Noisy Intermediate-Scale Quantum algorithms

Alba Cervera-Lierta

Machine Learning in Quantum Physics and Chemistry

Warsaw, August 25, 2021

the matter lab

UNIVERSITY OF TORONTO

Outlook

0. What is Quantum Computation?
1. Quantum computing in the NISQ era
2. Variational Quantum Algorithms
3. Squeezing the NISQ lemon

Break

4. NISQ algorithms
5. NISQ horizon

Coding time! (if we have time)

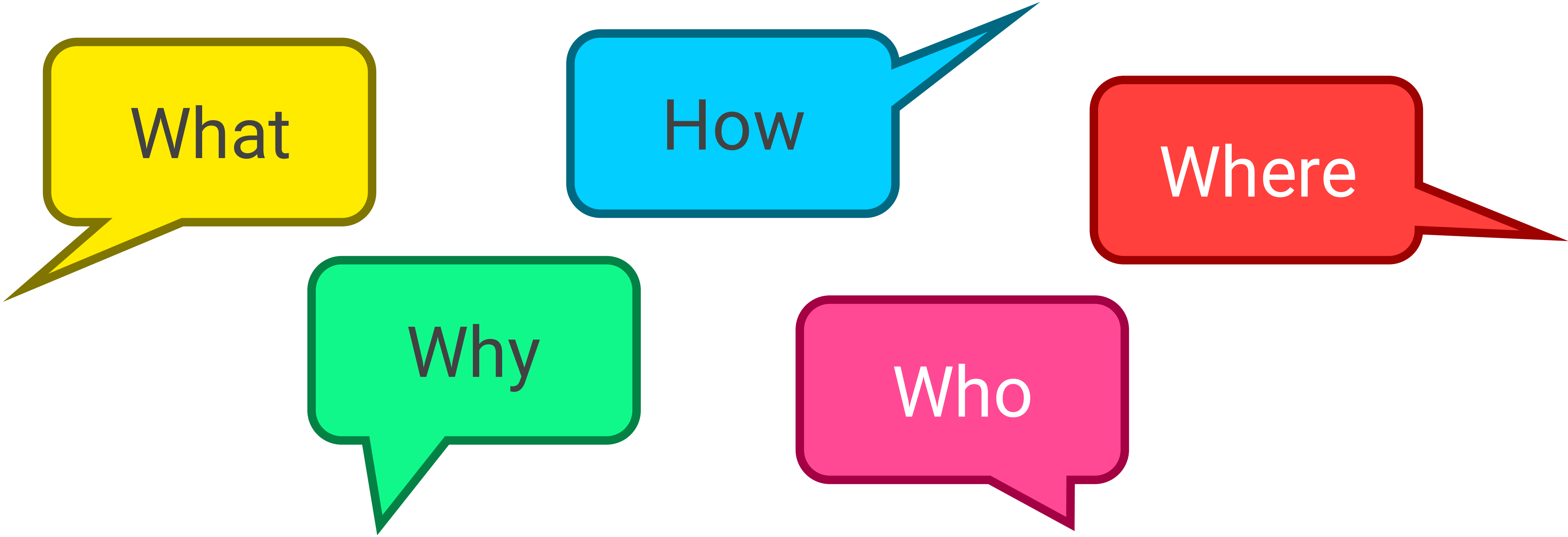
Slides:

`albacl.github.io`

Tutorials (Tequila):

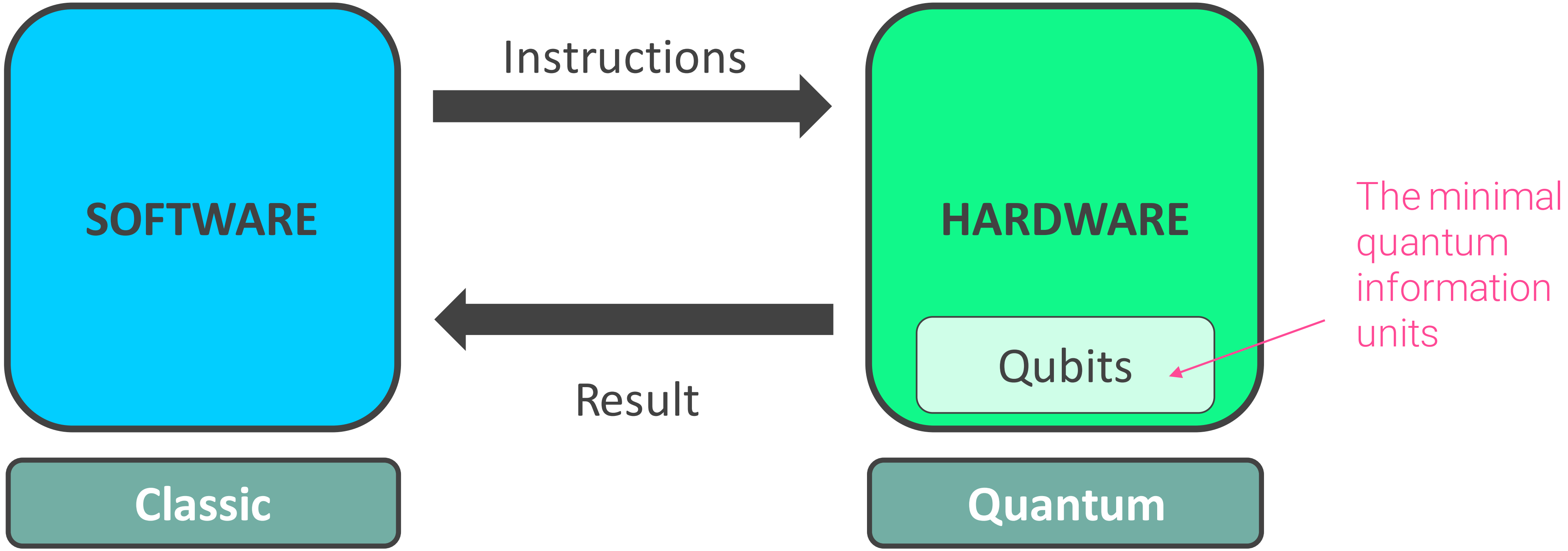
`github.com/AlbaCL/VQA_tutorials`

The basics of Quantum Computation



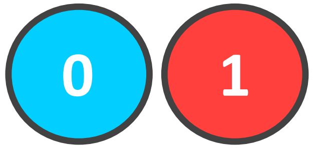
What is a quantum computer

A device capable of processing data in a quantum mechanical form.
A device that uses the properties of quantum mechanics to process data.



How does it work

Bit



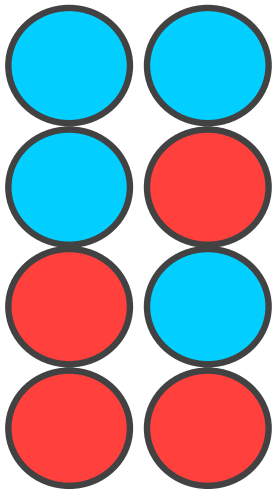
Qubit

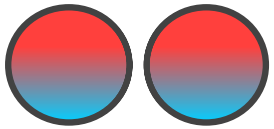
$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|0\rangle \equiv \begin{pmatrix}1\\0\end{pmatrix}$$

$$\alpha, \beta \in \mathbb{C}$$

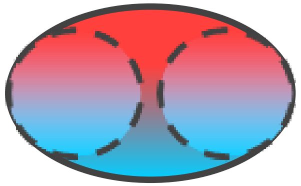
$$|1\rangle \equiv \begin{pmatrix}0\\1\end{pmatrix}$$





$$|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$$

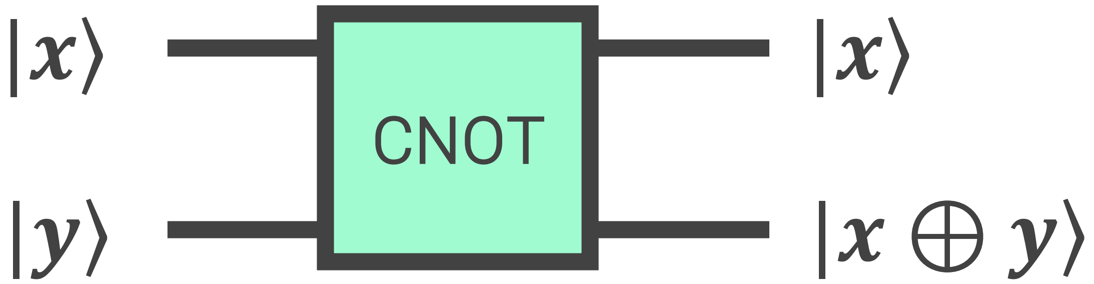
entanglement



$$|\psi\rangle \neq |\psi\rangle_1 \otimes |\psi\rangle_2$$

A new paradigm in computation

A single operation (logic gate) affects all posible qubit states.



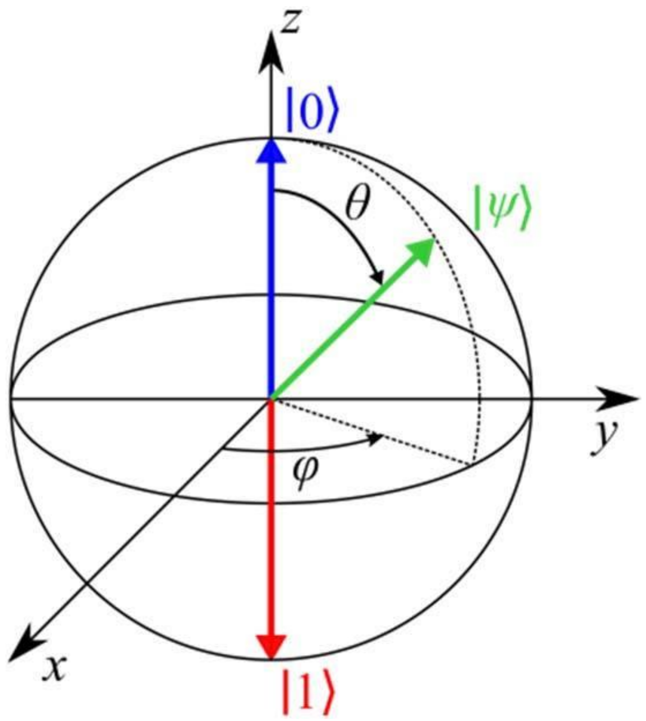
x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

$$|\psi_0\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$$

$$CNOT|\psi_0\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|11\rangle + \delta|10\rangle$$

4 “sums” with a single physical operation!

Some math…



Bloch sphere representation
(1 qubit)

- **Pure states** (those that can be written in state form): surface of the Bloch sphere
- **Mixed states** (can only be written with the density matrix formalism): inside of the Bloch sphere

SU(2):

- Three generators: σ_x, σ_y and σ_z (the Pauli matrices)

- Isomorphic to SO(3), meaning the evolution of the qubit states can be represented with rotations on the Bloch sphere

$$R_x(\theta) = e^{i\,\theta\sigma_x}, \text{ etc}$$

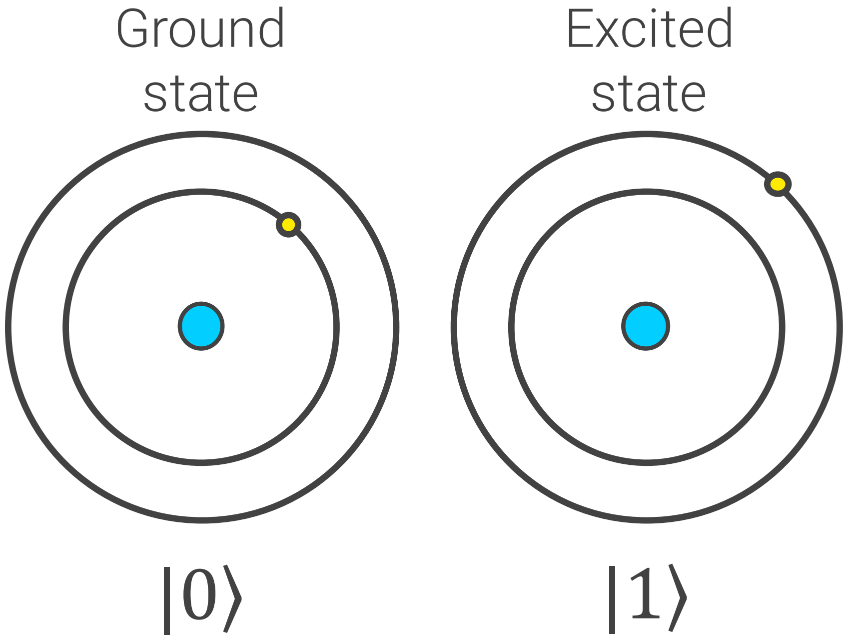
In general (n-qubits) quantum logic operations are represented by Hermitian matrices (unitary complex matrices) [SU(2^n) group]

Qubit states have $2^n - 2$ degrees of freedom (you need this number of complex variables to fully represent an arbitrary state).

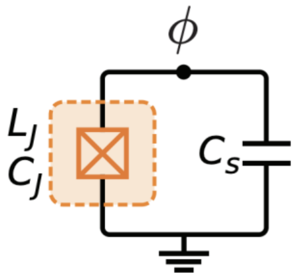
How does it work

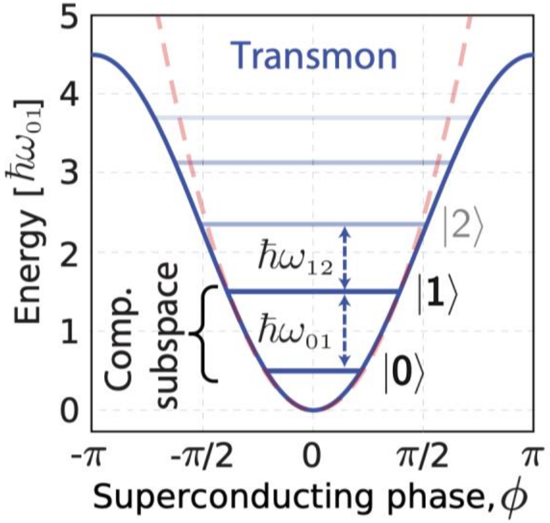
Qubit: physical system that 1) is quantum and 2) have two well-defined states

Example: atomic orbitals



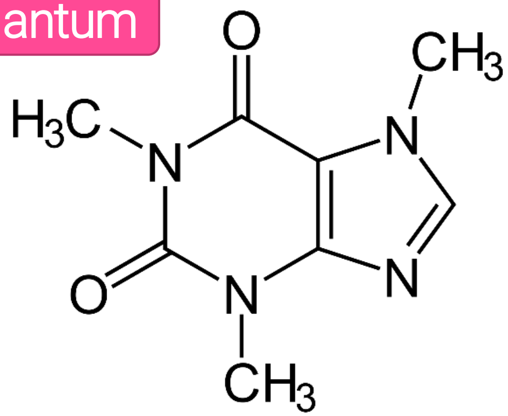
Example: superconducting circuit (transmon qubit)



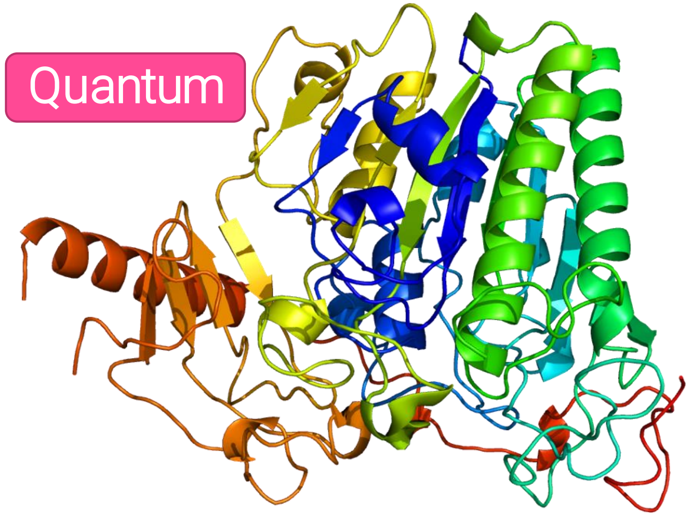


Why do we need a quantum computer

Quantum

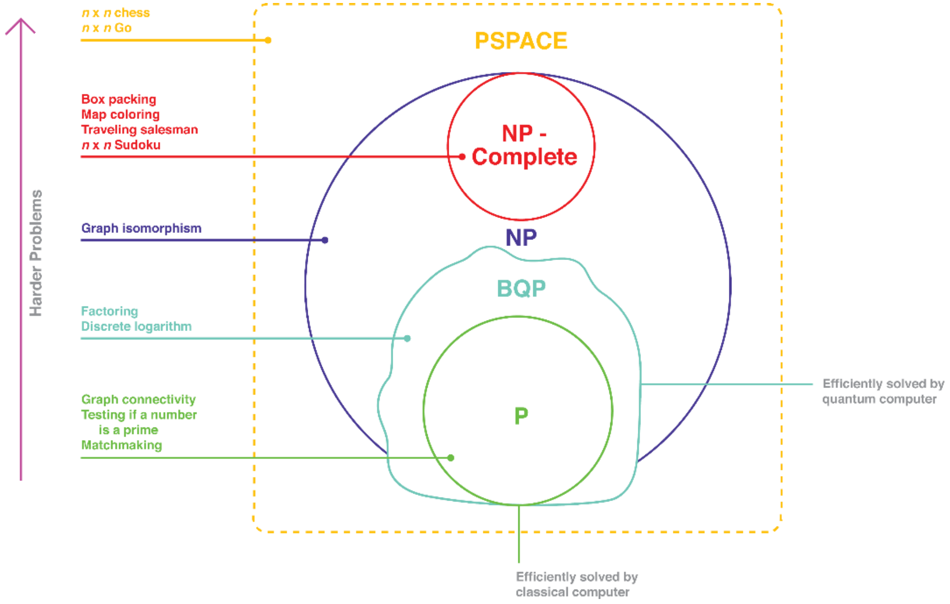


Quantum



Not Quantum

Quantum

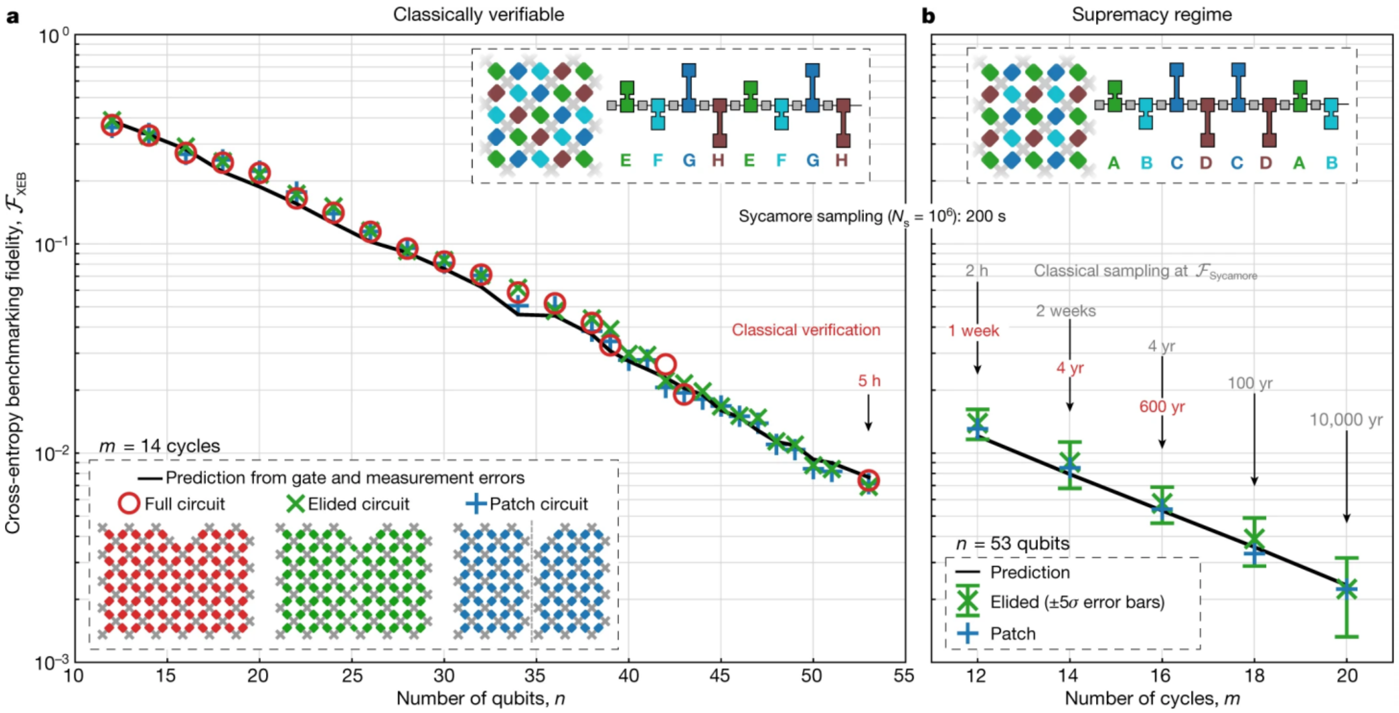


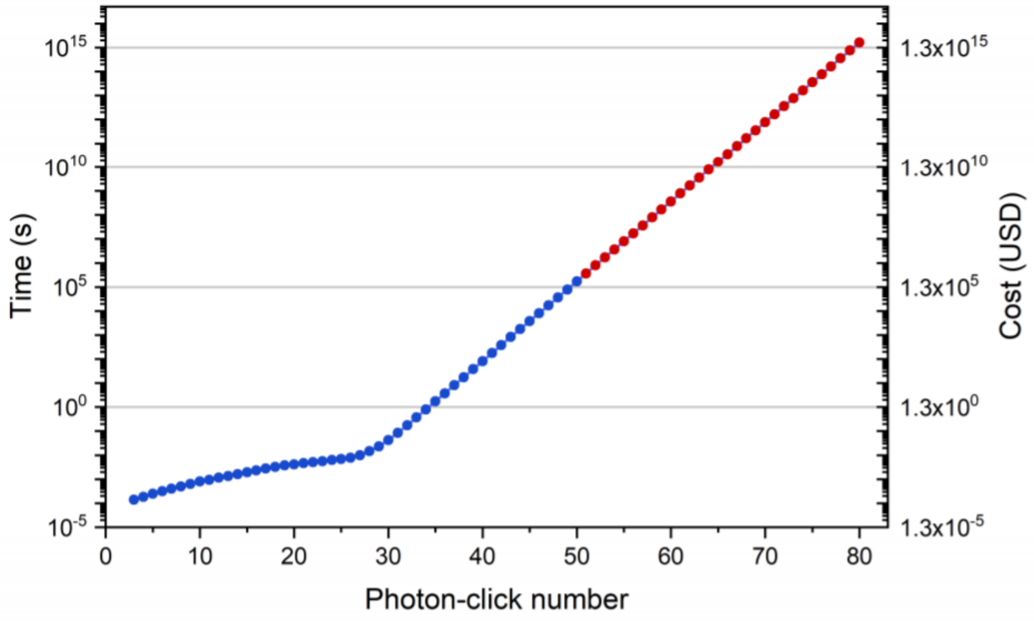
Powerful, but Not Quantum

MareNostrum supercomputer (BSC)

Why do we need a quantum computer

Less time… and less energy!





