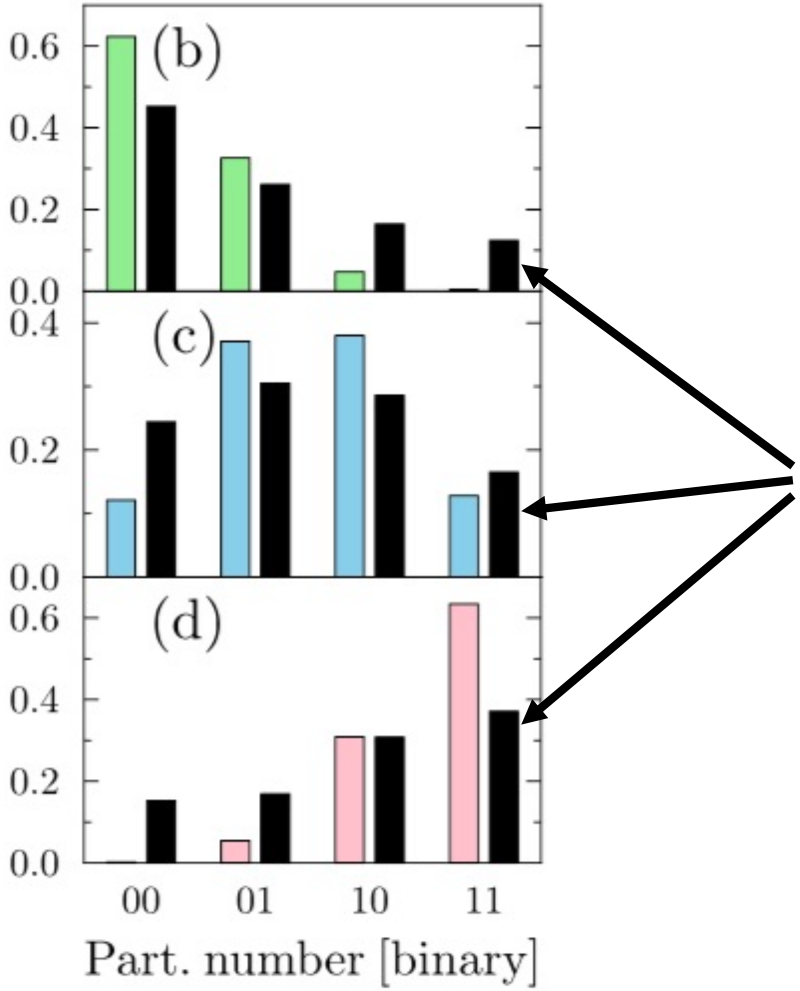
Example: Qubit counting statistics

Initial state



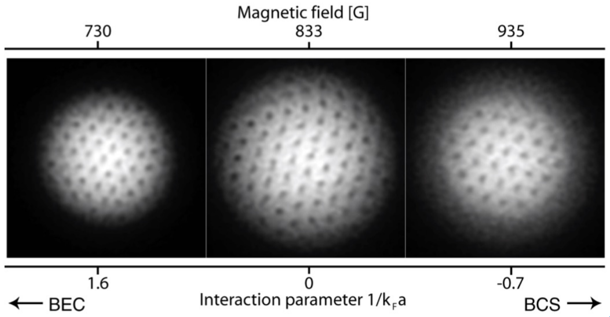
$$P(A) = C_{n_q}^{A} p^{A} (1-p)^{n_q - A}$$
$$p = \sin^2(\varphi/2)$$

3 qubits and 2 register qubits

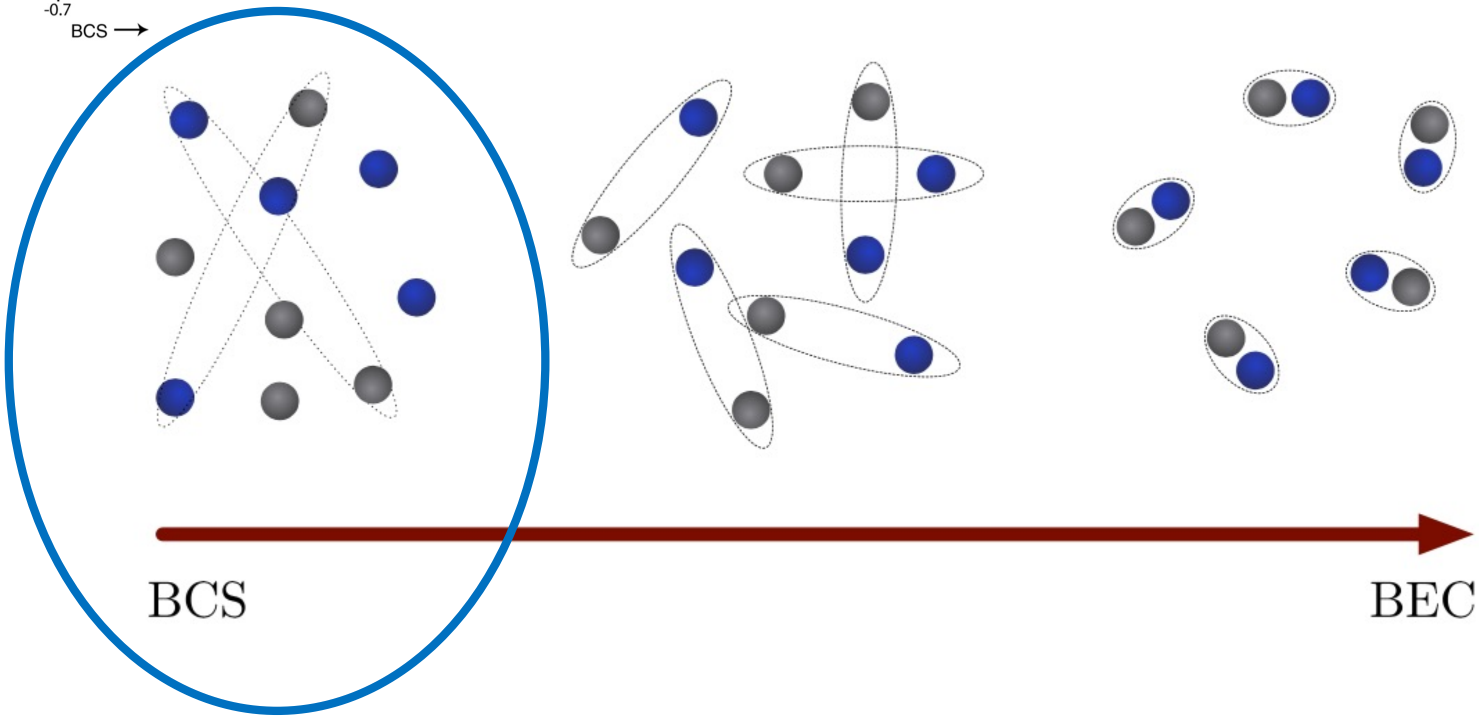


Result on IBMq_vigo 5 qubits device

But what is the connection with interacting systems ???