"Quantum supremacy using a programmable superconducting processor", Nature **574**, 505–510(2019)

"Quantum computational advantage using photons", Science eabe8770 (2020)

Who is building a quantum computer (industry)

Google
IBM
Microsoft
rigetti
XANADU
Alibaba Group 阿里巴巴集团
Honeywell
D:Wave The Quantum Computing Company™
ORCA Computing
QILIMANJARO
PsiQuantum
IQM
SILICON QUANTUM COMPUTING
AQT
OXFORD IONICS
IONQ

Enablers

BlueFors
Q-CTRL
qci Quantum Circuits, Inc
ANYON Computing with Manufactured Quanta™
EeroQ

Software

ZAPATA
CAMBRIDGE QUANTUM COMPUTING LIMITED
HORIZON QUANTUM COMPUTING
MULTIVERSE COMPUTING
ENTROPICA LABS
RIVERLANE
1QBit
PHASECRAFT
QCWARE

... and many more!

Where are they being constructed

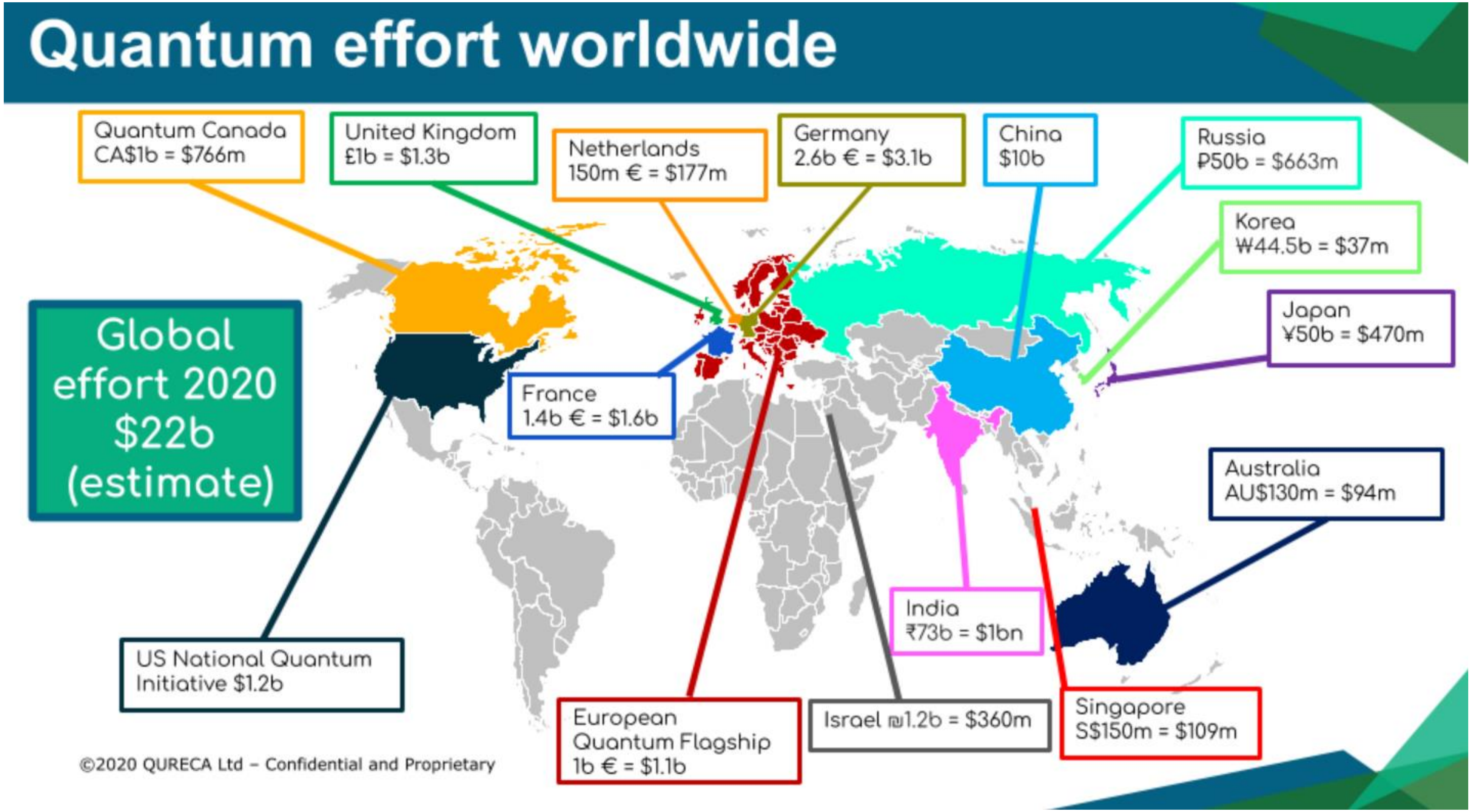


Image: "Overview on quantum initiatives worldwide", Araceli Venegas-Gomez (Qureca Ltd)

Quantum computing in the NISQ era

Quantum Computing in the NISQ era and beyond

John Preskill

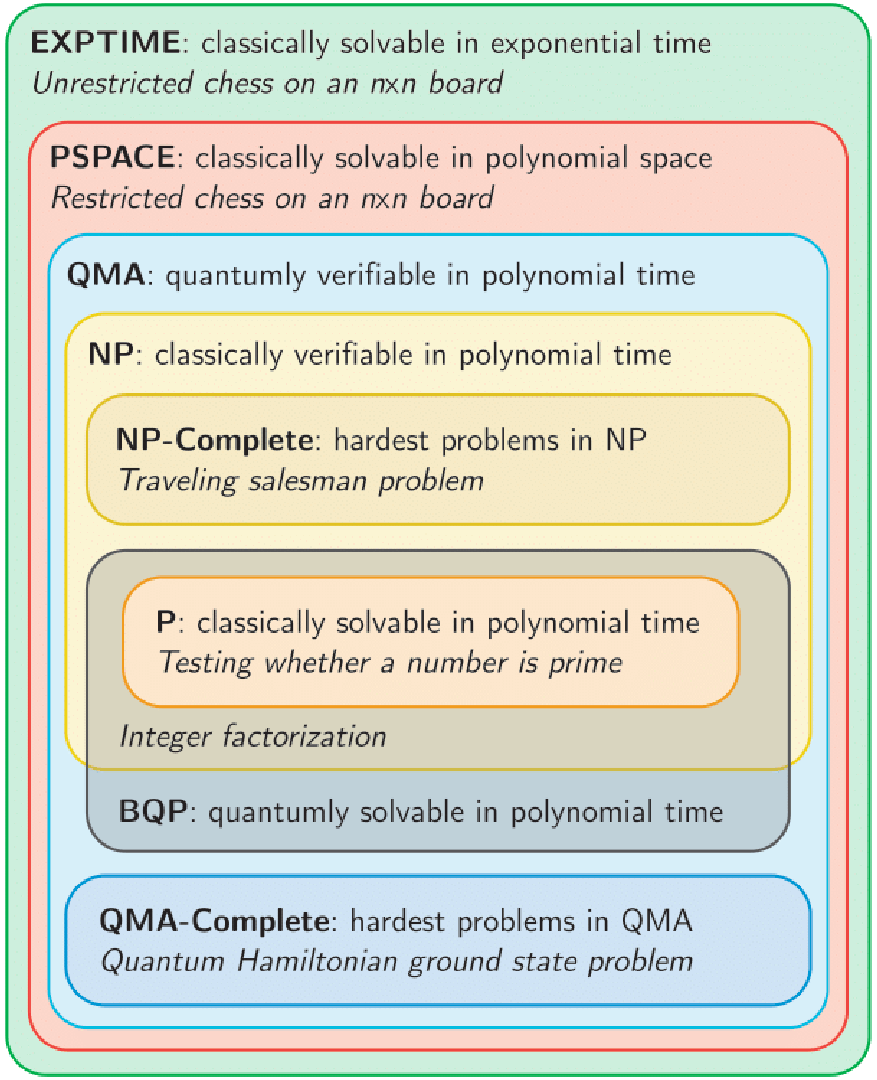
Quantum 2, 79 (2018)

Noisy intermediate-scale quantum (NISQ) algorithms

Kishor Bharti,[1,*] Alba Cervera-Lierta,[2,3,*] Thi Ha Kyaw,[2,3,*] Tobias Haug,[4] Sumner Alperin-Lea,[3] Abhinav Anand,[3] Matthias Degroote,[2,3,5] Hermanni Heimonen,[1] Jakob S. Kottmann,[2,3] Tim Menke,[6,7,8] Wai-Keong Mok,[1] Sukin Sim,[9] Leong-Chuan Kwek,[1,10,11,†] and Alán Aspuru-Guzik[2,3,12,13,‡]

arXiv:2101.08448

The power of quantum



Why do we need a quantum computer?

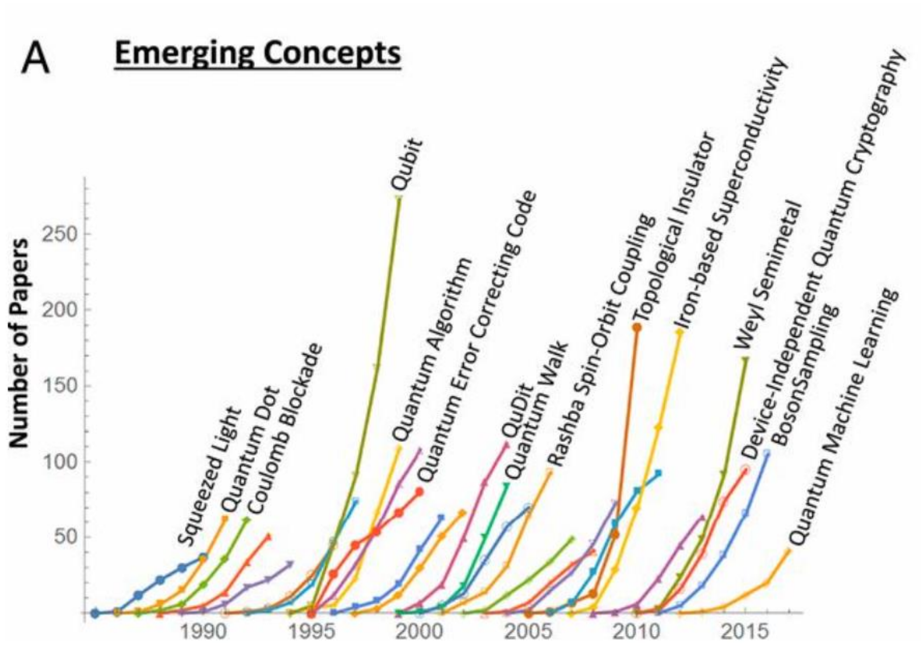
- Quantum simulation
- Solve problems beyond P and BPP

Quantum computers are powerful but not limitless

Which problems are BQP?

Approximate solutions to NP problems?

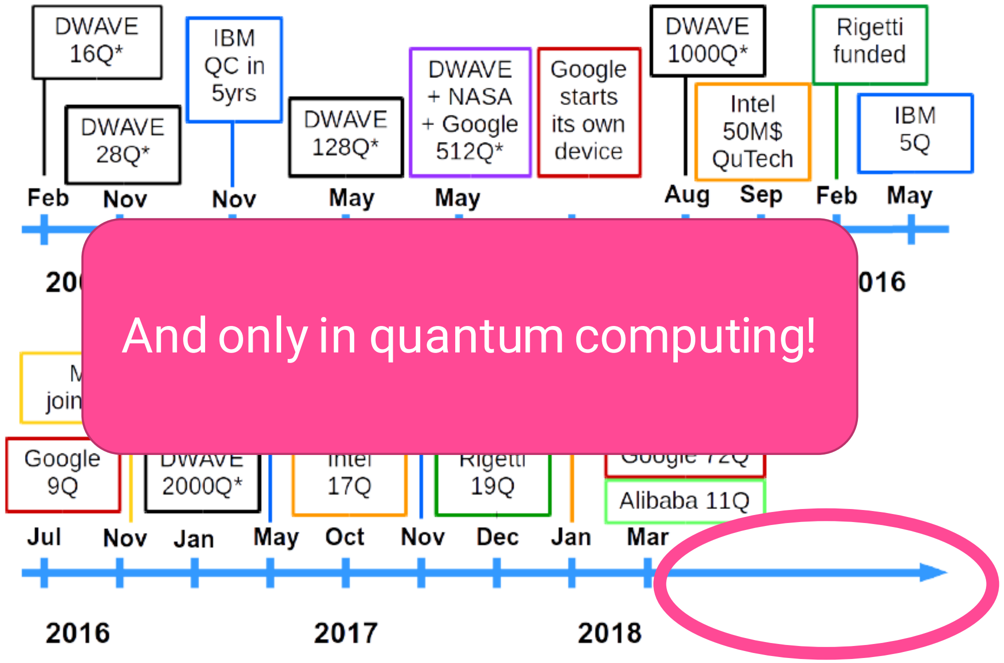
The power of quantum



qubit, April '95, Schumacher, *Quantum coding*. PRA 51, 2738–2747

Predicting research trends with semantic and neural networks with an application in quantum physics, M. Krenn, A. Zeilinger, PNAS 117 (4) 1910-1916 (2020)

From a popular science talk in 2018:



And only in quantum computing!

Trapped ions companies: IonQ, Honeywell, Alpine QT

Quantum supremacy using a programmable superconducting processor, Google AI, Nature 574, 505(2019).

Quantum computational advantage using photons, USTC (Chao-Yang Lu, Jian-Wei Pan's group), Science 370, 1460 (2020).

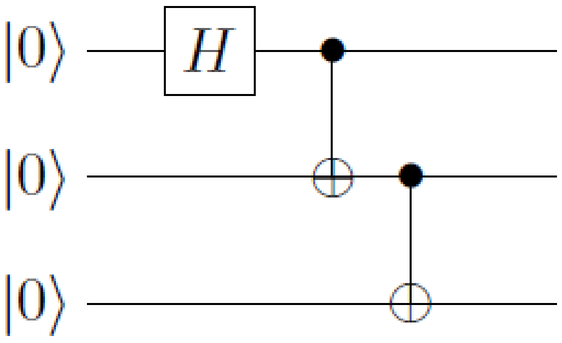
Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

Quantum Computing Paradigms

- Measurment-based (one-way)
- Quantum annealing
- Adiabatic Quantum Computing
- Quantum simulators
- Boson Sampling
- Digital-analog quantum computation
- Gate-based quantum computation

Gate-based Quantum Computing

Name	Definition	Mathematically	Diagramatically	Experimentally
Qubits	2-level quantum systems	Two-dimensional complex vector	Lines	Photons, superconducting circuits, trapped ions, …
Gates	Interactions between qubits that generate superposition and entanglement in a controlable way	SU(2^n) matrices where n is the number of qubits involved in the operation	Boxes that specify the gate and some vertical symbols that represent particular entangling gates (CNOT and SWAP)	Laser pulses (ions, photons), optical devices (SPDC, PS, …) microwave pulses
Measurement	Interaction with the individual qubits that forces its collapse to one of the two levels	Projector over the computational basis state	Box with a “meter” symbol	Coupling with a cavity, photon detectors,…



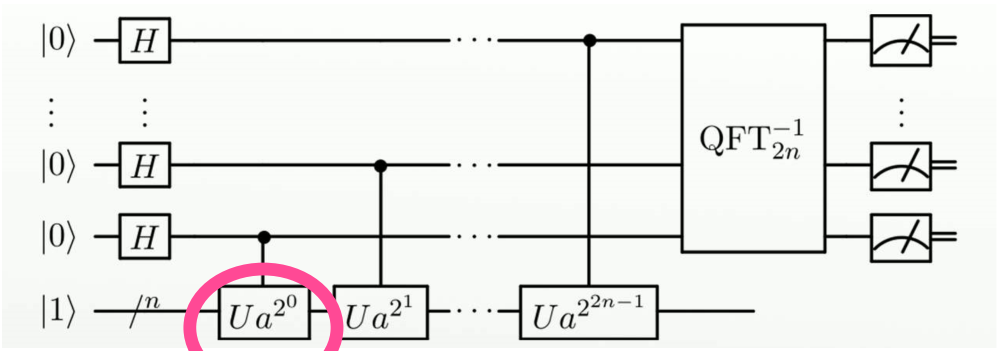
$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

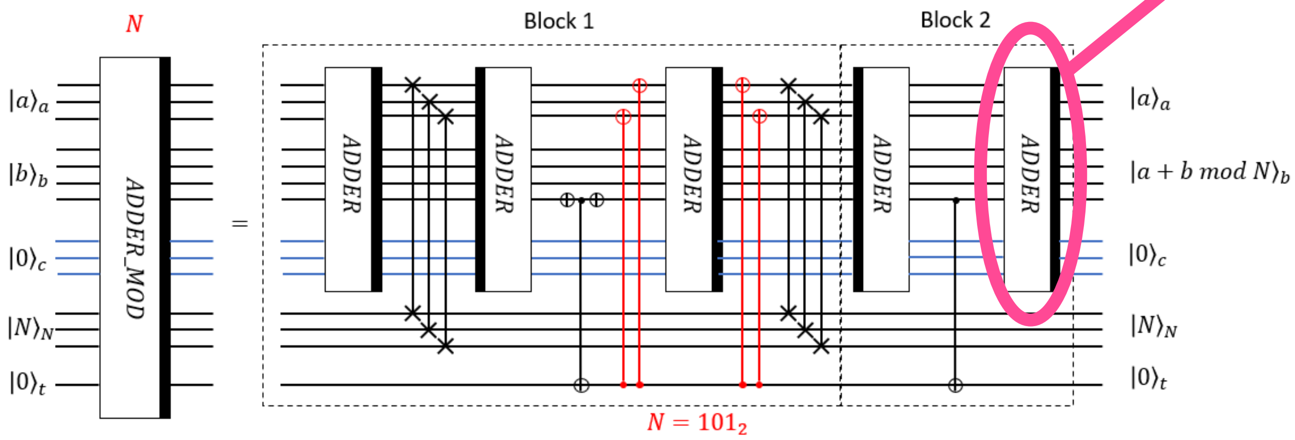
$$\mathrm{CNOT} = \mathrm{CX} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

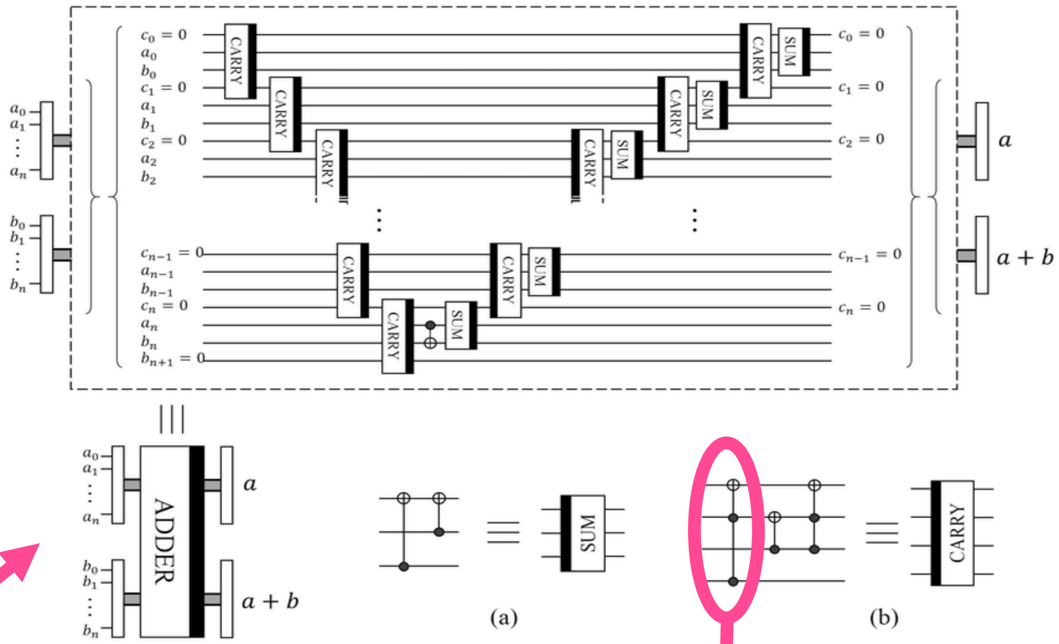
This quantum circuit generates the GHZ state

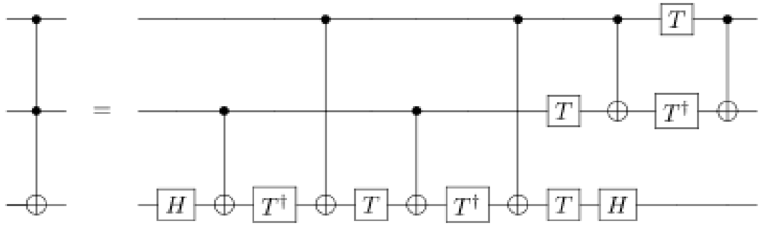
From theory to experiment

Integer factorization (Shor's) algorithm





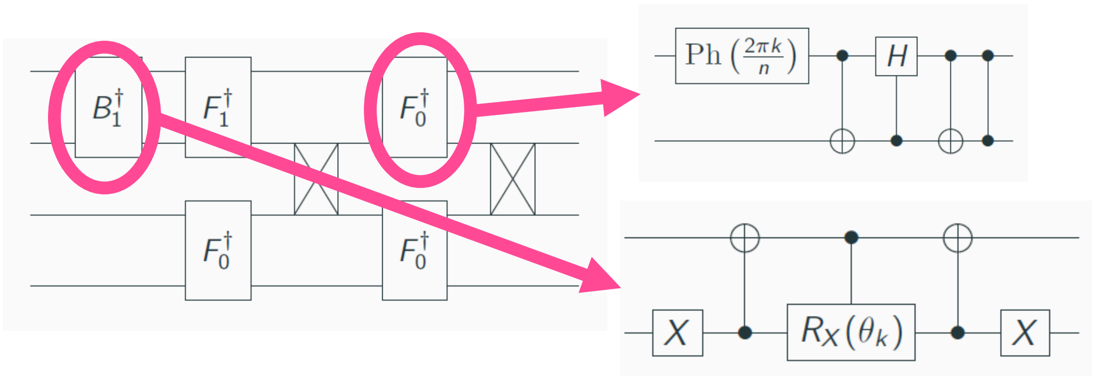




Shor, P.W. , IEEE Comput. Soc. Press: 124 (1994), A. Ekert, R. Jozsa, Rev. Mod. Phys. 68, 733 (1996)

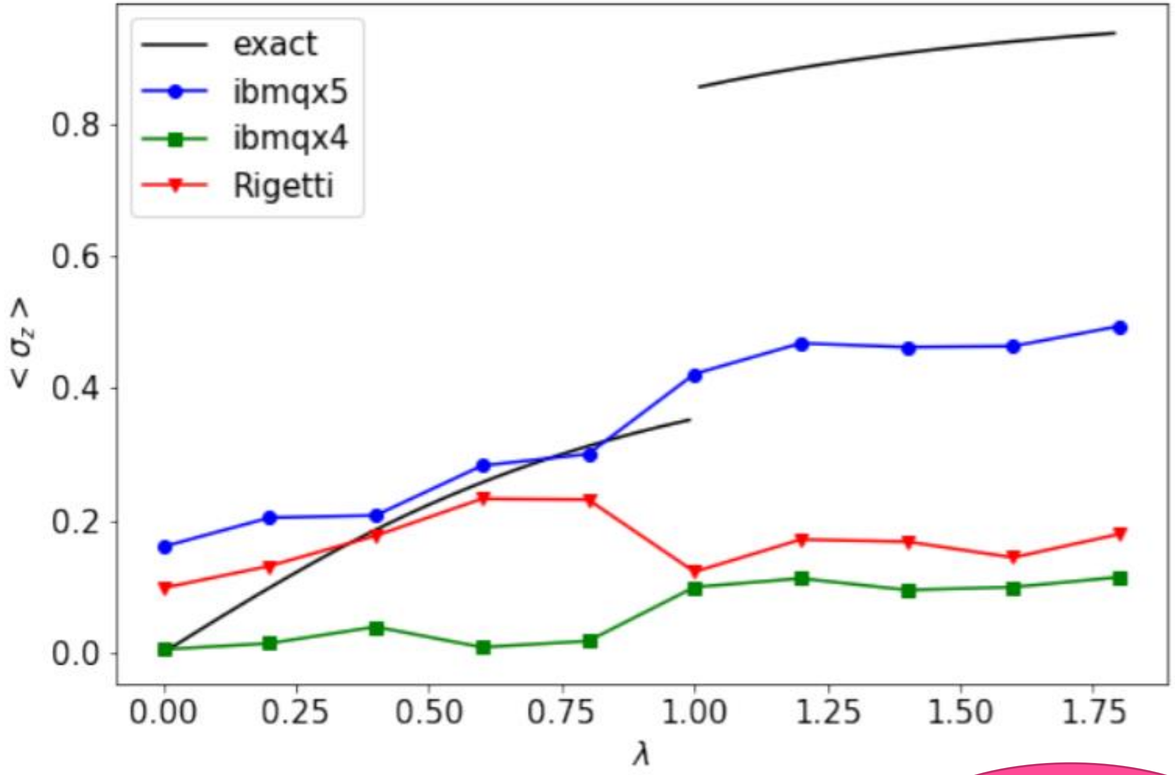
From theory to experiment

Example: $n = 4$ Ising model simulation.



~35 gates circuit depth
~500 ns entangling gates
~100 ns single-qubit

<17500 ns = 17.5 μs



Qubits coherence time → ~50 μs

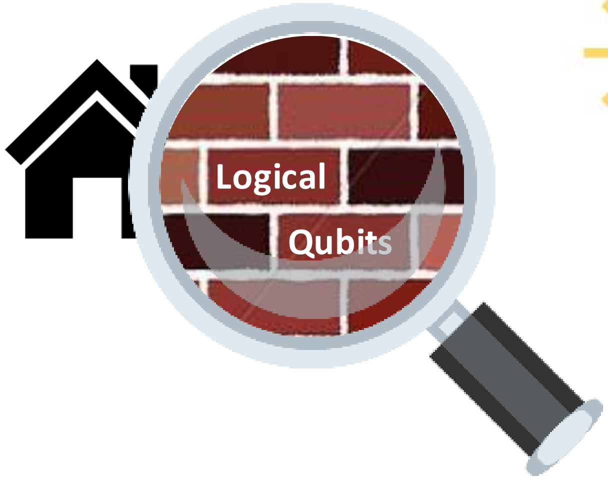
March-May 2018

Errors coming from readout, cross-talk, relaxation, … are relevant and difficult to track

NISQ vs Fault-Tolerant

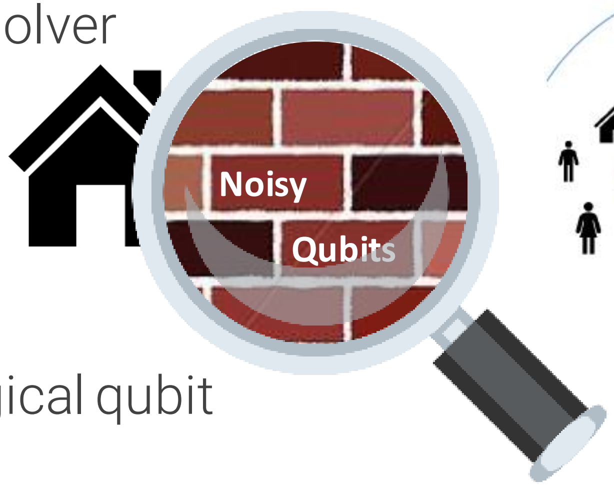
Who lives in the Fortress?

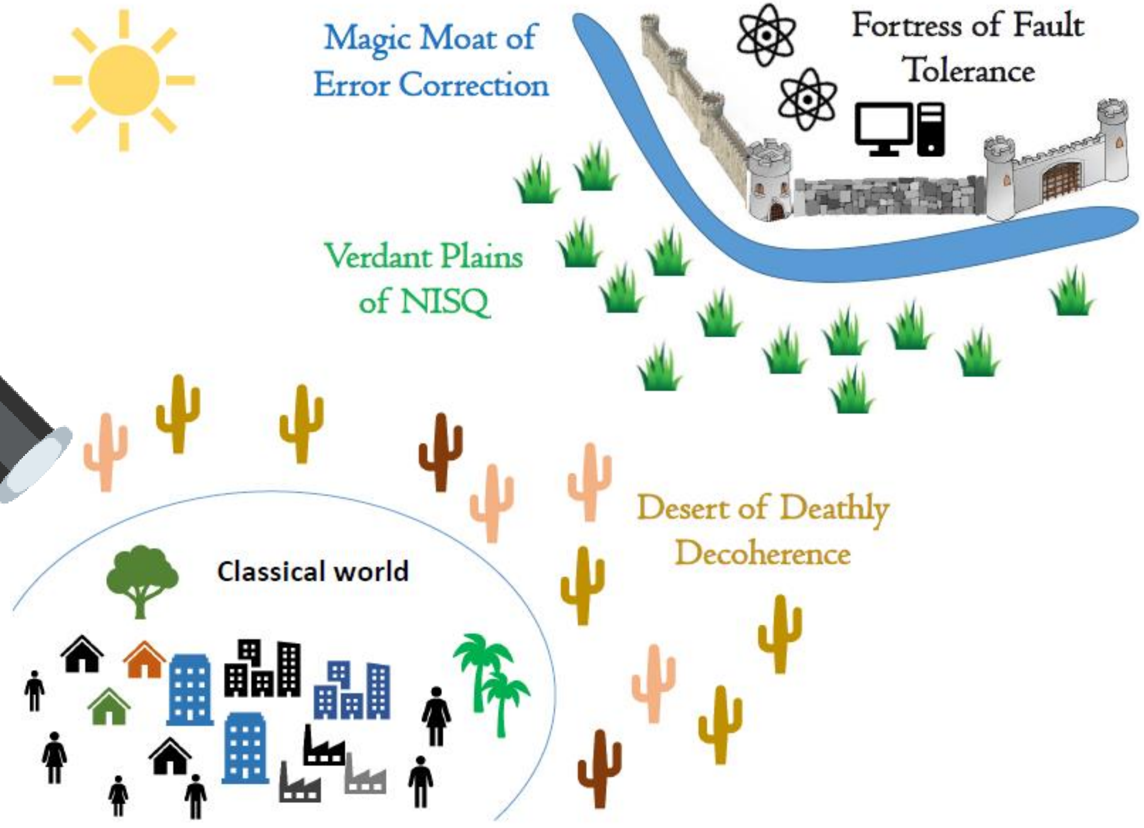
- Factorization algorithm
- Grover search algorithm
- ...



Who lives in the Plains?

- Variational Quantum Eigensolver
- QAOA
- ...

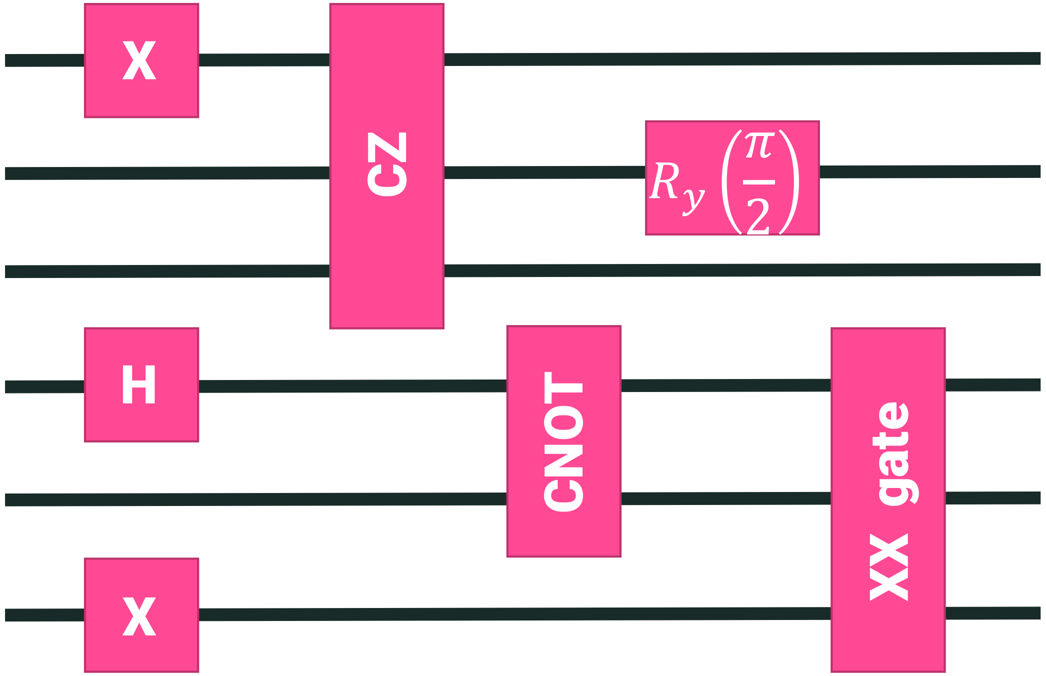




~1000 noisy qubits/logical qubit

Image: "Quantum computing: near- and far-term opportunities", Ewan Munro, Medium @quantum_wa

NISQ circuits

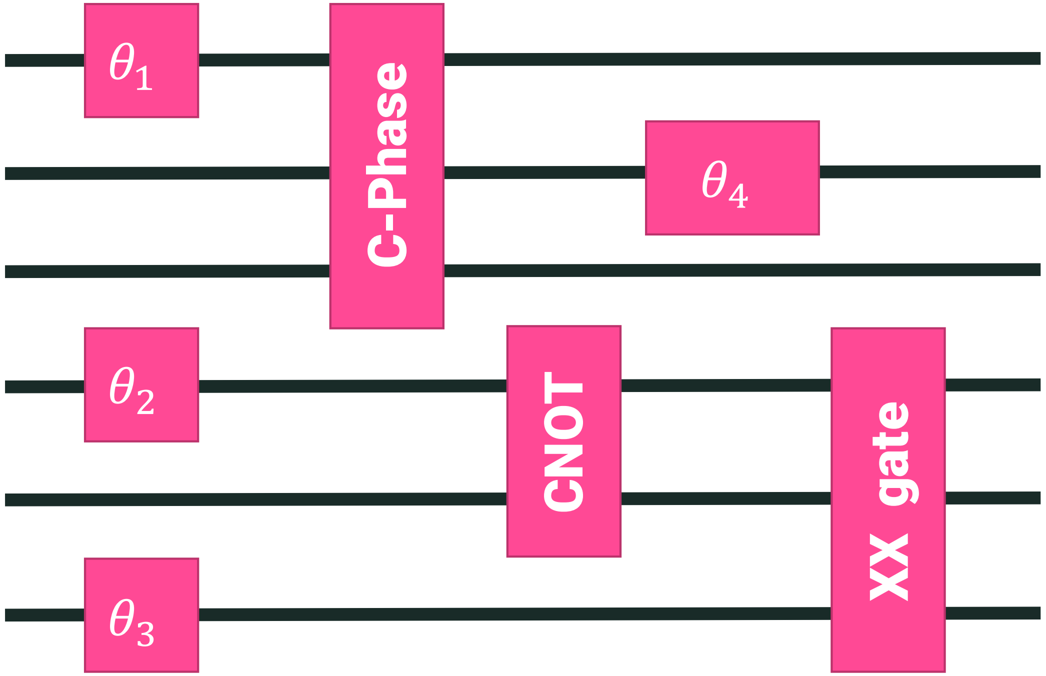


Imperfect gate operations.

We cannot run:

- Algorithms that require perfect control (e.g. Grover, QFT, …)
- Circuits that require many gates

NISQ circuits



Imperfect gate operations.

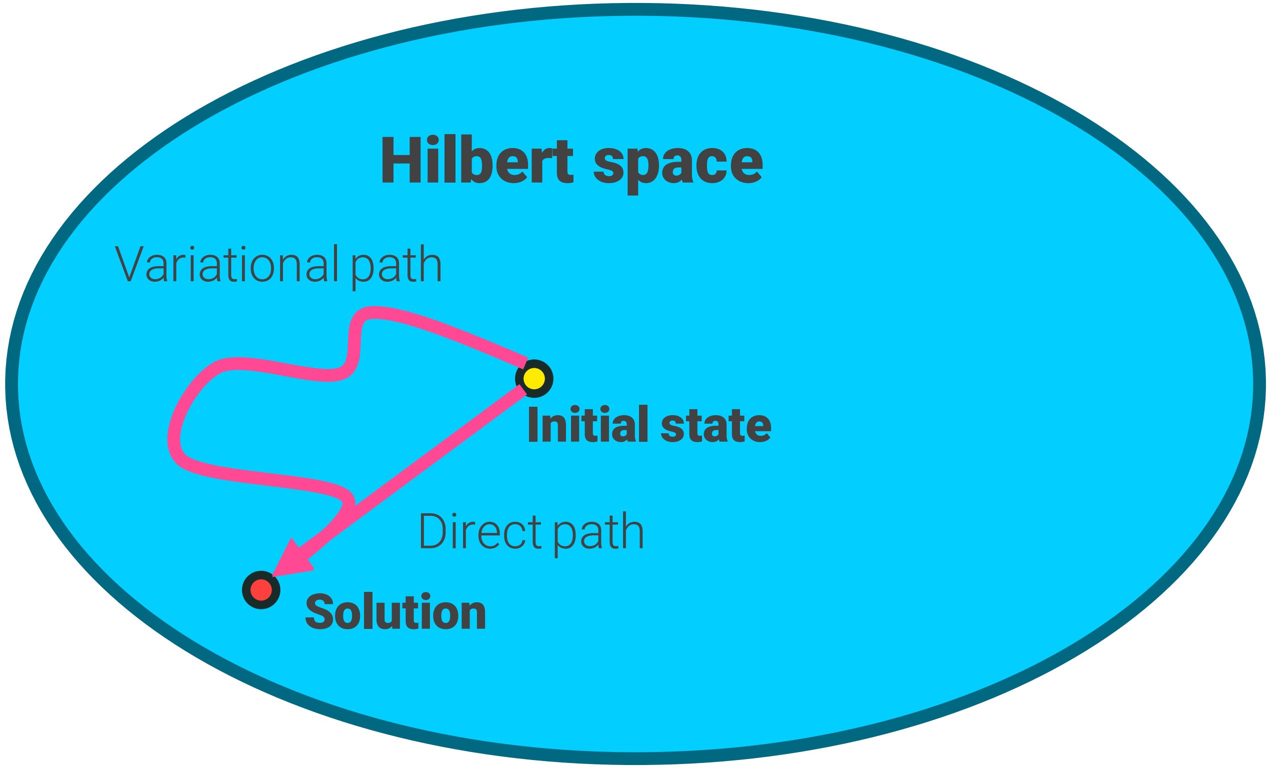
We can run:

- Algorithms that do not require particular gates

- Circuits that require a few gates

Can we design algorithm resistant to these imperfect operations?

Clever ways to explore the Hilbert space



We can not apply exact algorithms…

… but we can explore the Hilbert space in other ways.

We can use our quantum computer as a machine that generates variational states and find a way to converge towards the solution

Noisy Intermediate-Scale Quantum

Why is QC hard experimentally?

- Qubits have to interact strongly (by means of the quantum logic gates)…
- …but not with the environment…
- …except if we want to measure them.

What is the state-of-the-art in digital quantum computing?

- ~50 qubit devices
- Error rates of ~10^-3
- No Quantum Error Correction (QEC)

Noisy Intermediate-Scale Quantum (NISQ) computing

- 50-100 qubits
- Low error rates
- No QEC

What can we do in NISQ?