Cooper pairs and superfluidity are rather universal phenomena: (condensed matter, Atomic physics, Nuclear physics, …)

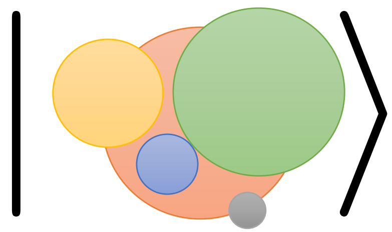


This problem is an archetype of spontaneous symmetry breaking.
A “easy” way to describe it is to break the particle number symmetry, i.e. consider wave-function that mixes different particle number

Example $|\Phi_0\rangle = \Pi_i (u_i + v_i a_i^\dagger a_{\bar{i}}^\dagger)|-\rangle$

→ Mixes states with 0, 2, 4, … particles

We say that a symmetry (particle number) is broken



But ultimately number of Particle should be restored !

A schematic view

Making projection on particle number

$$\bigotimes_{n_r} |0\rangle$$

$$|\Psi\rangle = \sum_k \alpha_k |\varphi_k\rangle$$

Information Transfer on the mixing of particle number

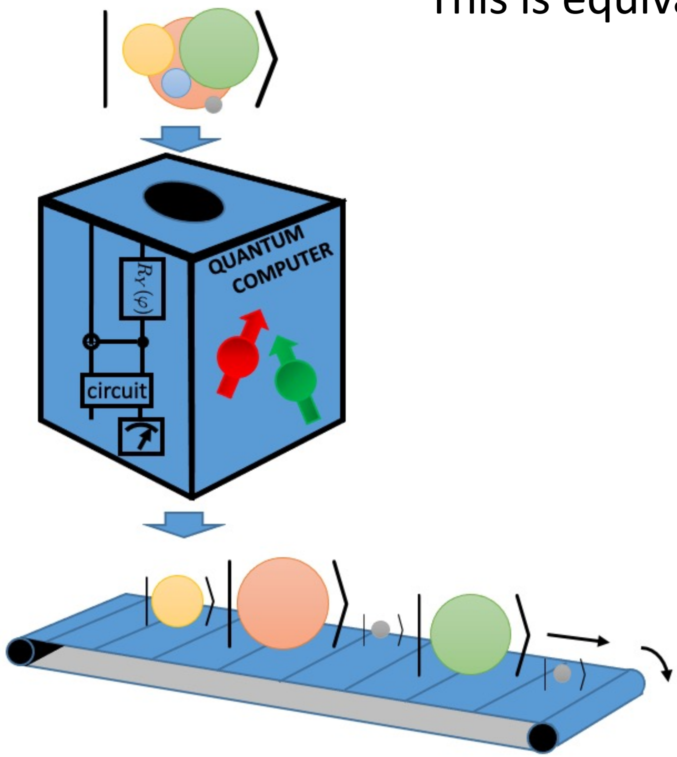
$$\sum_k \alpha_k |01001 \cdots 1\rangle \otimes |\varphi_k\rangle$$

= Particle number written as a binary number

We can measure the register qubit

This is equivalent to project on $|\varphi_k\rangle$

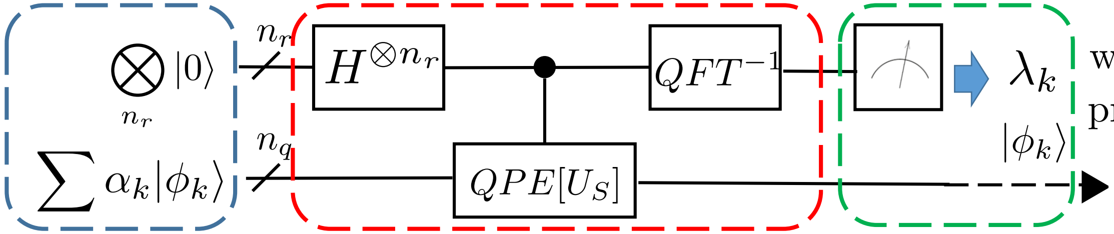
An even more schematic view



Then I can use this Wave-function for post-processing

D. Lacroix, “*Symmetry-Assisted Preparation of Entangled Many-Body States on a Quantum Computer*”, PRL 125, 230502 (2020).

Eigenvalues-Ground state and excited states



Initial configuration with symmetry breaking state

Symmetry operator eigenvalues encoding In the register qubits

Symmetry restoration through measurement

Measurement

Example of an event:

$$|011\cdots010\rangle^{(\lambda)} \otimes |\phi_{A^{(\lambda)}}\rangle$$

$$= A^{(\lambda)}$$

BCS/HFB state

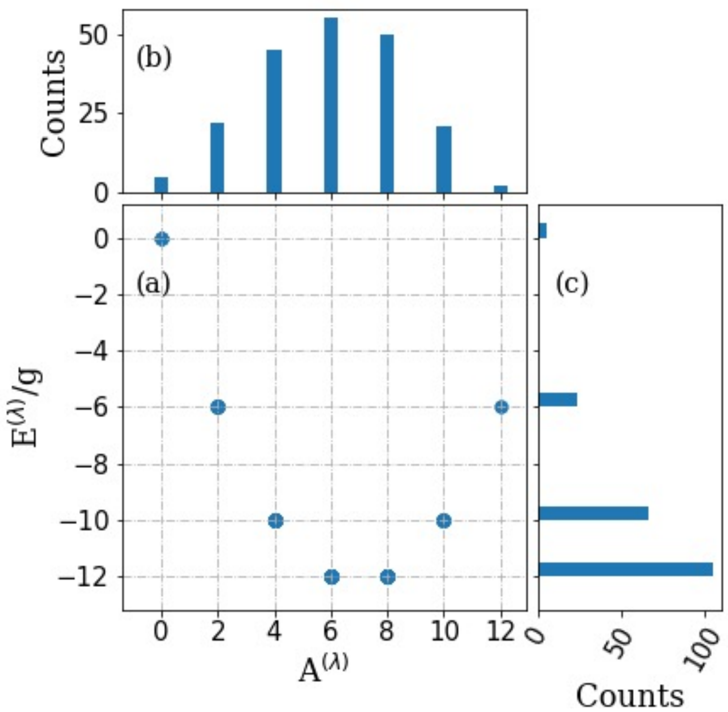
$$|\Psi\rangle = \prod_n \left[\cos\left(\frac{\varphi}{2}\right) I_n \otimes I_{n+1} + \sin\left(\frac{\varphi}{2}\right) Q_n^+ Q_{n+1}^+\right] |-\rangle$$

Measurement

Projected BCS or HFB state with varying number Of particles

$$\langle \phi_{A^{(\lambda)}} | H | \phi_{A^{(\lambda)}} \rangle$$

6 pairs



H was encoded on the full Fock space with $A < n_q$

For the degenerate case, this should give the exact solution

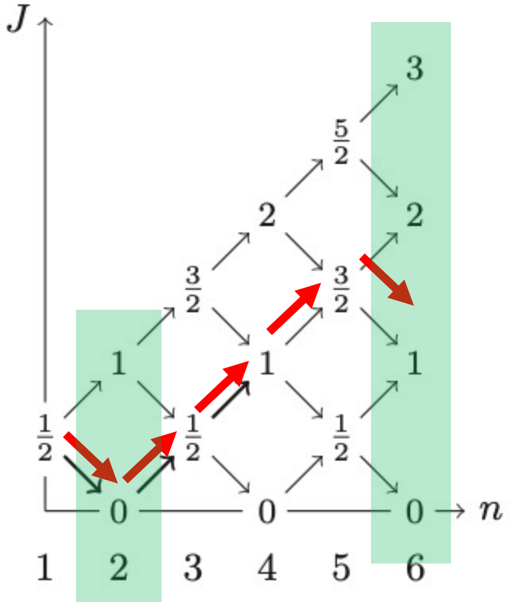
Exact solution

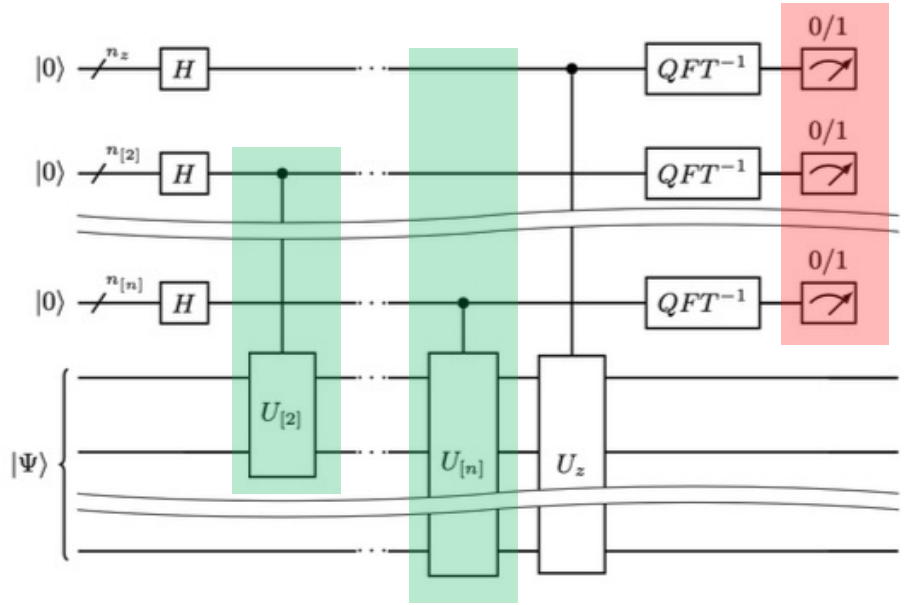
$$E/g = -\frac{1}{4}(A-\nu)(2n_q - A - \nu - 2).$$

Wave-function on qubits

$$|\Psi\rangle = \sum_{s_i \in \{0,1\}} \Psi_{s_1,\cdots,s_N} |s_1,\cdots,s_n\rangle. \quad \Rightarrow \quad |\Psi\rangle = \sum_{S,M} \sum_{g=1}^{d_{S,M}} c^g_{S,M} |S,M\rangle_g.$$

Sequential method to create a total spin





P. Siwach and DL, arXiv:2106.10867

Wave-function on qubits

$$|\Psi\rangle = \sum_{s_i\in\{0,1\}} \Psi_{s_1,\cdots,s_N}|s_1,\cdots,s_n\rangle. \quad \Rightarrow \quad |\Psi\rangle = \sum_{S,M}\sum_{g=1}^{d_{S,M}} c^g_{S,M}|S,M\rangle_g.$$

Sequential method to create a total spin

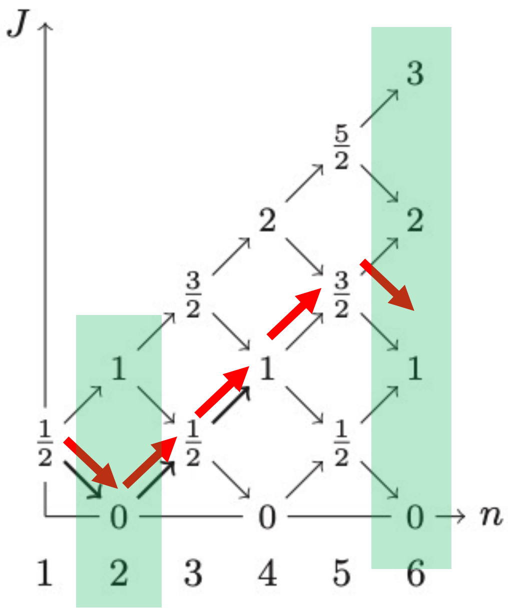
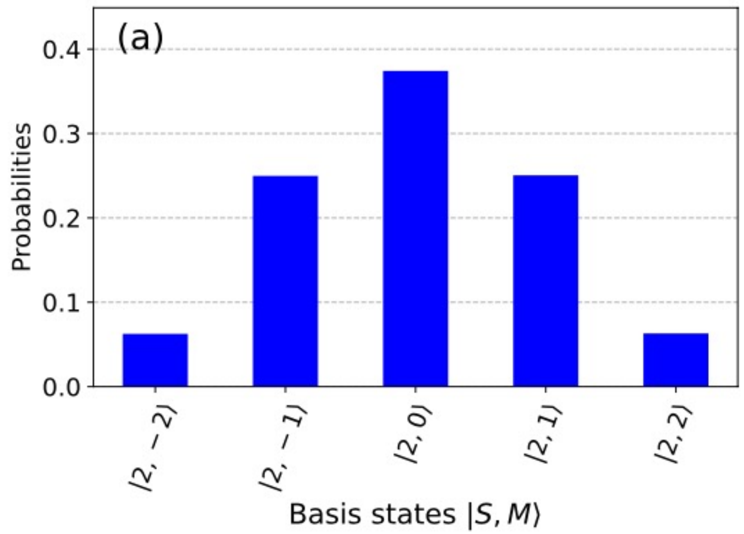
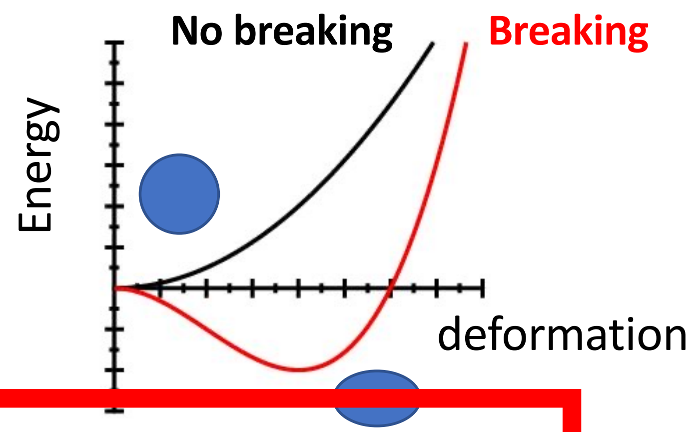