- Good trial field to study physics
- Possible applications?
- A step in the path towards Fault Tolerant QC

Noisy Intermediate Scale Quantum computation

Different qubits architectures

A few qubits (~100)

Noise

Decoherence

Classical optimizers

Quantum McGyver carrying a quantum advantage experiment

Something interesting:

Undestanding quantum physics

Something useful:

Quantum advantage

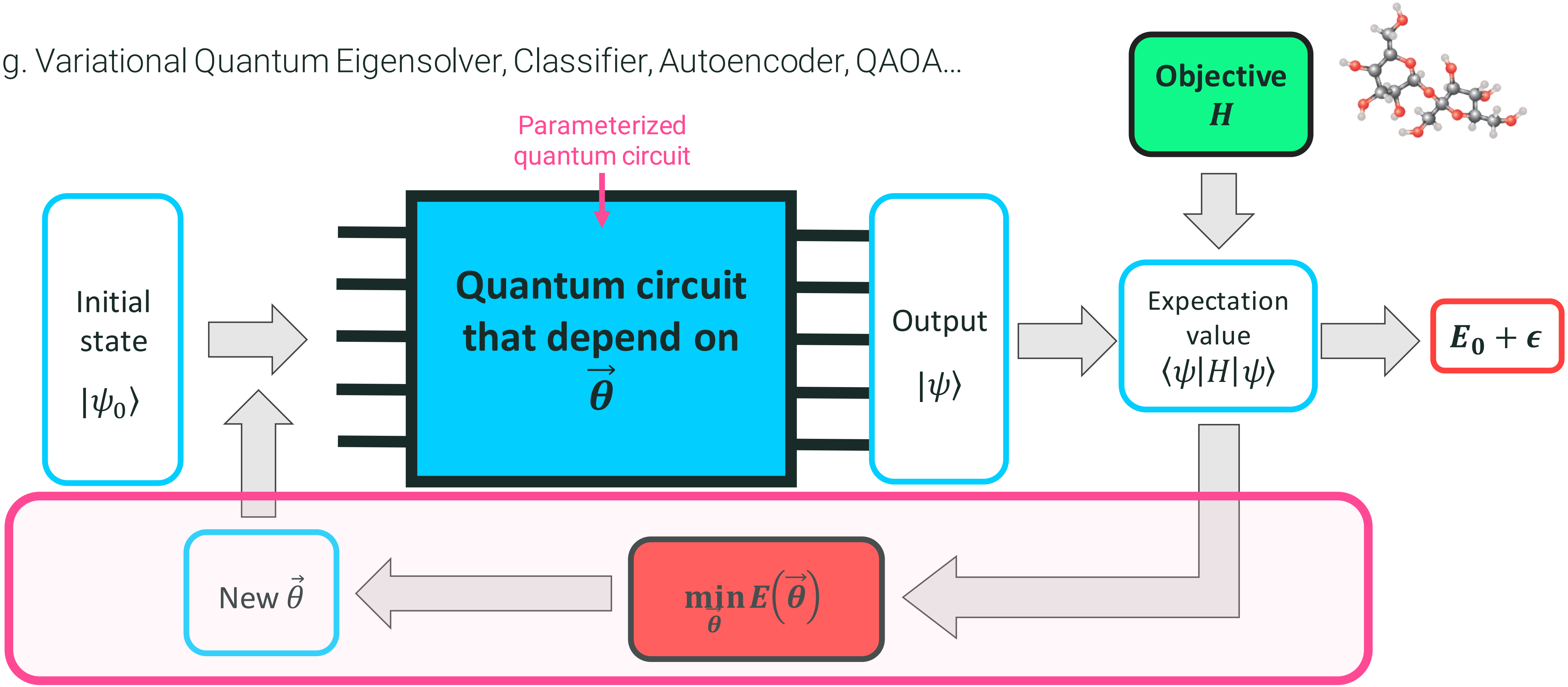
Variational Quantum Algorithms

Variational Quantum Algorithms is one of the most used NISQ paradigms, but it is not the only one

The parents of VQA are the Variational Quantum Eigensolver (VQE) and the Quantum Approximate Optimization Algorithm (QAOA).

Variational Quantum Algorithms

e.g. Variational Quantum Eigensolver, Classifier, Autoencoder, QAOA…



Classical optimization

Variational principle: $\mathrm{E} = \langle\psi|H|\psi\rangle \geq E_0$

Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

Parameterized quantum circuits

Our Parameterized Quantum Circuit (PQC)

$$\boxed{|\Phi(\theta)\rangle = U(\theta)|0\rangle}$$

$$E_0 = \min_{\theta}\langle\Phi(\theta)|H|\Phi(\theta)\rangle = \min_{\theta}\langle 0|U^{\dagger}(\theta)HU(\theta)|0\rangle$$

Assumptions:

1. There exist a set of parameters that approximates the ground state $\exists\ \theta^* \mid |\Phi(\theta^*)\rangle \simeq |gs\rangle$
2. Our PQC can represent that solution
3. We can converge towards the solution (we do not get trapped in local minima)
4. The PQC can be run on a NISQ computer

Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

Parameterized quantum circuits

How can we design $U(\theta)$?

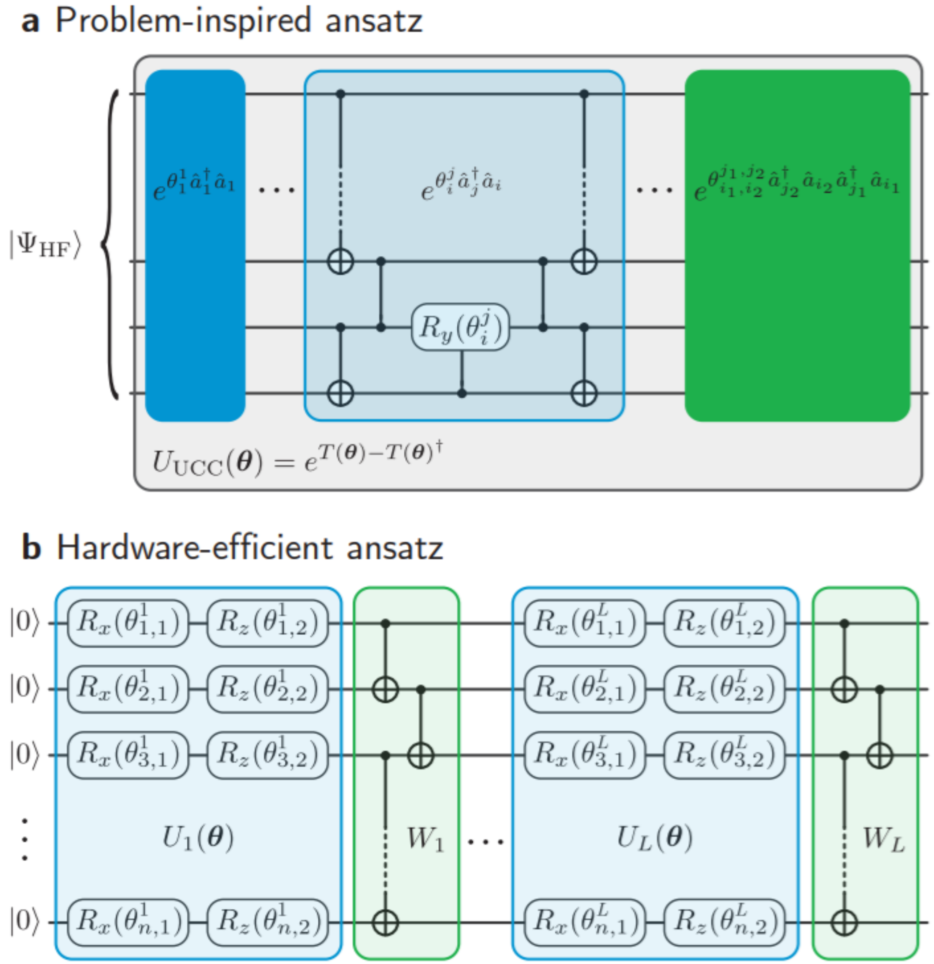
Two strategies:

1. Problem-inspired PQC ansatz.

   a) Approximation to the solution by construction.
   b) High-circuit depth/# gates in general (not always hardware-friendly)

2. Hardware-efficient ansatz.

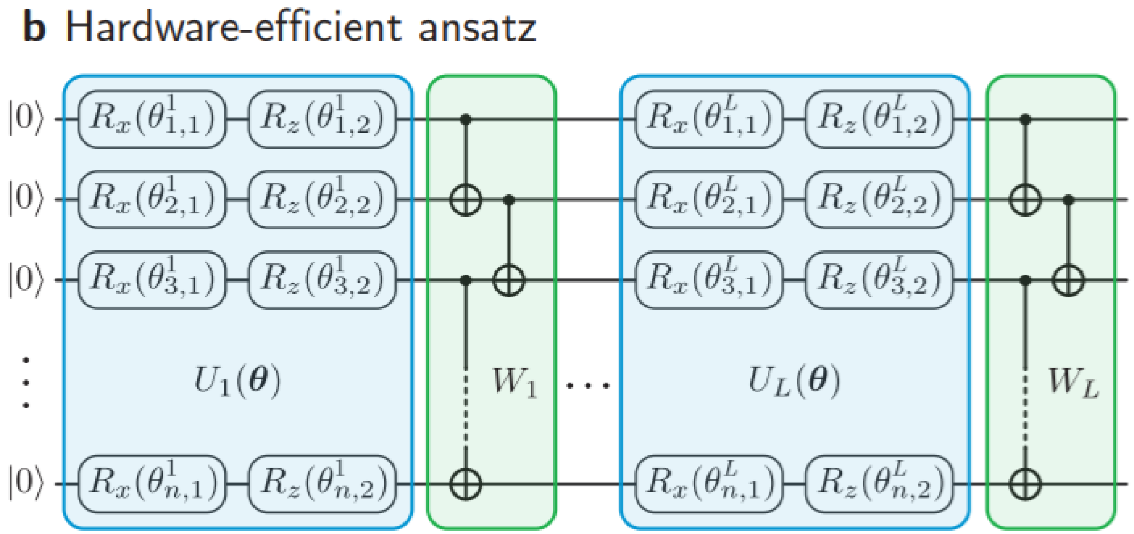
   a) Heuristic ansatz
   b) Low circuit depth/# gates in general- (hardware-friendly)



Parameterized quantum circuits

Problem-inspired ansatz: UCCSD, QAOA, etc (see VQE section ahead).

Hardware-efficient ansatz:

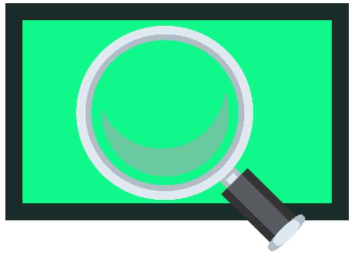


- Low circuit depth
- Hardware-friendly gates (native gates that can be implemented experimentally)
- Respectful with qubit connectivity
- Useful for general problems (no problem-inspired ansatzes).

Example:
Layers of subcircuits.
Each layer: single-qubit gates + entangling gates

Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

Objective function



It encodes the problem in a quantum operator, e.g. a Hamiltonian

$$\langle H \rangle_{\mathcal{U}(\boldsymbol{\theta})} \equiv \langle 0 | \mathcal{U}^{\dagger}(\boldsymbol{\theta}) \, H \mathcal{U}(\boldsymbol{\theta}) | 0 \rangle$$

The objective is decomposed into Pauli strings which expectation value can be measured with the quantum computer.

$$H = \sum_{k=1}^{M} c_k \hat{P}_k \longrightarrow \langle H \rangle_{\mathcal{U}} = \sum_{k=1}^{M} c_k \langle \hat{P}_k \rangle_{\mathcal{U}}$$

An objective can also be the fidelity w.r.t. a particular target state that we are trying to match.

$$F\left(\Psi, \Psi_{\mathcal{U}(\boldsymbol{\theta})}\right) \equiv |\langle \Psi | \Psi_{\mathcal{U}(\boldsymbol{\theta})} \rangle|^2$$

Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

Measurement

We need to find a way to extract information from our quantum computer.

In general, quantum devices project in a particular basis, normally the z-basis.

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\sigma_z|0\rangle = +1|0\rangle$$
$$\sigma_z|1\rangle = -1|1\rangle$$

$$|0\rangle \equiv \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle \equiv \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

This means we only measure the eigenvalues of the σ_z operator, namely the "0"s and the "1"s

Probability of obtaining |0>

$$\langle \hat{\sigma}_z \rangle = 2p_0 - 1$$

In other basis, we need to rotate the state to that particular basis first

$$\hat{\sigma}_x = R_y^\dagger\left(\frac{\pi}{2}\right)\hat{\sigma}_z R_y\left(\frac{\pi}{2}\right) = H_\mathrm{d}\hat{\sigma}_z H_\mathrm{d},$$
$$\hat{\sigma}_y = R_x^\dagger\left(\frac{\pi}{2}\right)\hat{\sigma}_z R_x\left(\frac{\pi}{2}\right) = S H_\mathrm{d}\hat{\sigma}_z H_\mathrm{d} S^\dagger$$

$$\langle \hat{\sigma}_y \rangle = \langle \Psi | \hat{\sigma}_y | \Psi \rangle = \langle \Psi | S H_\mathrm{d} \hat{\sigma}_z H_\mathrm{d} S^\dagger | \Psi \rangle$$

...and measure how many 0 we obtain as in the σ_z case

Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

Classical optimization

We need to navigate the quantum circuit parameter space, e.g. by using gradiend based methods

$$\theta_i^{(t+1)} = \theta_i^{(t)} - \eta\, \partial_i f(\boldsymbol{\theta})$$

The gradients are expectation values of the quantum circuit derivatives w.r.t. a parameter.

Example: parameter-shift rule

$$\mathcal{U}(\boldsymbol{\theta}) = V(\boldsymbol{\theta}_{\neg i})G(\theta_i)W(\boldsymbol{\theta}_{\neg i}) \qquad G = e^{-i\theta_i g}$$

Eigenvalues of g are $\pm\lambda$

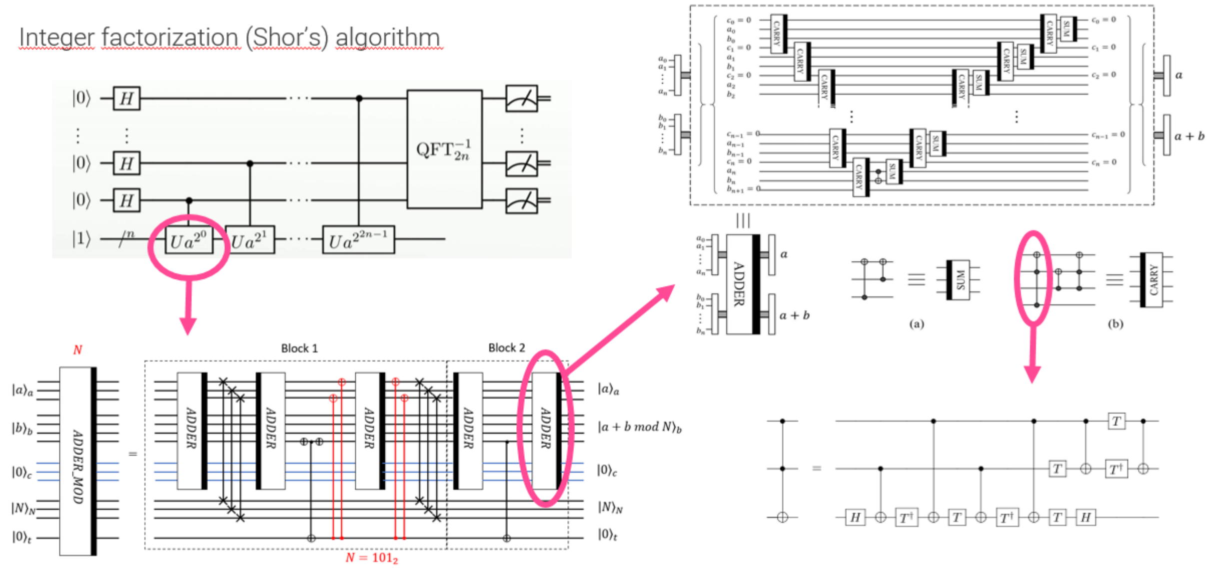
$$\partial_i \langle f(\boldsymbol{\theta})\rangle = \lambda\left(\langle f(\boldsymbol{\theta}_+)\rangle - \langle f(\boldsymbol{\theta}_-)\rangle\right) \qquad \boldsymbol{\theta}_\pm = \boldsymbol{\theta} \pm (\pi/4\lambda)\boldsymbol{e}_i$$

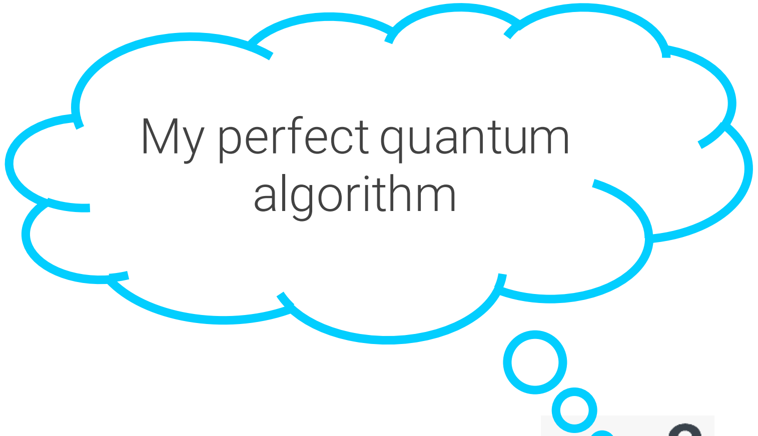
Gradient-free: genetic algorithms, reinforcement learning, ...

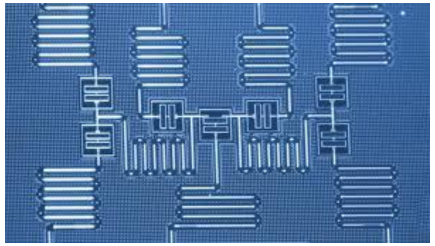
Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

Squeezing the NISQ lemon

Integer factorization (Shor's) algorithm







Quantum Error Mitigation

A set of classical post-processing techniques and active operations on hardware that allow to correct or compensate the errors from a noisy quantum computer.

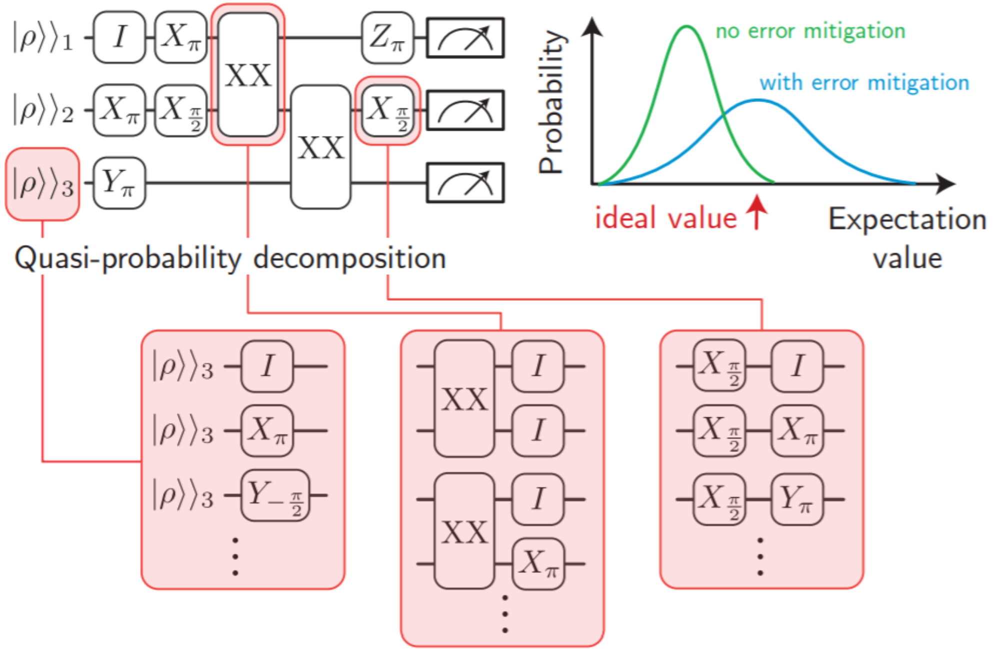
Zero-noise extrapolation

Instead of running our circuit unitary U, we run different circuits $U(UU^{\dagger})^{n}$ (increasingly noisy). Extrapolate the result for zero-noise U

Stabilizer based approach

relies on the information associated with conserved quantities such as spin and particle number conserving ansatz. If any change in such quantities is detected, one can pinpoint an error in the circuit.