Illustration $|\Psi\rangle = \bigotimes_n H|0\rangle$



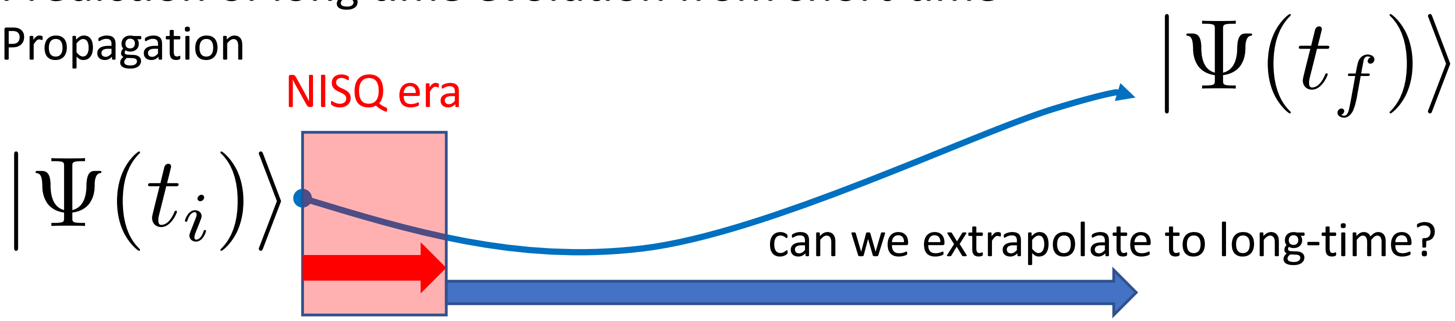
P. Siwach and DL, arXiv:2106.10867

The recent applications we made (in many-body systems)



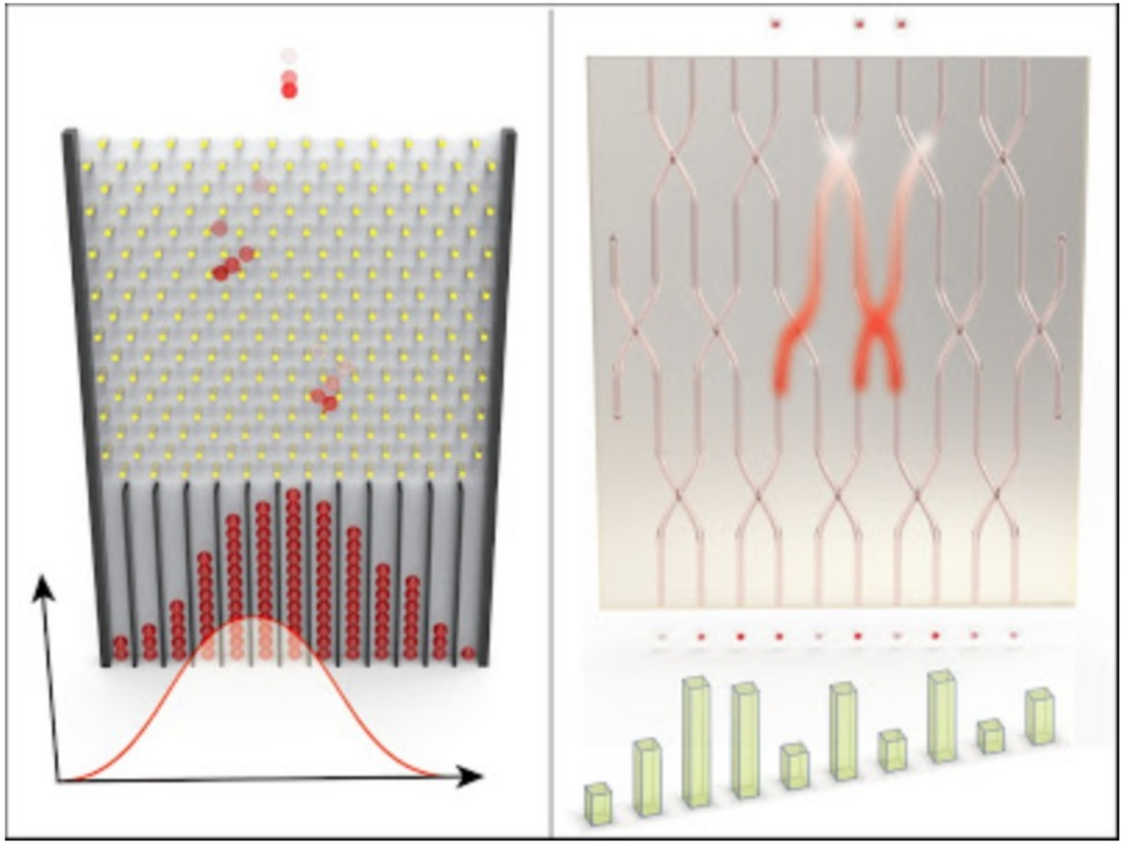
- Breaking symmetries and restoration of symmetries in many-body systems on quantum computers
  - Application to the counting of particle number (for superfluid systems)

- Replacing bosons by pairs of fermions to probe quantum supremacy
  - Prediction of long time evolution from short-time Propagation

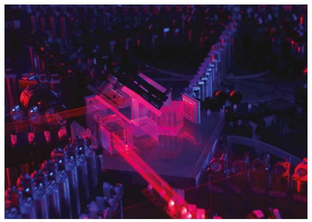


Original motivation : probe quantum supremacy with Fermi Cooper pairs

Bosons Pair sampling problem

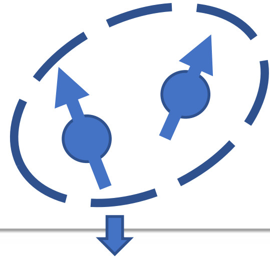


Mainly in photonic quantum computers

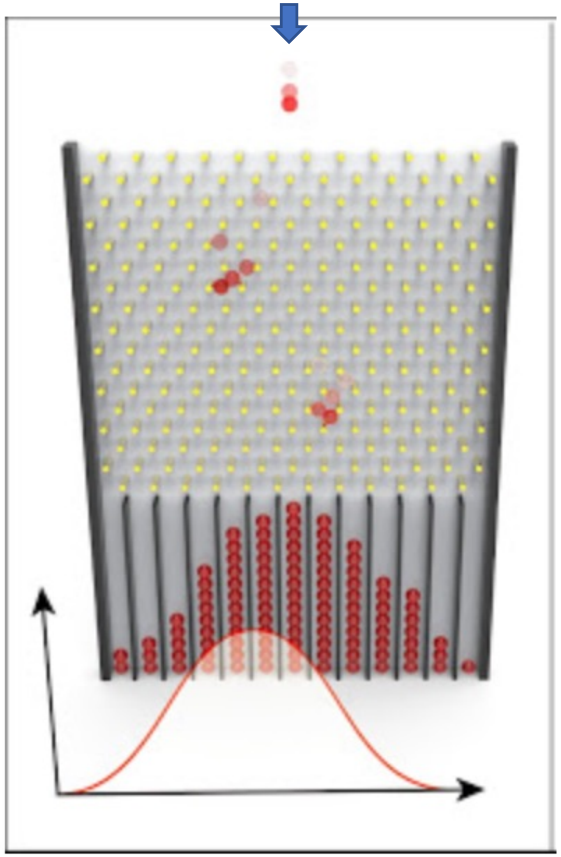


This photonic computer did a task that would take a classical computer 2.5 billion years.