Probabilistic error cancellation



Quantum Error Mitigation

Quantum Optimal Control strategies

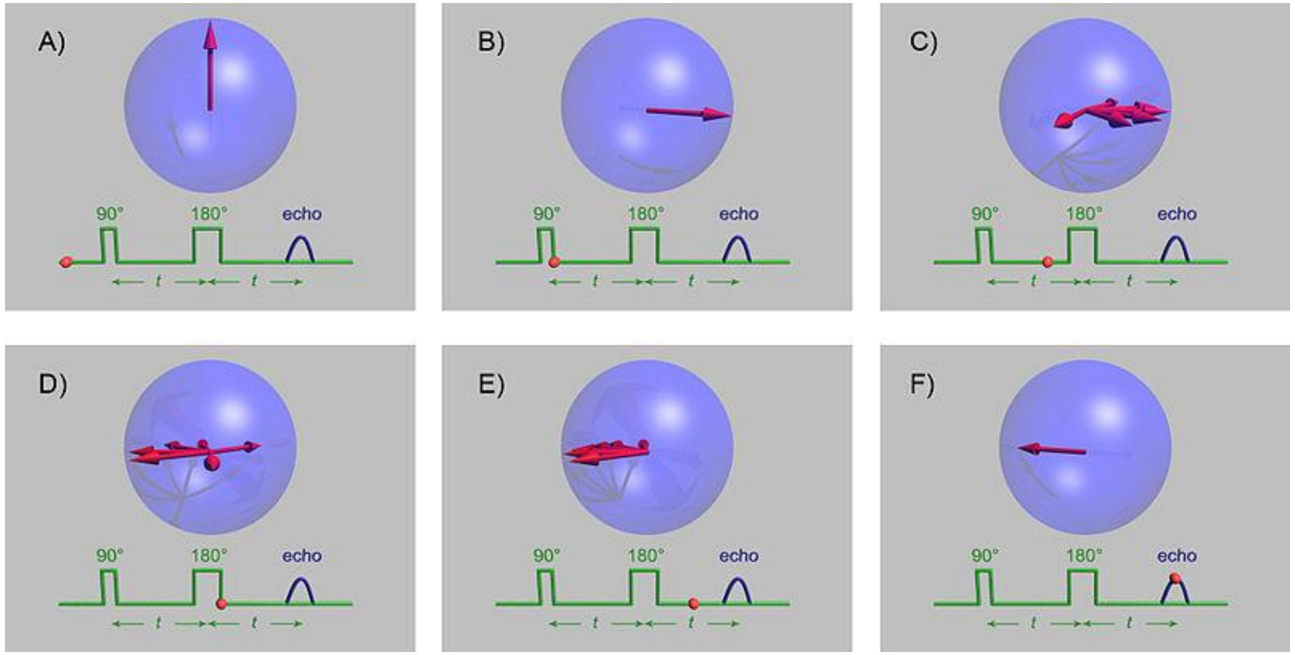
Dynamical Decoupling:

Designed to suppress decoherence via fancy pulses to the system so that it cancels the system-bath interaction to a given order in time dependent perturbation theory

Pulse shaping technique:

passive cancellation of system-bath interaction.

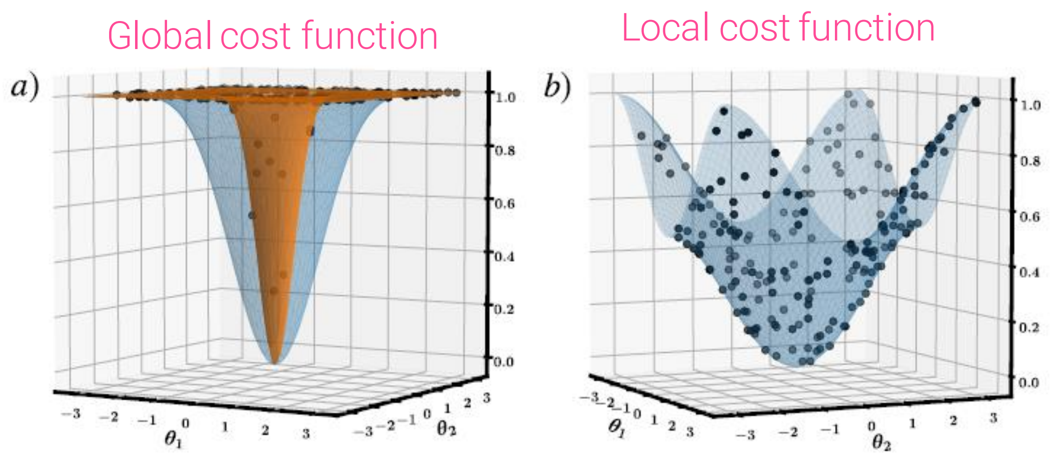
Among many others…



The *barren-plateaux* problem

Compute the gradients with the quantum circuit and use these values to run a classical minimizer, e.g. Nelder-Mead, Adam, …

With no prior knowledge about the solution, $\vec{\theta}$ parameters are initialized at random.



Consequence: *barren-plateaux*

The expected value of the gradient is zero!
The expected value of the variance is also zero!

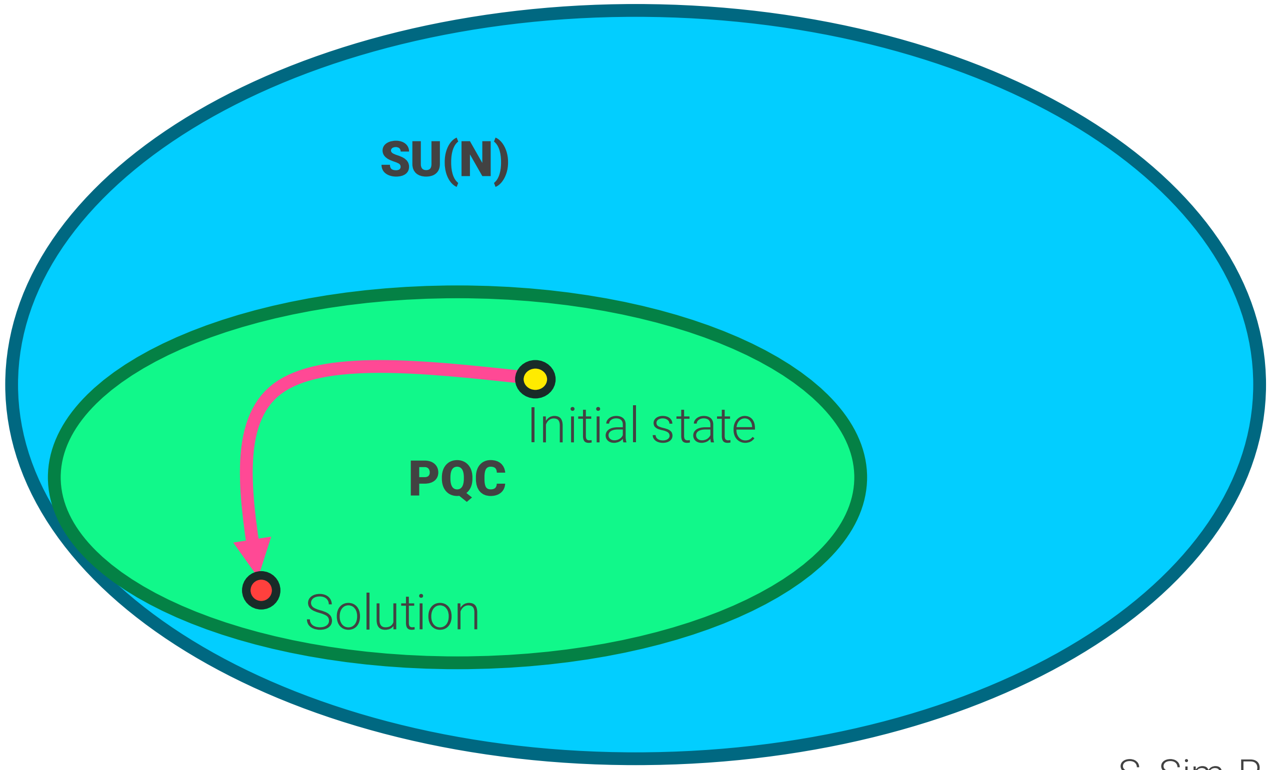
Solutions

- Use parameters close to the solution.
- Use local cost functions instead of global ones.
- Introduce correlations between parameters.

Ref.: M. Cerezo *et. al.* Nature Communications 12, 1791 (2021)

Expressibility

When setting a PQC ansatz we have to be careful to not narrow the Hilbert space accesible by the PQC so we can reach a good approximation of the solution state.

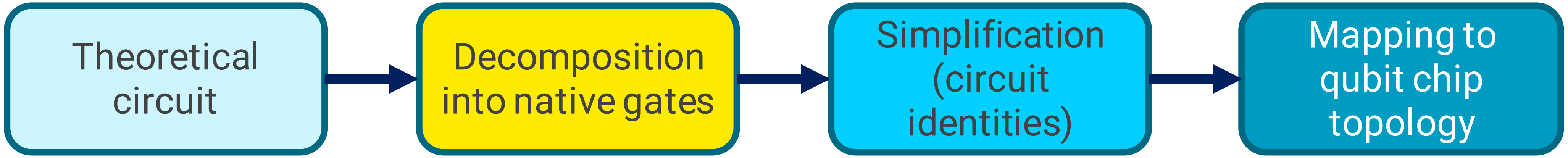


We can quantify the expressibility of a PQC by computing the distance between a Haar distribution of the states and states generated by the PQC.

$$A_U^{(t)} = \left\| \int_{\mathrm{Haar}} (|\psi\rangle\langle\psi|)^{\otimes t} d\psi - \int_{\boldsymbol{\theta}} (|\psi_{\boldsymbol{\theta}}\rangle(\langle\psi_{\boldsymbol{\theta}}|)^{\otimes t} d\psi_{\boldsymbol{\theta}} \right\|$$

S. Sim, P. D. Johnson, A. Aspuru-Guzik, Adv. Quantum Technol. 2 1900070 (2019)

Circuit compilation



Native and universal gate sets:

Solovay-Kitaev theorem: With a universal gate set we can approximate with epsilon accuracy any SU(N) with a circuit of polynomial depth.

Gottesman–Knill theorem: Circuits composed by gates from the Clifford group (Clifford circuits) can be simulated efficiently with a classical computer.

Gate sets are usually composed by Clifford gates + one non-clifford gate, e.g. {H, S, CNOT} + T

However, depending on the hardware implementation, some gates are easier to control.
e.g. CZ gates for superconducting circuits, XX gates for trapped ions.

The more native gates, the shorter and simpler the circuit

Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

Applications

Variational Quantum Eigensolver (VQE)

Resources:

- Artur Izmaylov “Quantum Chemistry on a Quantum Computer” course on Youtube
- “Quantum Chemistry in the Age of Quantum Computing”,
  Y. Cao et al, Chem. Rev., 119, 19, 10856–10915 (2019), arXiv:1812.09976 [quant-ph]

Tutorials:

- *Qiskit*: https://qiskit.org/textbook/ch-applications/vqe-molecules.html
- *Tequila*: https://github.com/aspuru-guzik-group/tequila-tutorials
- *Pennylane*: https://pennylane.ai/qml/demos/tutorial_vqe.html

Electronic structure problem

The electronic structure Hamiltonian describes the dynamics of an atom or a molecule.

In the Born-Oppenheimer approximation, it has two main terms:

$$\hat{H}_{mol} = \hat{H}_{nucl}(\vec{R}) + \hat{H}_{elec}(\vec{R}, \vec{r})$$

The wavefunction can be factorized as well

$$\psi(\vec{R}, \vec{r}) = \phi_{nucl}(\vec{R})\chi_{elec}(\vec{R}, \vec{r})$$

The part of interest for chemistry is solving the electronic one:

$$\hat{H}_{elec}\chi_{elec}(\vec{R}, \vec{r}) = E_{elec}(\vec{R})\chi_{elec}(\vec{R}, \vec{r})$$

$$\hat{H}_{elec} = -\sum_i \frac{\nabla^2_{\vec{r}_i}}{2} - \sum_{i,j} \frac{Z_i}{|\vec{R}_i - \vec{r}_j|} + \sum_{i,j>i} \frac{1}{|\vec{r}_i - \vec{r}_j|}$$

Kinetic energy electrons

Interaction electrons-nucleus

Interaction between electrons

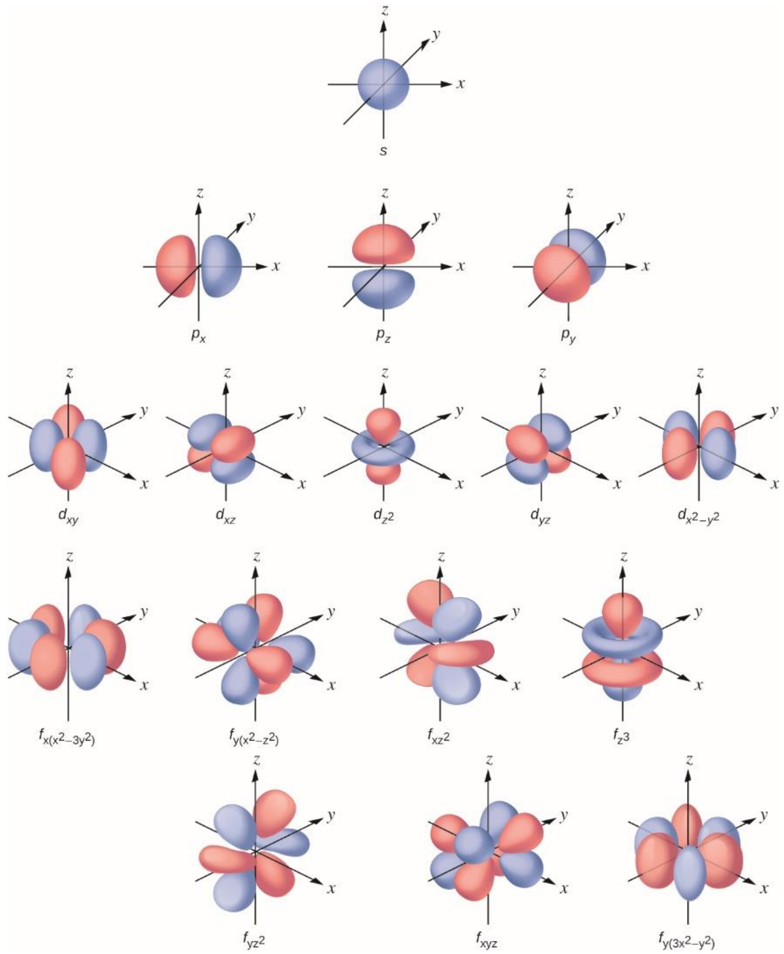
Electronic structure problem

How does the wave-function look like?

Single electrons wavefunction are the electronic orbitals.

Two-electron wavefunctions are a combinations of these orbitals in what are called Slater determinants.

Slater determinants manipulation in the first quantization might be cumbersome, so we move to the second quantization or Fock space:



$$|\psi\rangle = \sum_{orbitals} C_k \, |n_1, \ldots, n_k\rangle$$

Electronic wave-function

Occupation number of that orbital
= 0 (no orbital)
or 1 (there is an electron in that orbital)

Electronic structure problem

The electronic Hamiltonian in the second quantization becomes:

$$H_{2q} = \sum_{p,q} h_{pq} a_p^\dagger a_q + \sum_{p,q,r,s} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s$$

Single-excitations
1-electron moves from one orbital to another

Double-excitations
2-electrons move from one orbital to another

"Couple-Cluster Single-Double" model (CCSD)

Creation and annihilation operators:

$a_p^\dagger$ Adds an electron to the "p" orbital

a_q Removes an electron from the "q" orbital

CCSD on a quantum computer

$$H_{2q} = \sum_{p,q} h_{pq} a_p^\dagger a_q + \sum_{p,q,r,s} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s$$

We can not compute expectation values of the creation and annihilation operators.

We apply a unitary transformation that maps these operators into Pauli strings:

$$\langle \mathcal{H} \rangle = \sum_{i\alpha} h_\alpha^i \langle \sigma_\alpha^i \rangle + \sum_{ij\alpha\beta} h_{\alpha\beta}^{ij} \langle \sigma_\alpha^i \sigma_\beta^j \rangle + \ \ldots$$

e.g. by means of the Jordan-Wigner transformation:

$$a_k^\dagger = \left(\prod_{j=1}^{k-1} -\sigma_j^z \right) \left(\frac{\sigma_k^x + i\sigma_k^y}{2} \right)$$

We have our objective function to minimize with our VQA!

Next, what do we use as a PQC ansatz?

UCCSD ansatz

We are looking for a quantum circuit (a.k.a. unitary operation) that generates the ground state of an electronic structure Hamiltonian:

$$U_{UCCSD} \sim e^{i\,H_{UCCSD}}$$

Coefficients to be determined (with our VQA!)

$$|\Psi(\boldsymbol{\theta})\rangle = e^{T(\boldsymbol{\theta}) - T(\boldsymbol{\theta})^\dagger}\,|\Psi_{\mathrm{HF}}\rangle$$

$$T(\boldsymbol{\theta}) = T_1(\boldsymbol{\theta}) + T_2(\boldsymbol{\theta}) + \cdots$$

$$T_1(\boldsymbol{\theta}) = \sum_{\substack{i \in \mathrm{occ} \\ j \in \mathrm{virt}}} \theta_i^j \hat{a}_j^\dagger \hat{a}_i$$

$$T_2(\boldsymbol{\theta}) = \sum_{\substack{i_1, i_2 \in \mathrm{occ} \\ j_1, j_2 \in \mathrm{virt}}} \theta_{i_1, i_2}^{j_1, j_2} \hat{a}_{j_2}^\dagger \hat{a}_{i_2} \hat{a}_{j_1}^\dagger \hat{a}_{i_1}$$

Hartree-Fock approximation: single electron orbitals (first-order approximation)

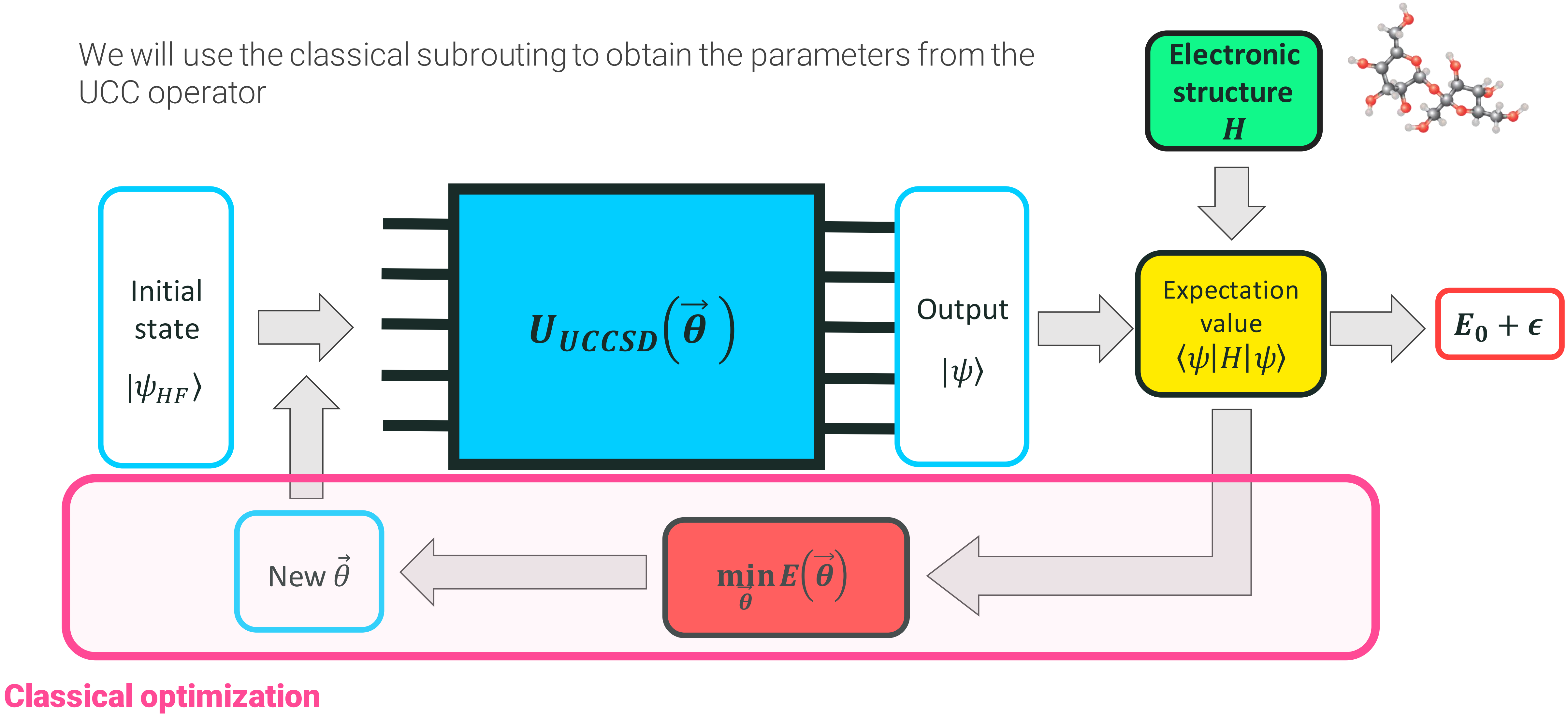
Couple-cluster operators (single, double, triple, … excitations).

We transform it into spin operators (Jordan-Wigner, etc) and use it as a unitary generators)

Remember that $e^{i\,\theta\,\sigma_x} = R_x(\theta)$ etc. From Pauli strings we can obtain the necessary quantum gates.

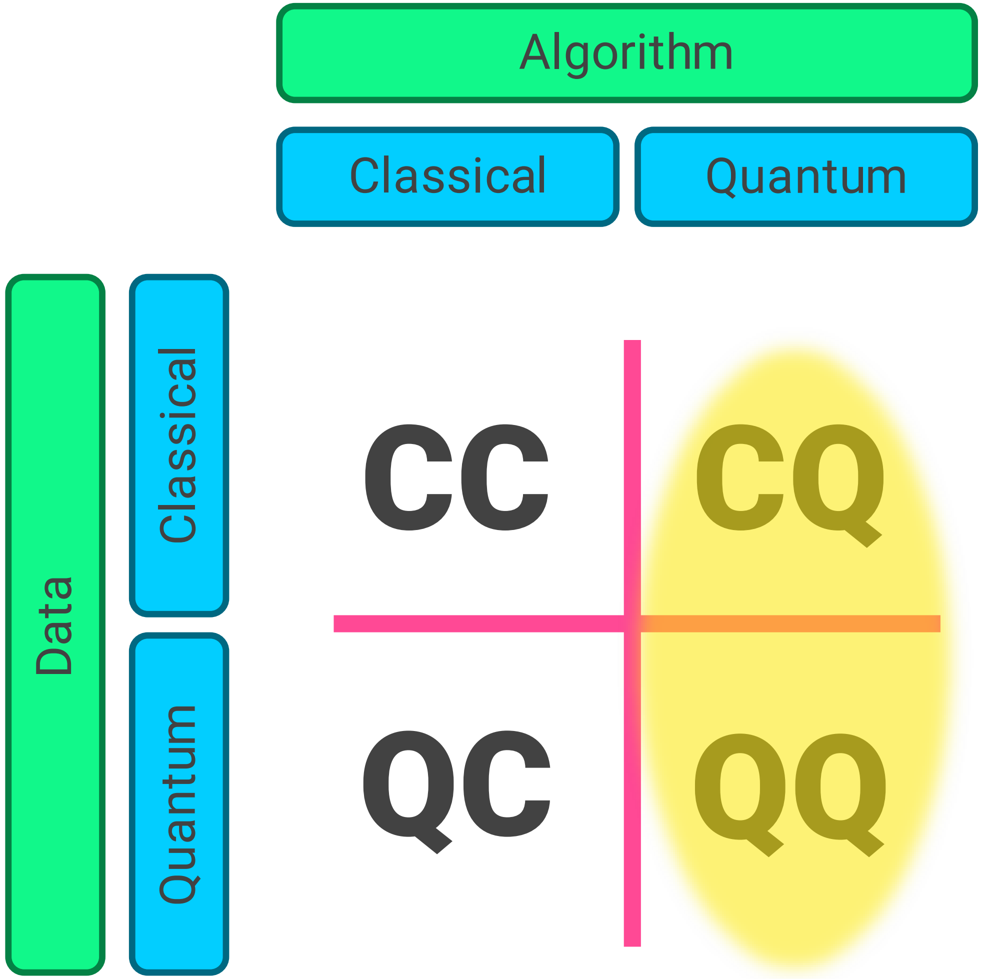
The Variational Quantum Eigensolver

We will use the classical subrouting to obtain the parameters from the UCC operator



Quantum Machine Learning

Machine Learning

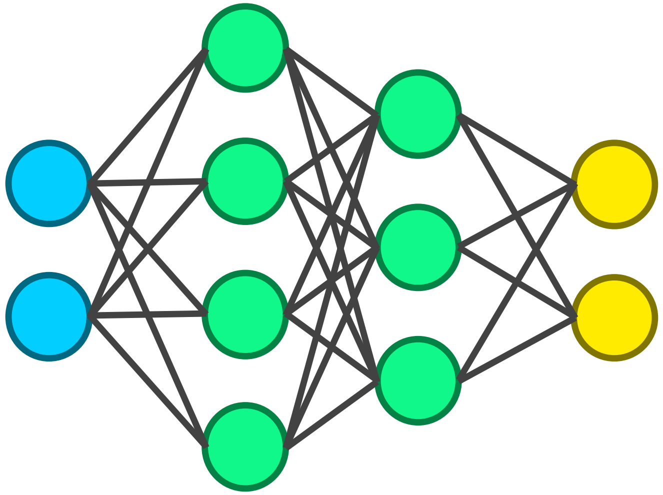


QML
Quantum algorithms feed with classical or quantum data

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

From classical to quantum NN

Classical

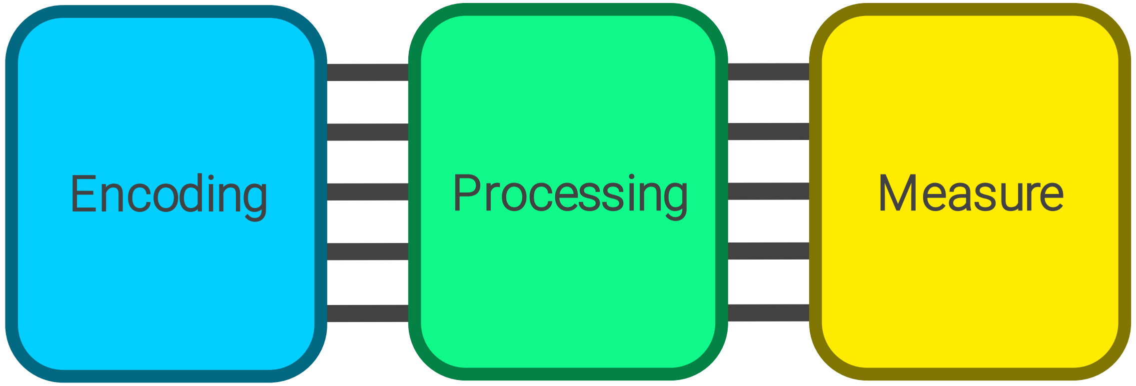


Input
neurons

Hidden
neurons

Output
neurons

Quantum
(circuit centric)



K Mitarai, M Negoro, M Kitagawa, K Fujii Phys. Revs A 98 (3), 032309 (2018)

E. Farhi and H.Neven, arXiv:1802.06002 (2018)

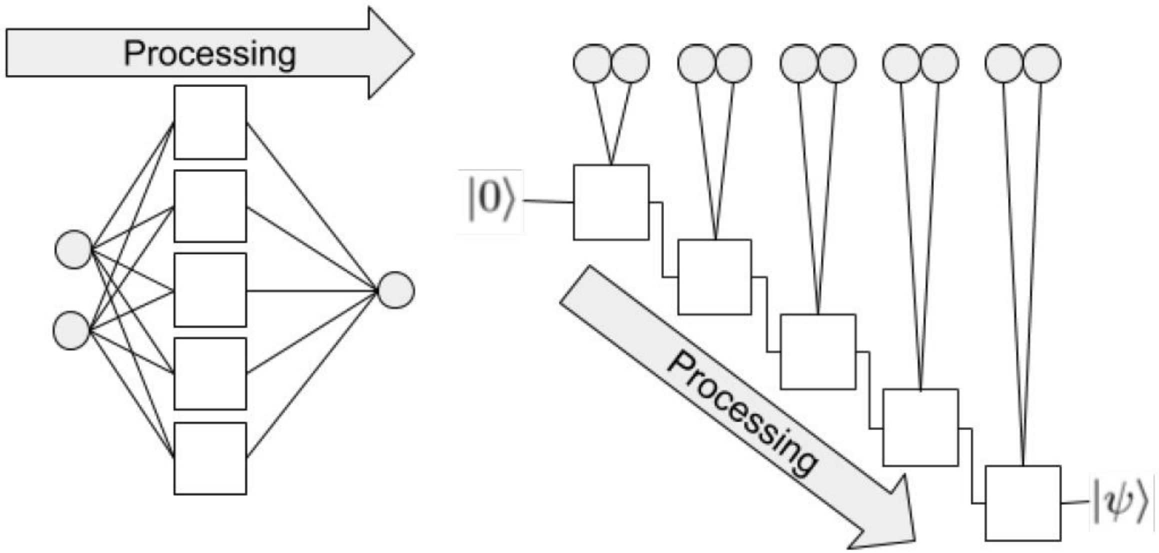
M. Schuld and N. Killoran, Phys. Rev. Lett. 122, 040504 (2019)

M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, Phys. Rev. A 101, 032308 (2020)

The minimal QNN

What is the most simple (but universal) NN? → Single hidden layer NN

What is the most simple (but universal) QNN? → Single-qubit QNN



(a) Neural network (b) Quantum classifier

Encoding the data

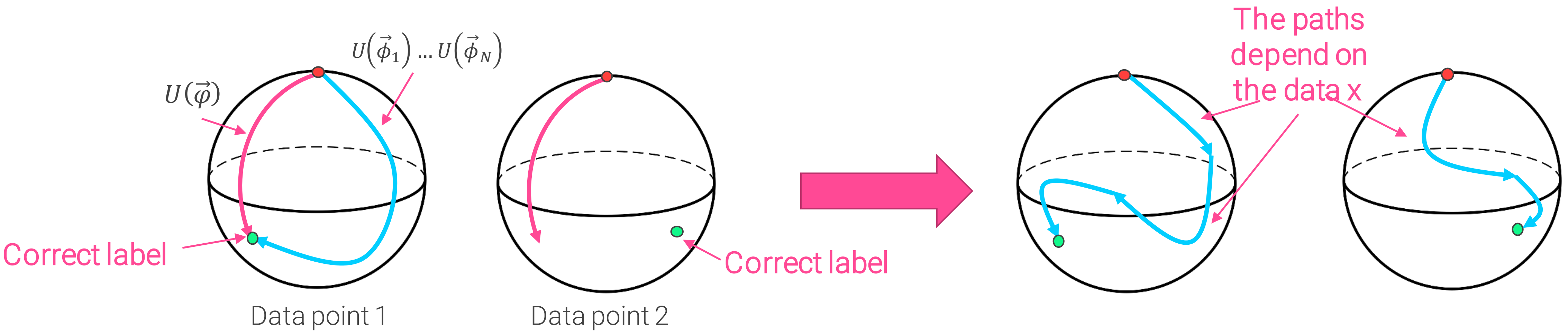
A product of unitaries can be written with a single unitary

$$U\left(\vec{\phi}_1\right) \dots U\left(\vec{\phi}_N\right) \equiv U(\vec{\varphi})$$

Data re-uploading

If we add some fixed parameter dependency (the data), the operation becomes flexible and data-depedent.

$$\mathcal{U}(\vec{\phi}, \vec{x}) \equiv U(\vec{\phi}_N)U(\vec{x}) \dots U(\vec{\phi}_1)U(\vec{x})$$

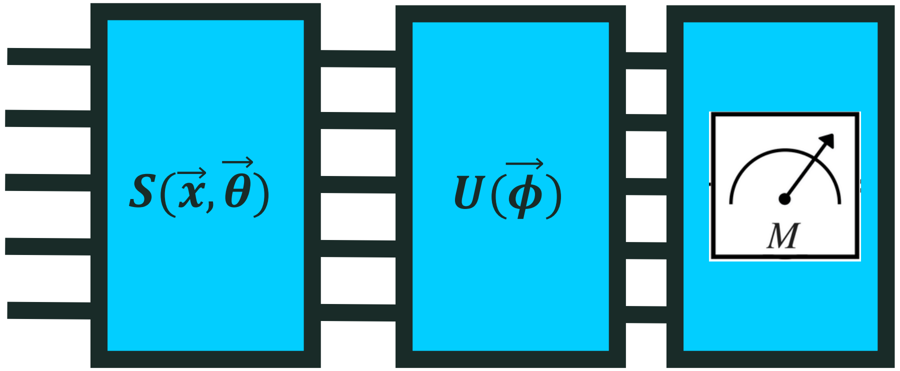


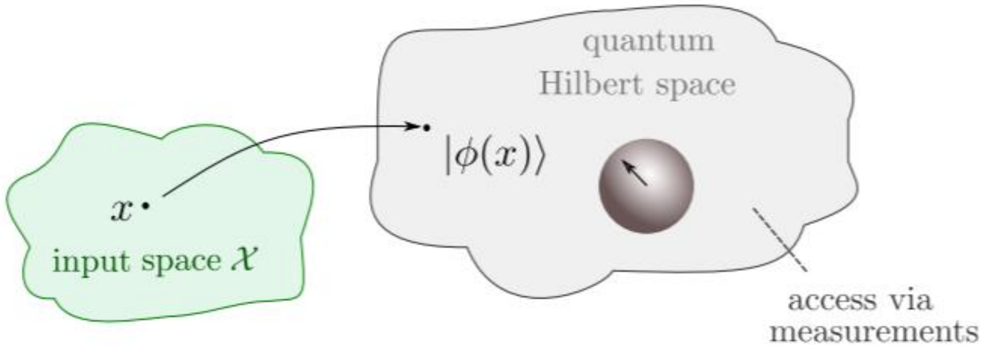
Supervised Learning

$$|\psi_0\rangle \rightarrow |\psi(\vec{x},\vec{\theta})\rangle \rightarrow |\psi(\vec{x},\vec{\theta},\vec{\phi})\rangle$$

Encode the data (quantum feature space)

Rotate to the correct measurement basis





We can then compute the Kernel

$$\kappa\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) \equiv \left\langle \Phi\left(\boldsymbol{x}_i\right) \middle| \Phi\left(\boldsymbol{x}_j\right)\right\rangle$$

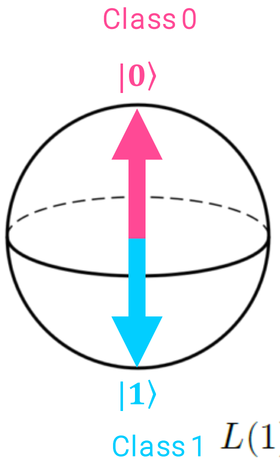
See Roman Krems lectures

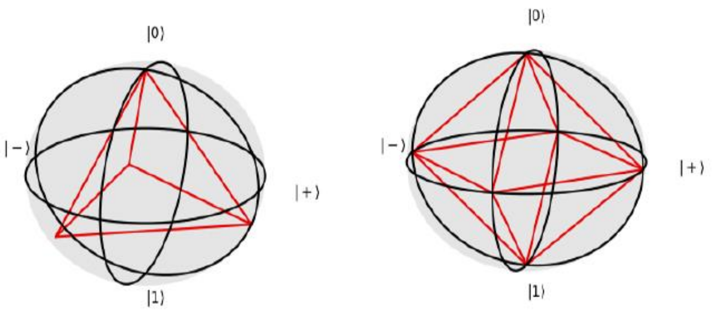
Or minimize the fidelity w.r.t. target states

$$C(\boldsymbol{\theta}) = \sum_{i=1}^{\mathcal{D}} \left(1 - |\langle y_i | \Psi(\boldsymbol{x}_i, \boldsymbol{\theta})\rangle|^2\right)$$

M. Schuld, arXiv:2101.11020 [quant-ph]

Example 1: single-qubit classifier



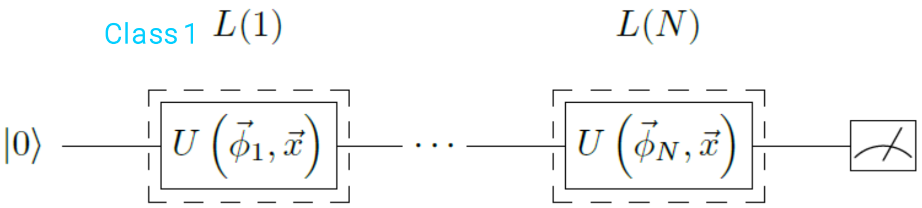


Target states

Divide the Bloch sphere into #classes sections

PQC

Layers of single-qubit gates where we encode the data and variational parameters into the angles.



$$L(i) = U\left(\vec{\theta}_i + \vec{w}_i \circ \vec{x}\right)$$

Loss function

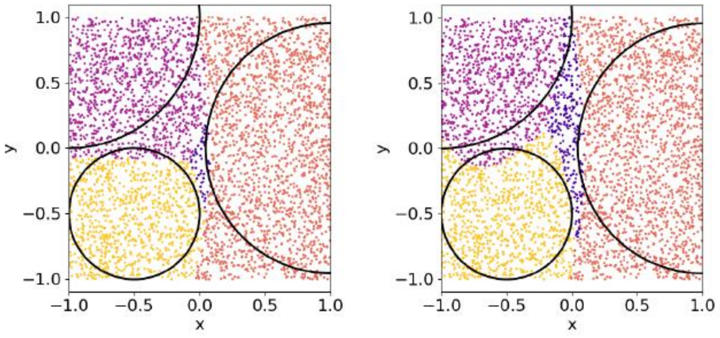
Overlap between the target state and the output state for all training points

$$\chi_f^2(\vec{\theta}, \vec{w}) = \sum_{\mu=1}^{M} \left(1 - |\langle \tilde{\psi}_s | \psi(\vec{\theta}, \vec{w}, \vec{x_\mu})\rangle|^2\right)$$

Quantum classifier

Once trained, we introduce the test points and classify them according to the qubit state.

Check the Tequila tutorial!



(a) 1 layer (b) 3 layers

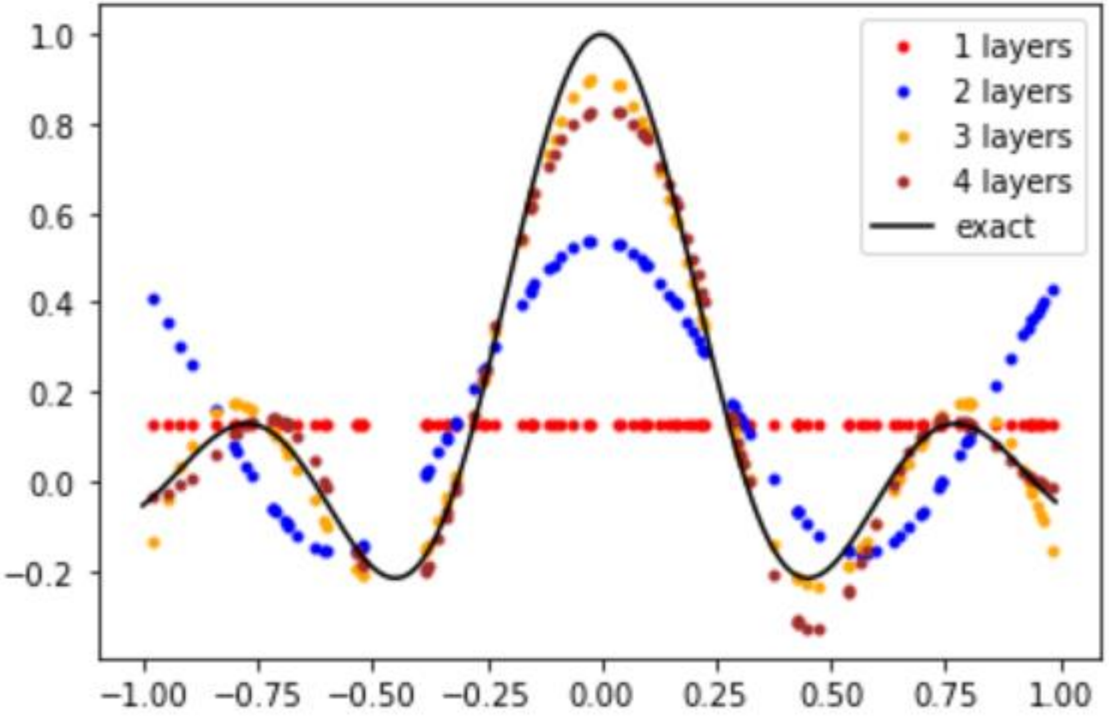
A. Pérez-Salinas, ACL, E. Gil-Fuster and J. I. Latorre, Quantum 4, 226 (2020)

Example 2: single-qubit approximant

Quantum circuits can be theoretically written as partial Fourier series and, therefore, they can be universal function approximators. The more data re-uploading, the more precision can be achieved.

Same PQC as the quantum classifier but the loss function will be:

$$\chi^2 = \frac{1}{M}\sum_{j=1}^{M}\left(\langle Z(x_j)\rangle - f(x_j)\right)^2$$



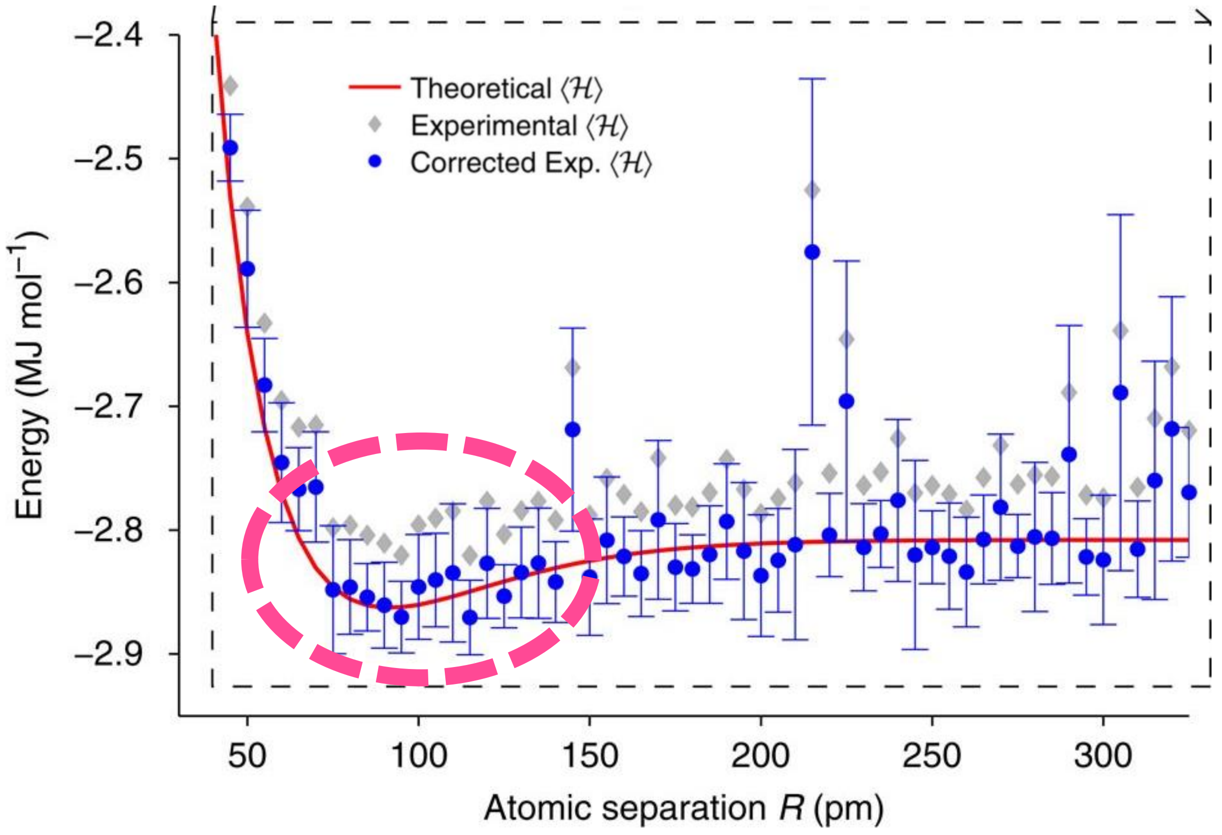
Check the Tequila tutorial!