Bosons



Idea:
2 fermions are similar to 1 boson



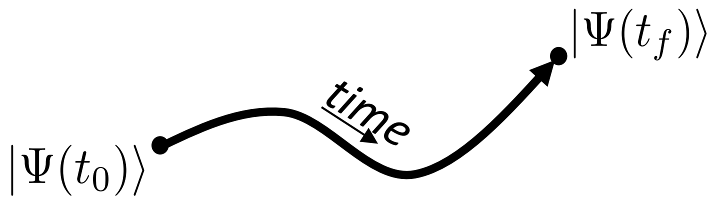
$|\Psi(t_i)\rangle$

$|\Psi(t_f)\rangle$

- Advantage : can be made on any device
- But for this we need to solve Efficiently the evolution

Solution of Schroedinger Equation on classical and quantum devices

Schrödinger equation



Formal solution

$$i\hbar\frac{d}{dt}|\Psi\rangle = H|\Psi\rangle \quad \Rightarrow \quad |\Psi(t)\rangle = e^{\frac{1}{i\hbar}(t-t_0)H}|\Psi(t_0)\rangle$$

H is usually a big matrix

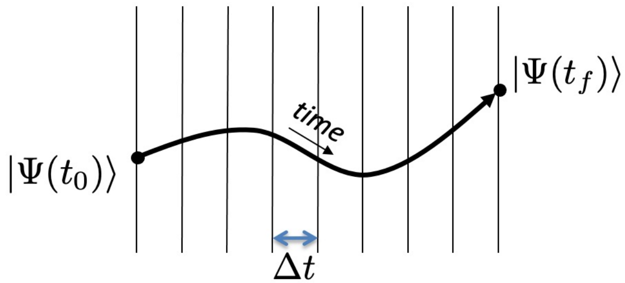
Integrating the Schroedinger Eq. on a classical computer

$$i\hbar\dot{\mathbf{F}}(t) = \mathbf{H}\times\mathbf{F}(t)$$

$$\mathbf{F}(t+\Delta t) = \exp\left(\frac{\Delta t}{i\hbar}\mathbf{H}\right)\times\mathbf{F}(t)$$

Time discretization

time: $\{t_i\}$ time-step: Δt



Direct

$$\exp\left(-\frac{\Delta t}{i\hbar}\mathbf{H}\right) \simeq 1 + \frac{\Delta t}{i\hbar}\mathbf{H} + \frac{1}{2!}\left(\frac{\Delta t}{i\hbar}\mathbf{H}\right)^2 + \cdots$$

$(\Delta t)^n$, non-unitary, any dim.

Crank-Nicholson

$$\mathbf{F}(t+\Delta t) = \frac{1-\frac{\Delta t}{2i\hbar}\mathbf{H}}{1+\frac{\Delta t}{2i\hbar}\mathbf{H}}\mathbf{F}(t)$$

$(\Delta t)^2$, unitary, 1D only

Split-Operator

$$\mathbf{F}(t+\Delta t) \simeq e^{-i\Delta t\frac{\mathbf{P}^2}{4\hbar m}}e^{-\frac{i}{\hbar}\Delta t\mathbf{V}}e^{-i\Delta t\frac{\mathbf{P}^2}{4\hbar m}}\times\mathbf{F}(t)$$

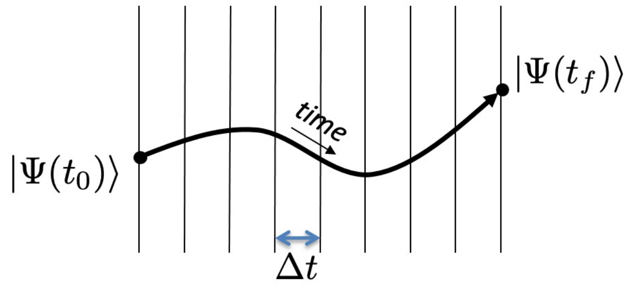
$(\Delta t)^2$, unitary, any dim.

Solution of Schroedinger Equation on classical and quantum devices

Integrating the Schroedinger Eq. on a quantum computer

$$|\Psi(t)\rangle = e^{\frac{1}{i\hbar}(t-t_0)H}|\Psi(t_0)\rangle$$

Quantum computers can only perform unitary transformations



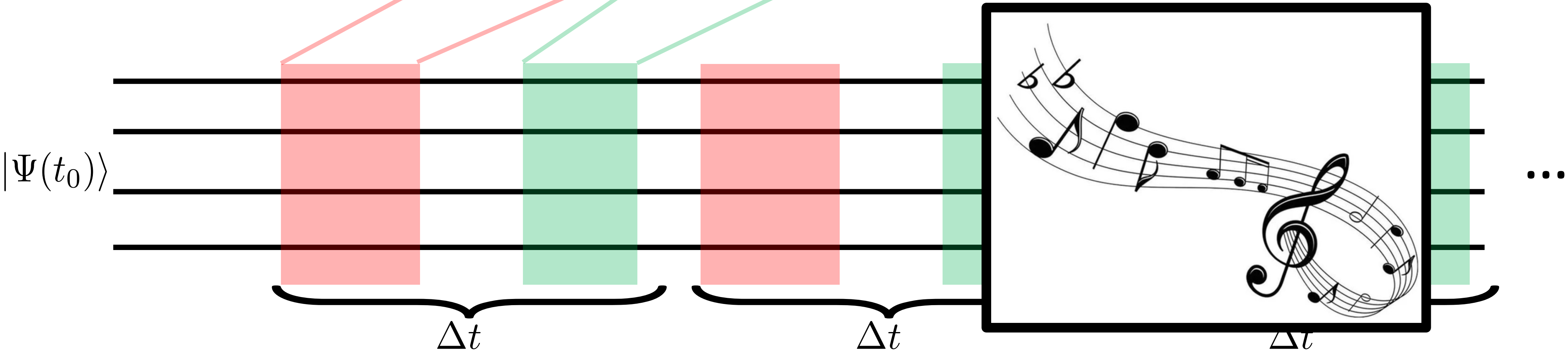
1. Time discretization

2. Decomposition of H into elementary blocks $H = \sum_l H_l$

3. Use a transformation (Trotter-Suzuki) $e^{-ix(A+B)} = \left(e^{-iAx/N}e^{-iBx/N}\right)^N + \mathcal{O}(t^2/N)$

Example : $e^{i\Delta t H_1/\hbar} = e^{-i\Delta t H_1/\hbar}e^{-i\Delta t H_2/\hbar}$