M. Schuld, R. Sweke, J. J. Meyer, arXiv:2008.08605 [quant-ph]

A. Pérez-Salinas, D. López-Núñez, A. García-Sáez, P. Forn-Díaz, J. I. Latorre, arXiv:2102.04032 [quant-ph]

What's the true goal of VQE?

Bond dissociation curve of the He–H$^+$ molecule.



GOAL: find $|\psi\rangle$ that minimizes $\frac{\langle\psi|\mathcal{H}|\psi\rangle}{\langle\psi|\psi\rangle}$.

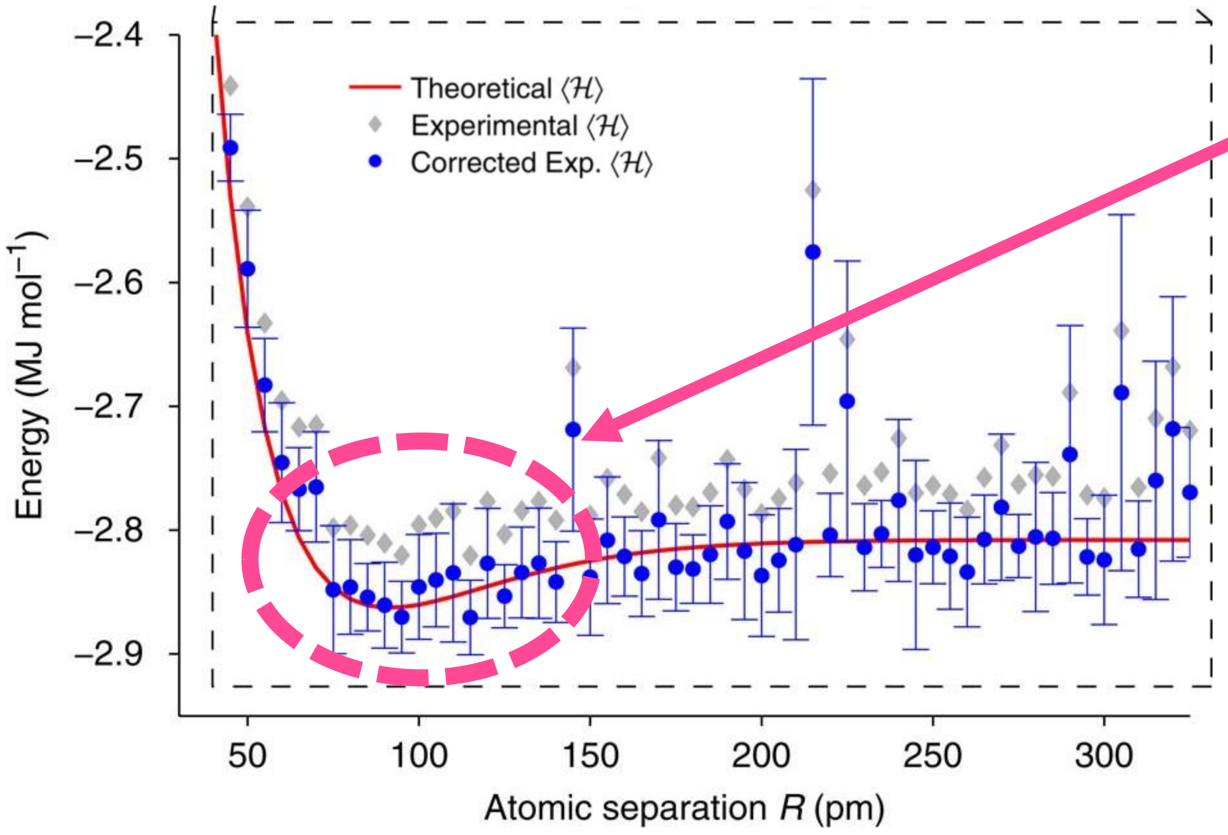
Find the atomic separation that minimizes the energy

$$\min\langle H(R)\rangle$$

A. Peruzzo, J. McClean, P. Shadbolt, M.-H.Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik , J. L. O'Brien, Nature Comm. **5**, 4213 (2014)

What's the true goal of VQE?

Bond dissociation curve of the He–H⁺ molecule.



To obtain this you need to scan from 0 to 300.

Each blue point is a VQE, that is, you have to **prepare, run and optimize** the quantum circuit.

Can we avoid to compute the uninteresting points?

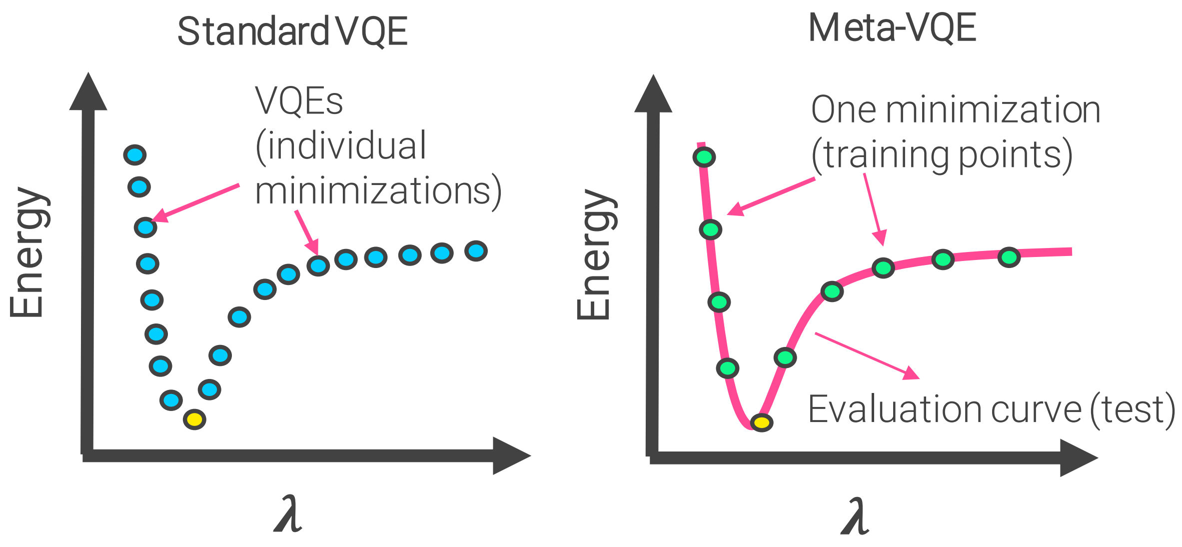
A. Peruzzo, J. McClean, P. Shadbolt, M.-H.Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik , J. L. O'Brien, Nature Comm. **5**, 4213 (2014)

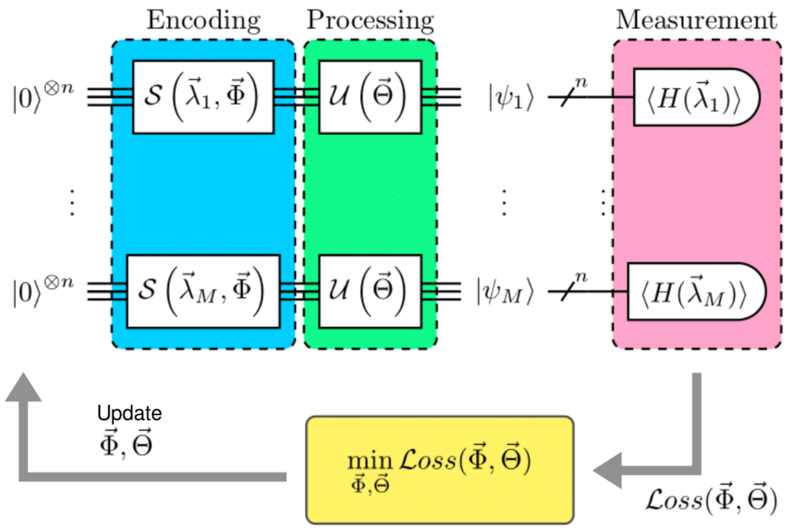
Example 3: Meta-VQE

Parameterized Hamiltonian $H\left(\vec{\lambda}\right)$

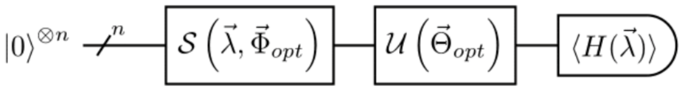
<u>Goal:</u> to find the quantum circut that **encodes** the ground state of the Hamiltonian for any value of $\vec{\lambda}$

- Training points: $\vec{\lambda}_i$ for $i = 1, \dots, M$
- Data re-uploading to encode the $\vec{\lambda}_i$ into the circuit
- Loss function with all $\left\langle H(\vec{\lambda}_i) \right\rangle$

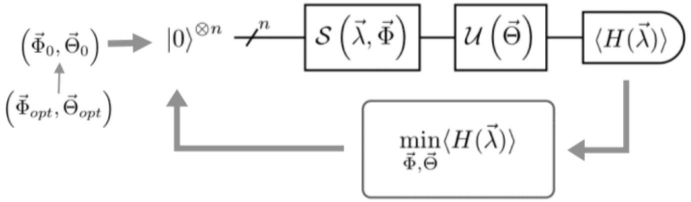




Option 1: run the circuit with test $\vec{\lambda}$ and obtain the g.s. energy profile.



Option 2: use $\vec{\Phi}_{opt}$ and $\vec{\Theta}_{opt}$ as starting point of a standard VQE optimization (<u>*opt-meta-VQE*</u>)

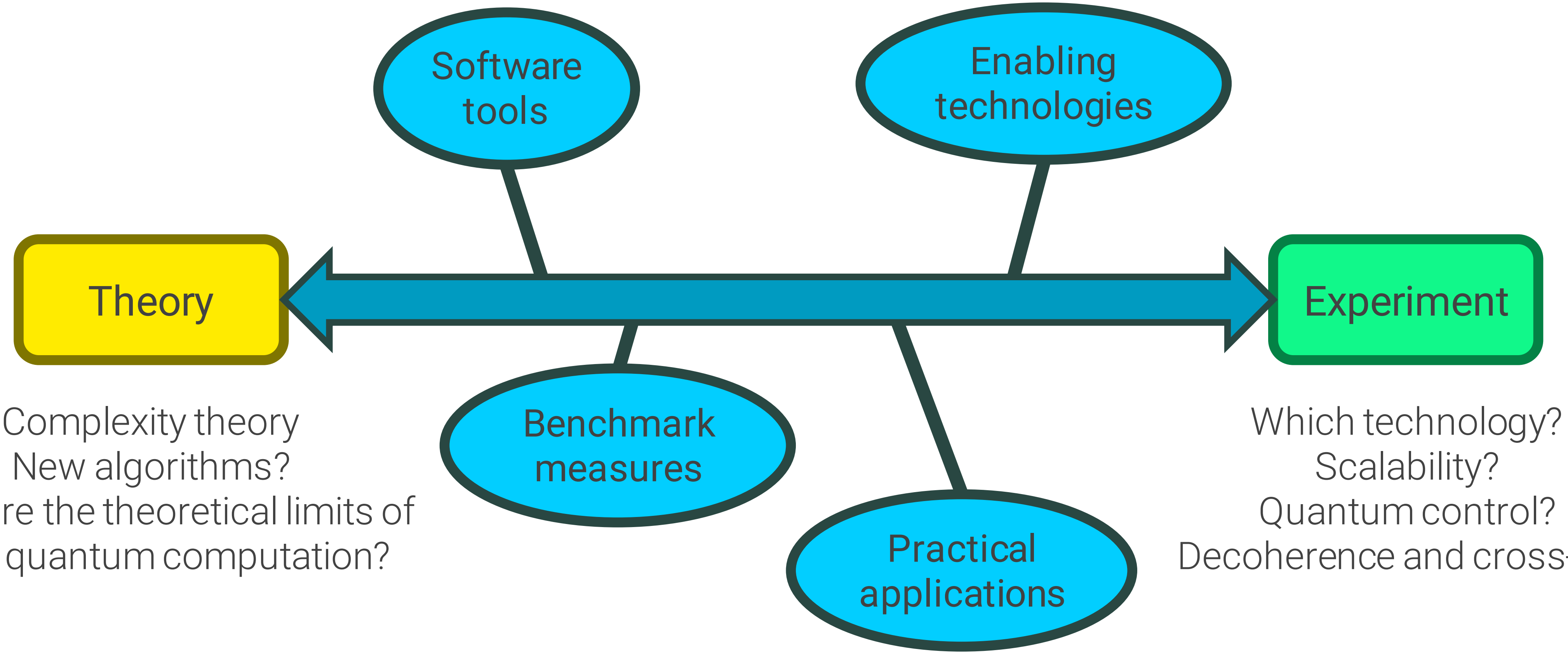


Check the Tequila tutorial!

ACL, J. Kottmann, A. Aspuru-Guzik, PRX Quantum 2, 020329 (2021)

NISQ horizon

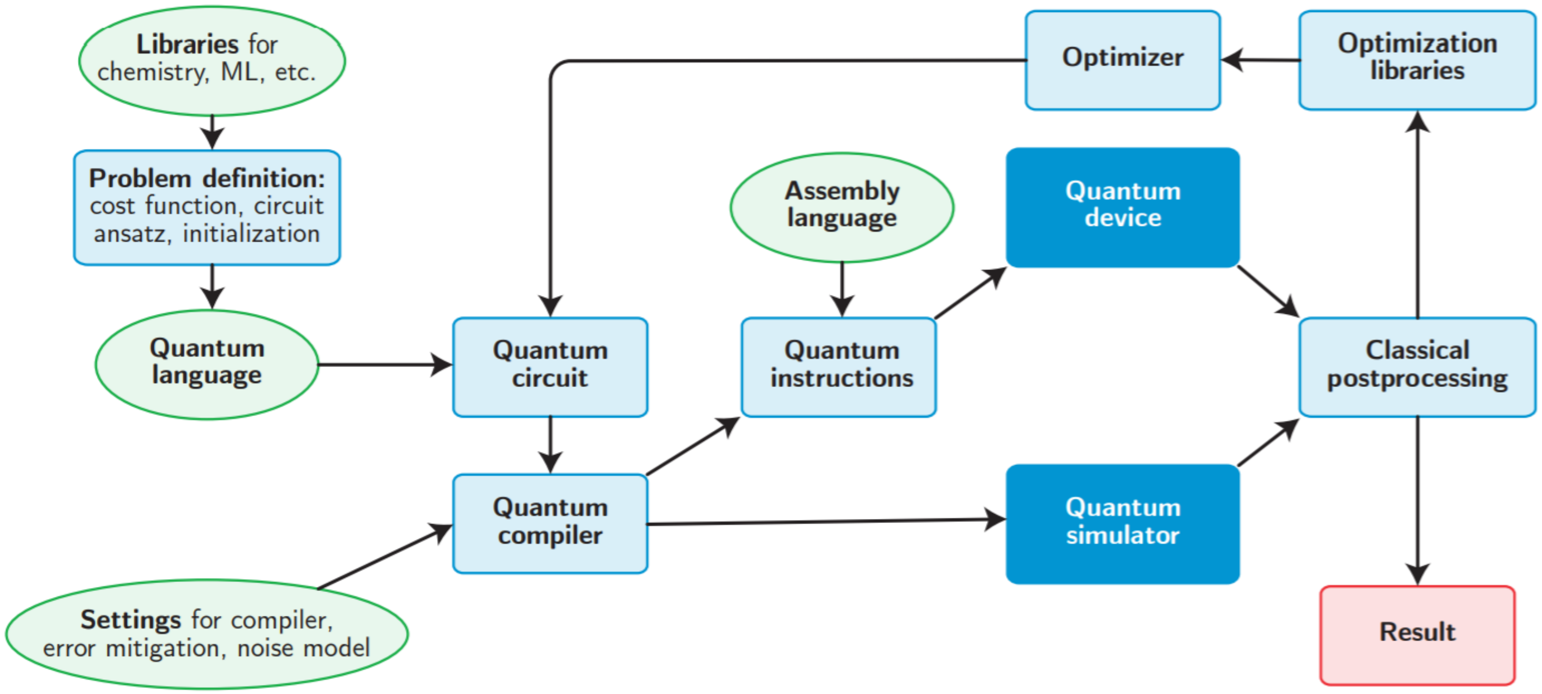
NISQ road



Complexity theory
New algorithms?
What are the theoretical limits of NISQ quantum computation?

Which technology?
Scalability?
Quantum control?
Decoherence and cross-talk

How to program a NISQ algorithm



Qiskit
Cirq
<Q#>
PENNYLANE
PYQUIL
Qibo
|Q>
TEQUILA
JAX
SciPy
GPyOpt
Phoenics
Mitiq
Orquestra
Psi4
OpenFermion

J. Kottmann, S. Alperin-Lea, T. Tamayo-Mendoza, et. al., arXiv:2011.03057 (accepted in Quantum Science and Technology)

Next goal: fault-tolerant quantum computing

Quantum Error Correction: protect the quantum information in a highly entangled state.

QEC comes with a big qubit overhead: thousands (posible milions) of qubits to implement a quantum advantage experiment.

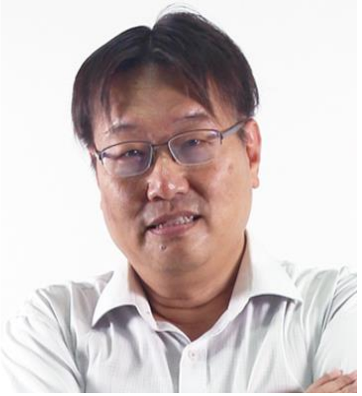
That's why we have NISQ… but most of the NISQ algorithms can also be implemented in the **Fault-Tolerant era.**

Noise limits NISQ algorithms such as VQAs, we do not always have performance garantees.

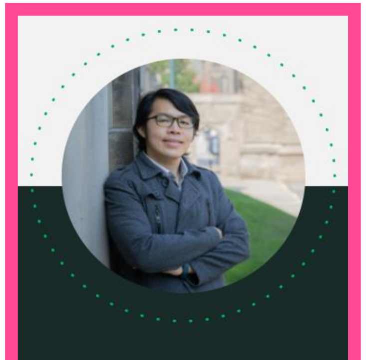
We don't know how much time will it take or even if it's possible to achieve F-T QC, but there is so much physics to explore along the way!