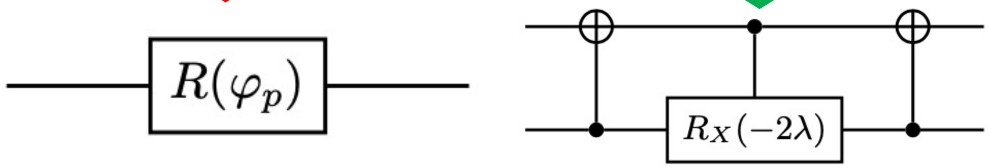
4. Transforms to circuit



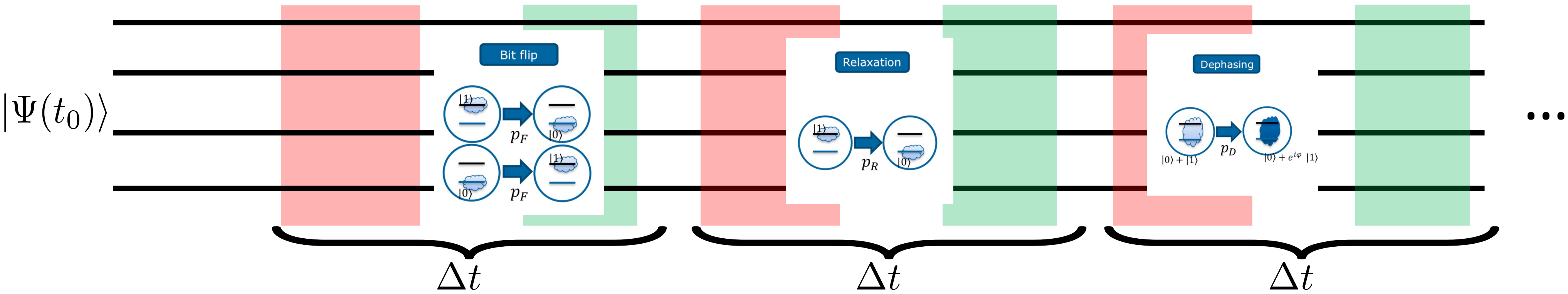
Pairing Hamiltonian

$$H_{\mathrm{P}} = \sum_{i>0} \varepsilon_i (a_i^\dagger a_i + a_{\bar{i}}^\dagger a_{\bar{i}}) - g \sum_{i,j>0} a_i^\dagger a_{\bar{i}}^\dagger a_{\bar{j}} a_j$$

$$\prod_p \begin{pmatrix} 1 & 0 \\ 0 & \exp\left(-2i\widetilde{\varepsilon}_p \Delta t\right) \end{pmatrix} \qquad \prod_{p>q} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\lambda_{pq}) & i\sin(\lambda_{pq}) & 0 \\ 0 & i\sin(\lambda_{pq}) & \cos(\lambda_{pq}) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

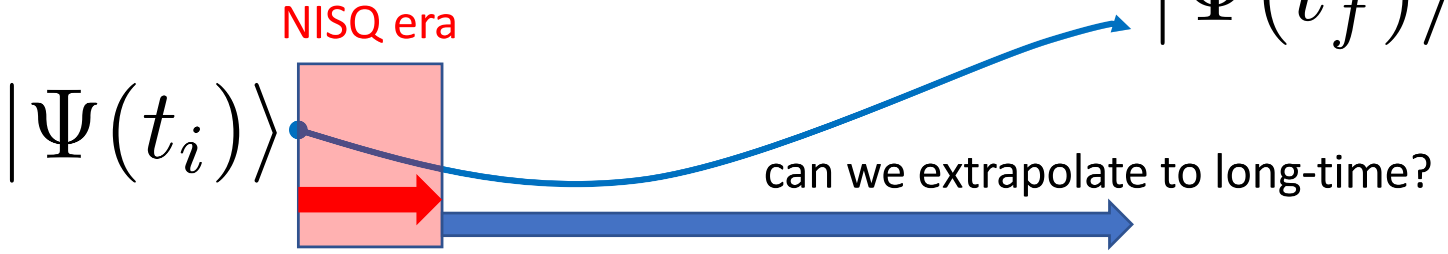


The problem is that we can nowadays perform only few operations and with a limited *fidelity*



Predicting long time dynamics from short-time evolution

➡ Prediction of long time evolution from short-time Propagation



What is the physical content of short-time evolution ?

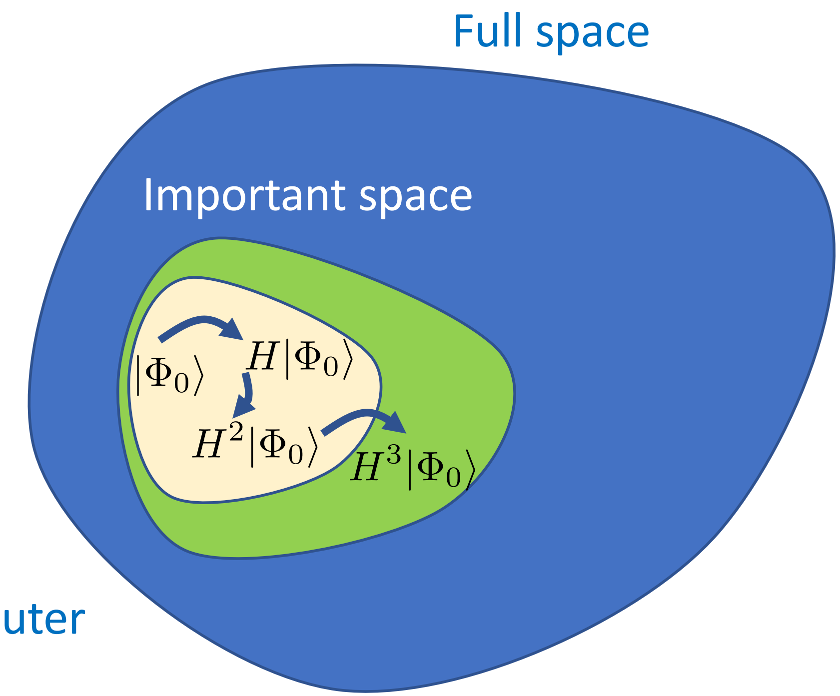
$$|\Phi(t)\rangle = \left(1 - itH + \frac{(-it)^2}{2!}H^2 + \cdots\right)|\Phi(0)\rangle$$

➡ $H^K|\Phi(0)\rangle$

Are the so-called Krylov states

But they cannot be computed easily on a quantum computer

➡ We propose instead to compute $\langle H^K\rangle_0$



Hamiltonian moments calculation on a quantum computer

With minimal qubits number

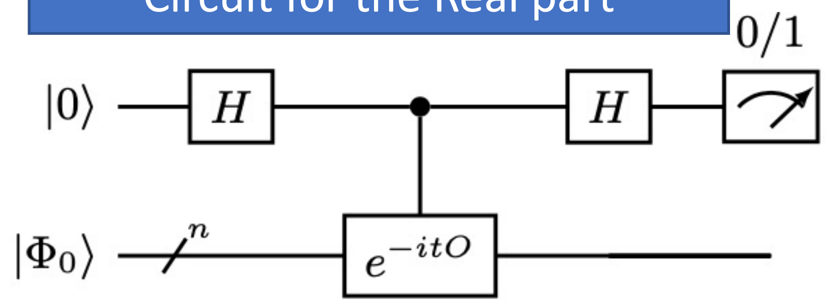
Generating function concept

$$F(t) = \langle \Phi_0 | e^{-itH} | \Phi_0 \rangle$$

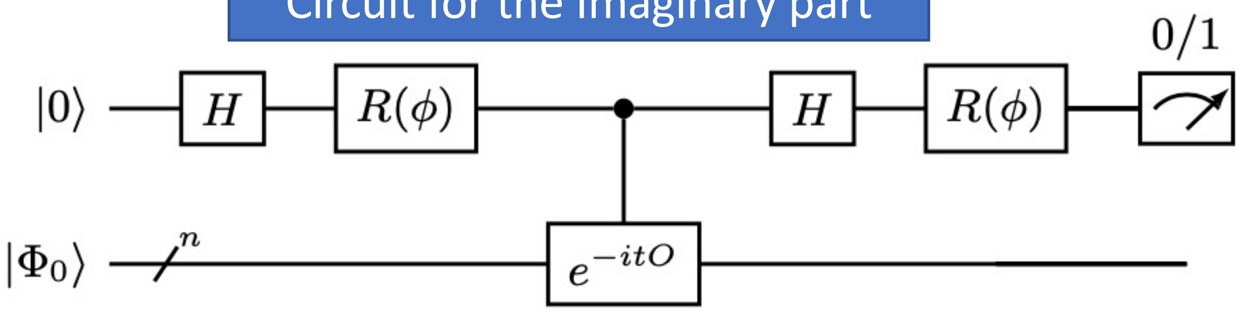
$$F(t) = 1 - it\langle H \rangle_0 + \frac{(-it)^2}{2}\langle H^2 \rangle_0 + \cdots \Rightarrow \langle H^K \rangle_0 = i^K \left. \frac{d^K F(t)}{dt^K} \right|_{t=0}$$

Practical method to get *F(t)*

Circuit for the Real part



Circuit for the Imaginary part



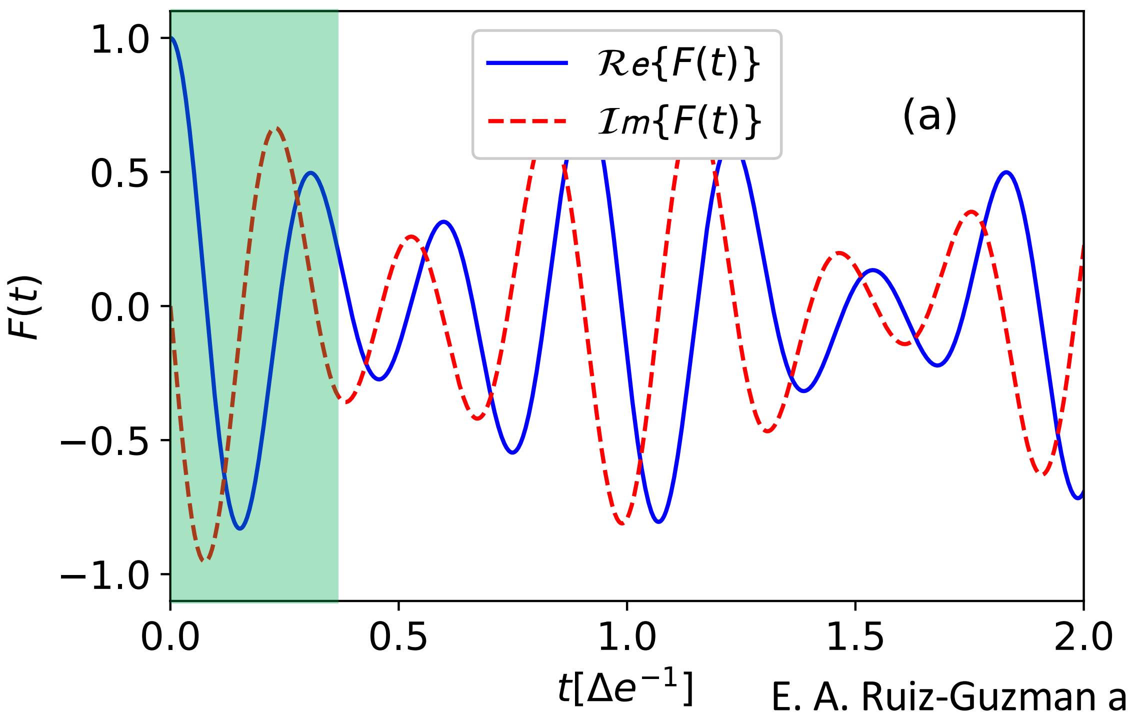
Hamiltonian moments calculation on a quantum computer
With minimal qubits number

Generating function concept

$$F(t) = \langle \Phi_0 | e^{-itH} | \Phi_0 \rangle$$

$$F(t) = 1 - it\langle H \rangle_0 + \frac{(-it)^2}{2}\langle H^2 \rangle_0 + \cdots \quad \Rightarrow \quad \langle H^K \rangle_0 = i^K \left. \frac{d^K F(t)}{dt^K} \right|_{t=0}$$

Illustration for the cooper pair problem



finite difference made on a classical computer $\Rightarrow \langle H^K \rangle_0$

E. A. Ruiz-Guzman and DL, in preparation

Next use the moments for post-processing

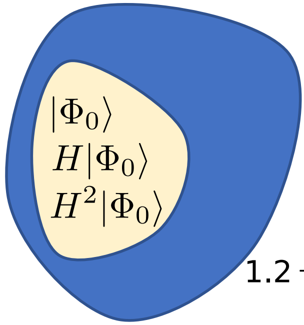
$\langle H^K \rangle_0$

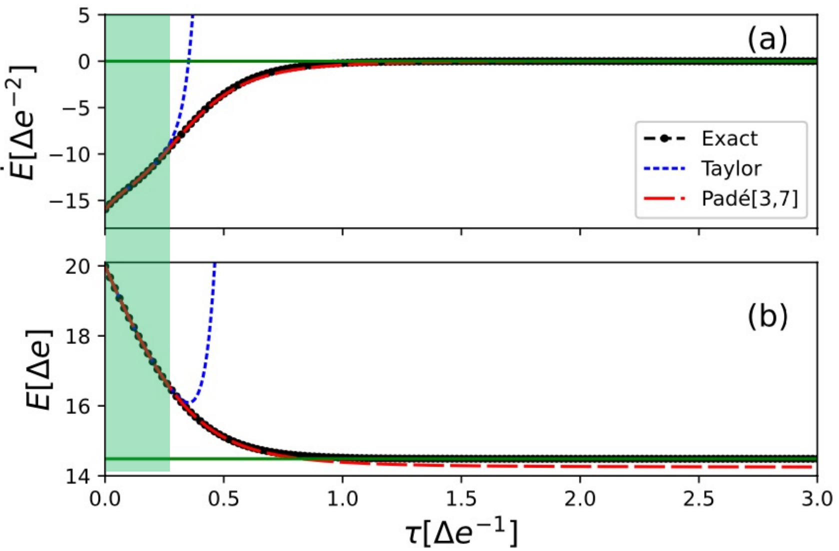
Ground state property (imaginary time evolution)

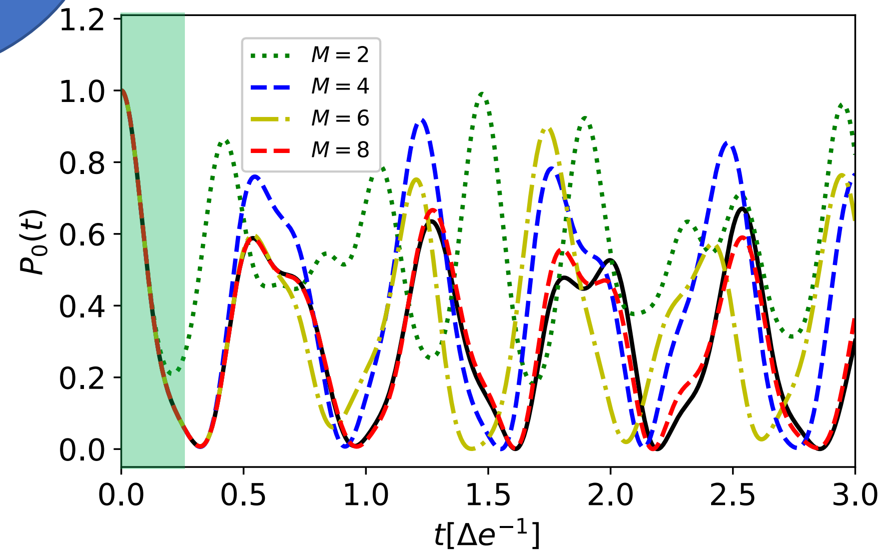
Evolution: Krylov without Krylov states

$$E(\tau) = \frac{\langle H e^{-\tau H} \rangle}{\langle e^{-\tau H} \rangle}$$

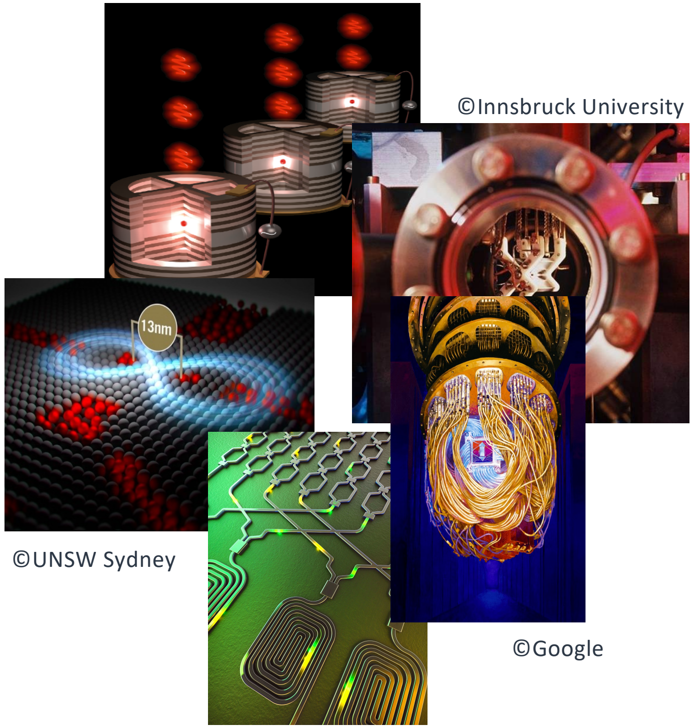
$$\frac{d}{d\tau} E(\tau) \simeq -\sum_{K=0}^{L-2} \frac{(-\tau)^K}{K!} \kappa_{K+2}.$$







E. A. Ruiz-Guzman and DL, arXiv:2104.08181



©C2N - CNRS

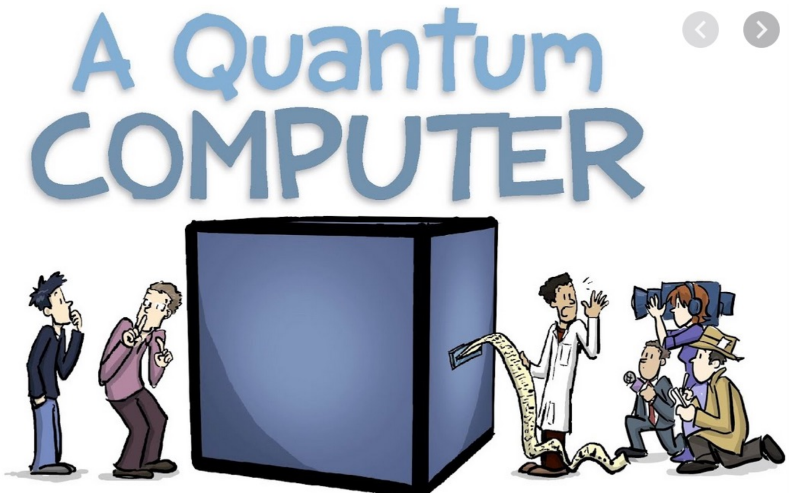
©Innsbruck University

©UNSW Sydney

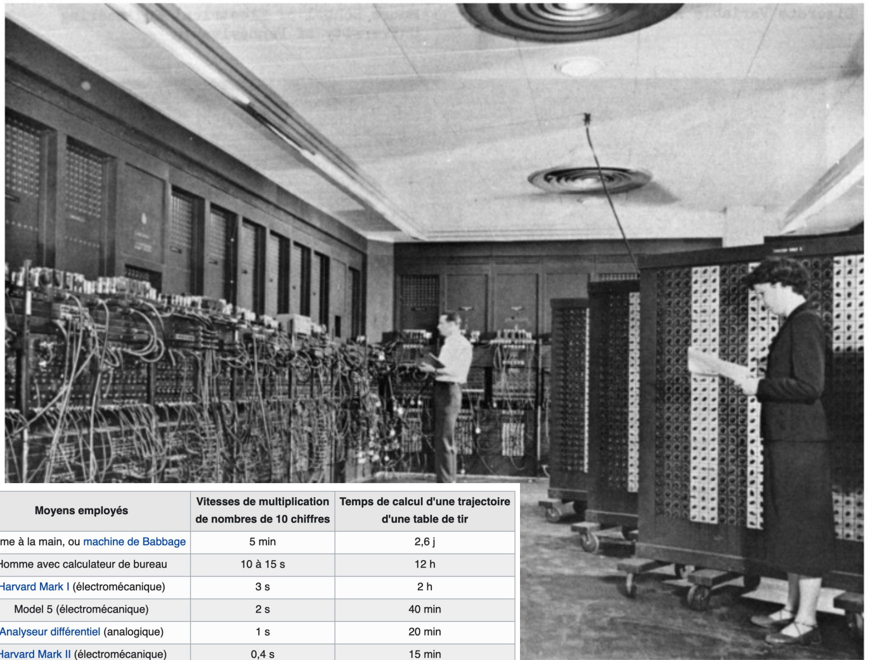
©Google

©U. of Bristol

- Quantum computing is a high risk/high benefit interdisciplinary field
- It might lead to unprecedented boost in theory (or more generally in complex problems)
- It leads to natural link between public research and private companies (IBM, Google, ...)
- Emerging QC programs in France



Eniac ~1950



Moyens employés	Vitesses de multiplication de nombres de 10 chiffres	Temps de calcul d'une trajectoire d'une table de tir
Homme à la main, ou machine de Babbage	5 min	2,6 j
Homme avec calculateur de bureau	10 à 15 s	12 h
Harvard Mark I (électromécanique)	3 s	2 h
Model 5 (électromécanique)	2 s	40 min
Analyseur différentiel (analogique)	1 s	20 min
Harvard Mark II (électromécanique)	0,4 s	15 min
ENIAC (électronique)	0,001 s	3 s

IBM ~2020

?

From B. Vulpescu