Acknowledgements

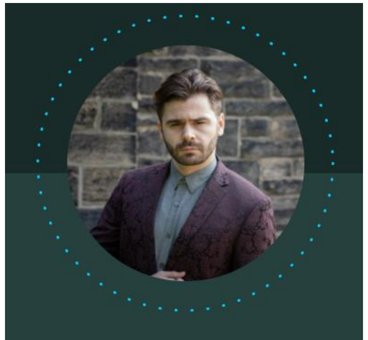

Alán Aspuru-Guzik

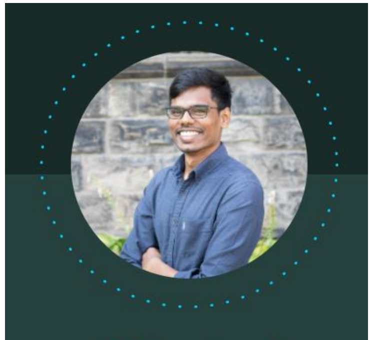

Leong-Chuan Kwek


Kishor Bharti


Thi Ha Kyaw


Tobias Haug

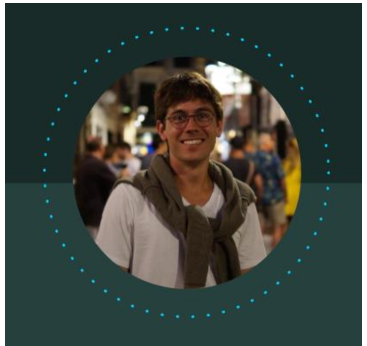

Sumner Alperin


Abhinav Anand

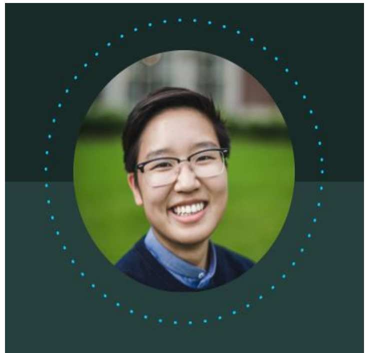

Matthias Degroote


Hermanni Heimonen


Jakob Kottmann


Tim Menke

Wai-Keong Mok


Sukin (Hannah) Sim

the matter lab

Centre for Quantum Technologies

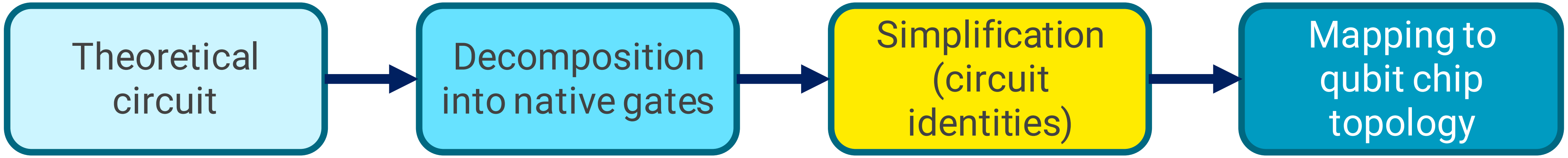
Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

~15 min Break...

Next:
Coding time!

Backup slides

Circuit compilation



Circuit simplification: use identities or tools like the ZX calculus (graph representation of quantum circuits)

"Interacting quantum observables: categorical algebra and diagrammatics", B. Coecke, R. Duncan, NJP 13 (4): 043016 (2011).

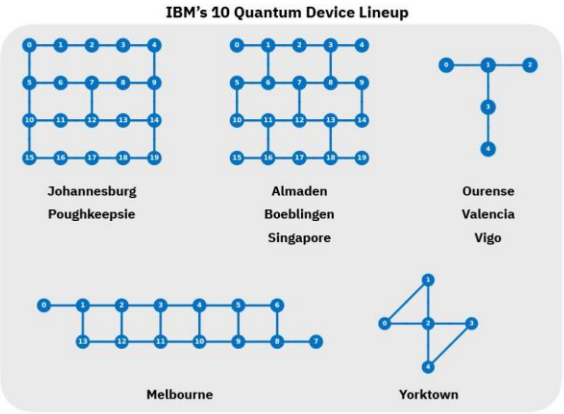
Noisy Intermediate-Scale Quantum (NISQ) algorithms, K. Bharti, ACL, T.H. Kyaw, et. al., arXiv:2101.08448 (2021)

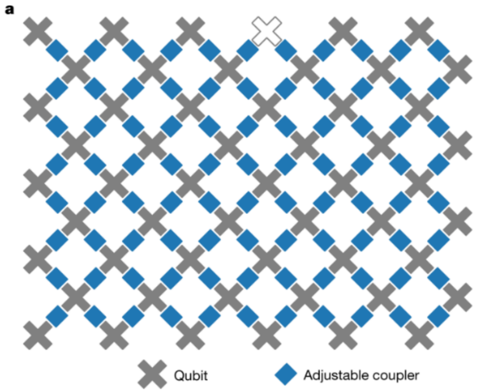
Circuit compilation

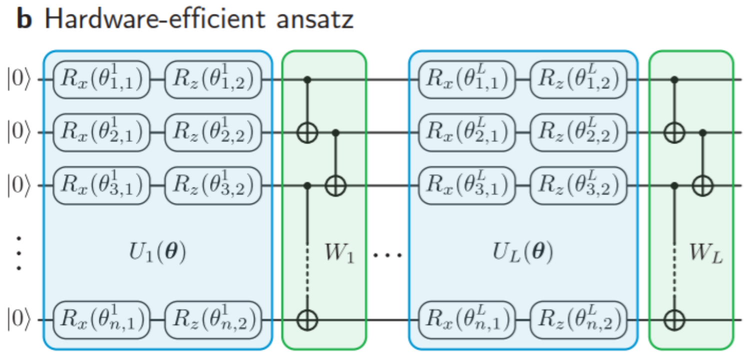
Theoretical circuit → Decomposition into native gates → Simplification (circuit identities) → Mapping to qubit chip topology

Circuit simplification: use identities or tools like the ZX calculi (graph representation of quantum circuits)

Qubits connectivity problem: not all qubits are physically connected, so we have to map our quantum circuits to the real devices.

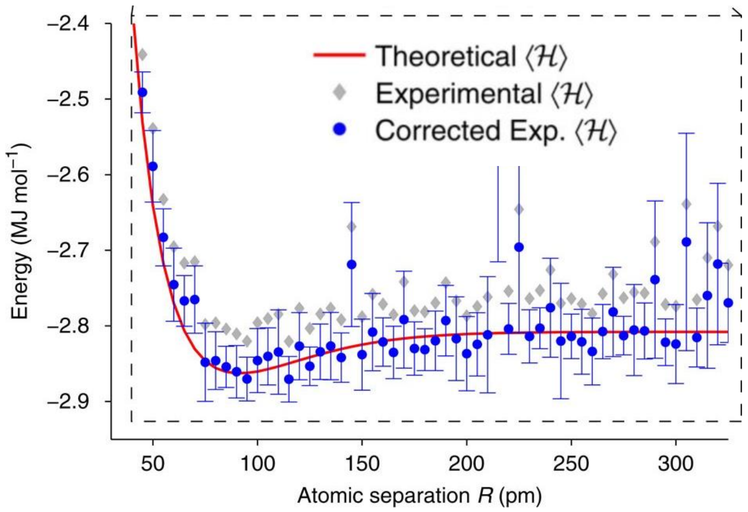




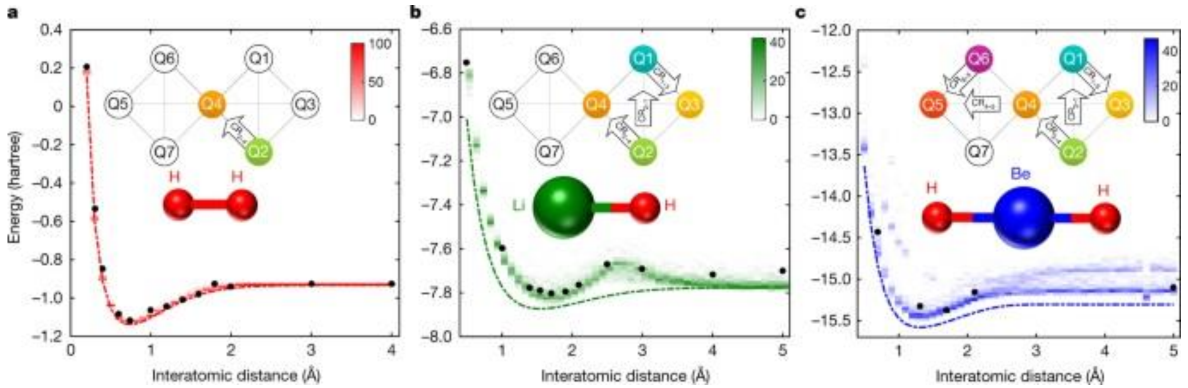


The Variational Quantum Eigensolver

Bond dissociation curve of the He–H⁺ molecule.

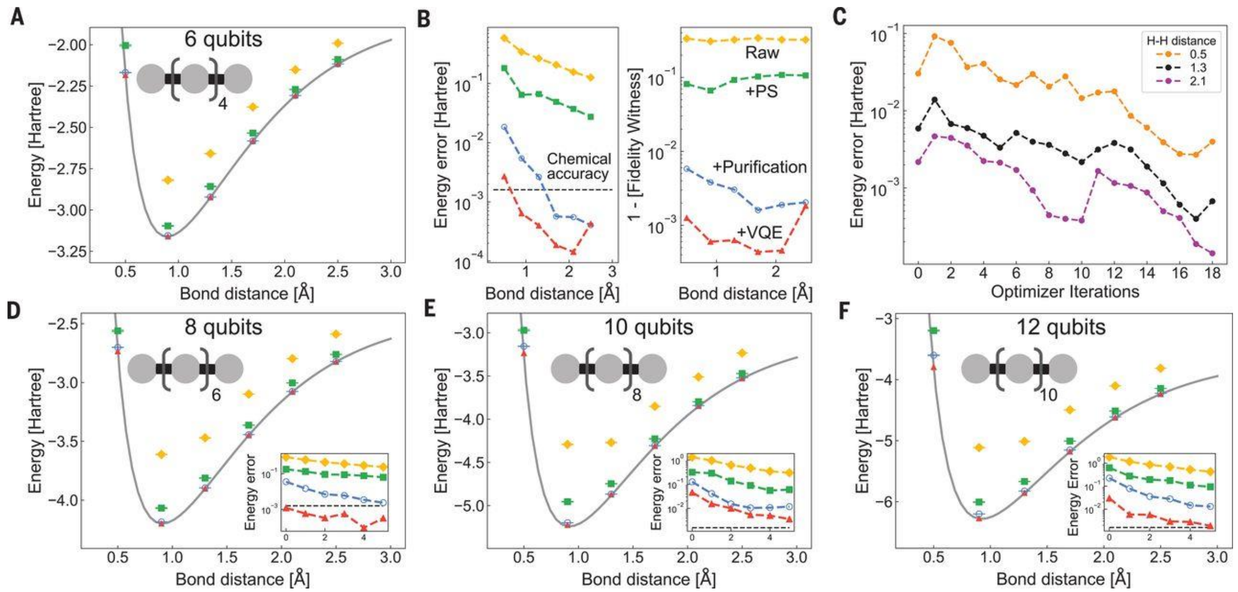


A. Peruzzo et. al., Nature Comm. 5, 4213 (2014)



A. Kandala et. al., Nature 549, 242–246 (2017)

H chains



Google AI Quantum and Collaborators, Science 369, 6507, 1084-1089 (2020)

Quantum Approximate Optimization Algorithm (QAOA)

<u>Resources:</u>

Musty Thoughts blog (Michał Stęchły):

https://www.mustythoughts.com/quantum-approximate-optimization-algorithm-explained

<u>Tutorials:</u>

- *Qiskit*: https://qiskit.org/textbook/ch-applications/qaoa.html
- *Pennylane*: https://pennylane.ai/qml/demos/tutorial_qaoa_intro.html, https://pennylane.ai/qml/demos/tutorial_qaoa_maxcut.html

Preliminaries

Time evolution:

$$i\hbar \frac{d}{dt}|\Psi(t)\rangle = H|\Psi(t)\rangle \qquad |\Psi(t)\rangle = e^{-i\,H\,t}|\Psi(0)\rangle$$

Trotterization

$$e^{A+B} = \lim_{n\to\infty}\left(e^{A/n}\ e^{B/n}\right)^n$$

$$H = H_1 + H_2$$

$$e^{-iHt} = e^{-itH_1 - itH_2} = \lim_{n\to\infty}\left(e^{-itH_1/n}\ e^{-itH_2/n}\right)^n$$

Apply alternatively
$e^{-itH_1}\ e^{-itH_2}$
in intervals of t/n

Adiabatic Quantum Evolution

$$H = H_M + H_P$$

$$H(s) = sH_M + (1-s)H_P$$

If s small, we end up in the ground state of H_P (under certain assumptions)

Quantum Approximate Optimization Algorithm

Can be understood as an approximation of the Trotter decompositiong of adiabatic evolution.

Mixing Hamiltonian

$$H_M \equiv \sum_{i=1}^{n} \hat{\sigma}_x^i$$

Problem Hamiltonian

$$H_P \equiv \sum_{i=1}^{n} C(e_i)|e_i\rangle$$

Construct the circuit ansatz by alternating the problem and mixing Hamiltonians where β and γ are the variational parameters to be optimized classically.

full superposition state (in general)

$$|\Psi(\gamma,\beta)\rangle \equiv e^{-i\beta_p H_M} e^{-i\gamma_p H_P} \cdots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_P} |D\rangle$$

Objective function: $\langle\Psi(\gamma,\beta)|\, H_P(\gamma,\beta)\, |\Psi(\gamma,\beta)\rangle$

E. Farhi, J. Goldstone, S. Gutmann, arXiv:1411.4028 (2014)

Some comparisons

Check Tequila QAOA vs VQE tutorial at

`github.com/AlbaCL/VQA_tutorials`

	QAOA	VQE	Adiabatic Quantum Evolution
Goal	Find an approximation of the ground state and its energy	Find an approximation of the ground state and its energy	End up in the ground state
Parameters	β, γ can take any value	θ, can take any value	s must be small
Computational paradigm	Digital	Digital	Analog
Circuit ansatz	Problem-specific, alternating	Problem-specific or other (e.g. Hardware-efficient)	-

Universal Approximation Theorem

activation function

\# neurons

Any continous function $f(x)$ can be approximated with ϵ accuracy by the function

$$h(\vec{x}) = \sum_{i=1}^{N} \alpha_i \varphi(\vec{w}_i \cdot \vec{x} + b_i)$$

$\alpha_i, b_i \in \mathbb{R}$

$\vec{w}_i \in \mathbb{R}^m$

output weigths

weigths

biases

where φ is a nonconstant, bounded and continuous function.

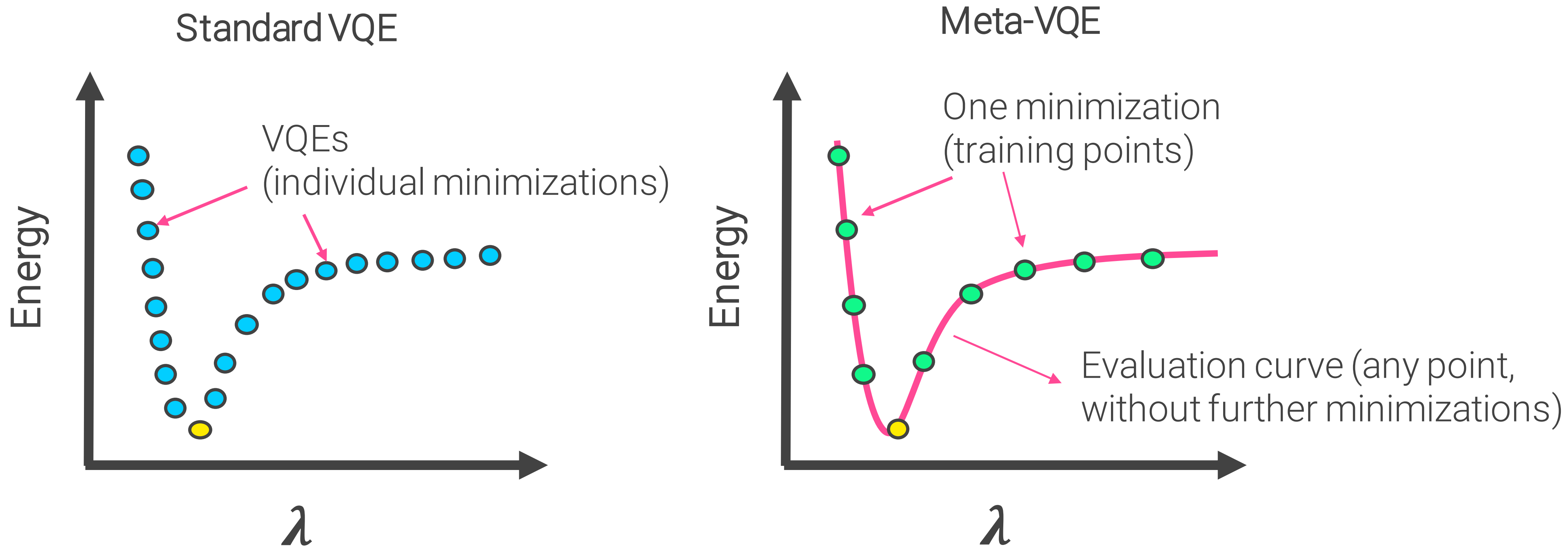
A single-layer neural network can approximate any continous function (providing enough neurons in the hidden layer)

K. Hornik, Neural Networks 4, 251 (1991).

Meta-VQE outlook

Parameterized Hamiltonian $H(\vec{\lambda})$

Goal: to find the quantum circut that **encodes** the ground state of the Hamiltonian for any value of $\vec{\lambda}$



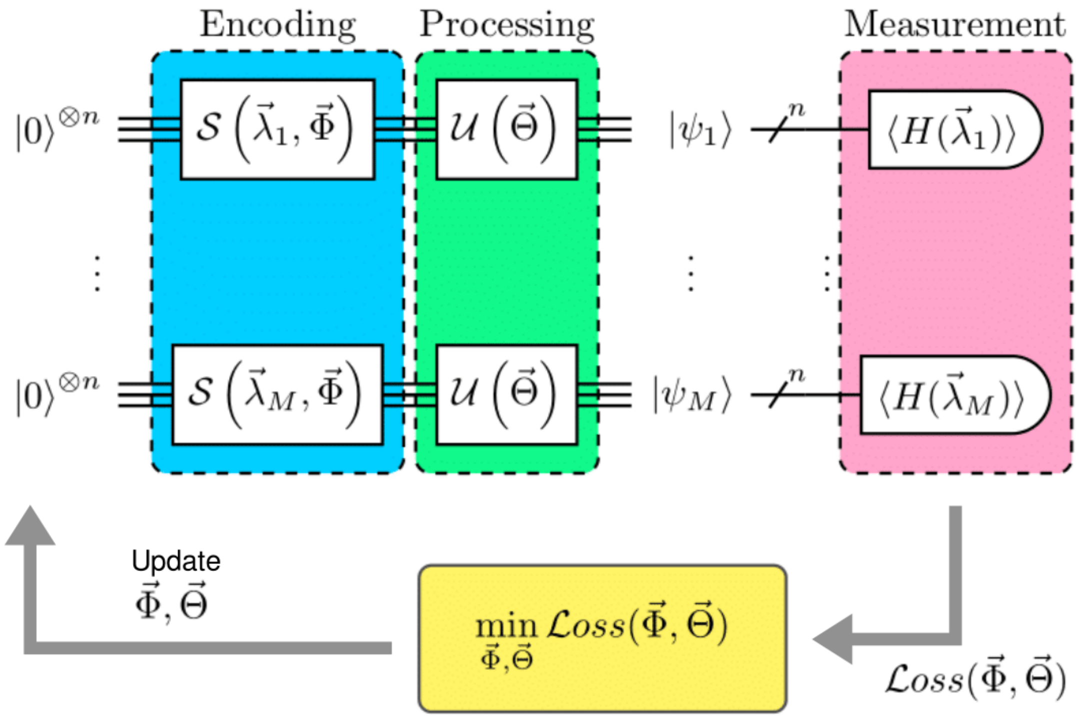
See also: K. Mitarai, T. Yan, K. Fujii, Phys. Rev. Applied **11**, 044087 (2019)

The Meta-VQE

Parameterized Hamiltonian $H\left(\vec{\lambda}\right)$

Training points: $\vec{\lambda}_i$ for $i = 1, \ldots, M$

Loss function with all $\left\langle H(\vec{\lambda}_i)\right\rangle$



<u>Goal:</u> to find the quantum circut that **encodes** the ground state of the Hamiltonian for any value of $\vec{\lambda}$

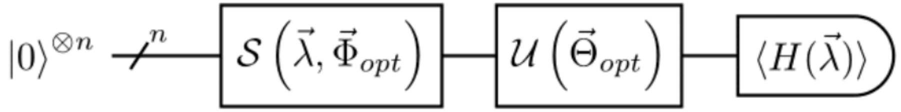
Output: $\vec{\Phi}_{opt}$ and $\vec{\Theta}_{opt}$

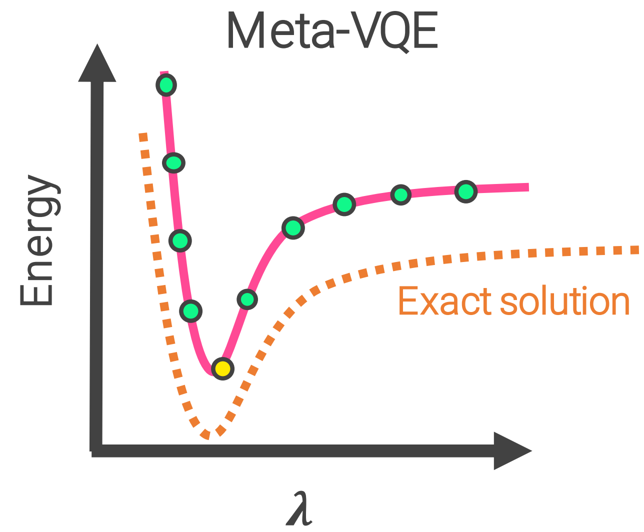
See also: K. Mitarai, T. Yan, K. Fujii, Phys. Rev. Applied **11**, 044087 (2019)

The Meta-VQE output

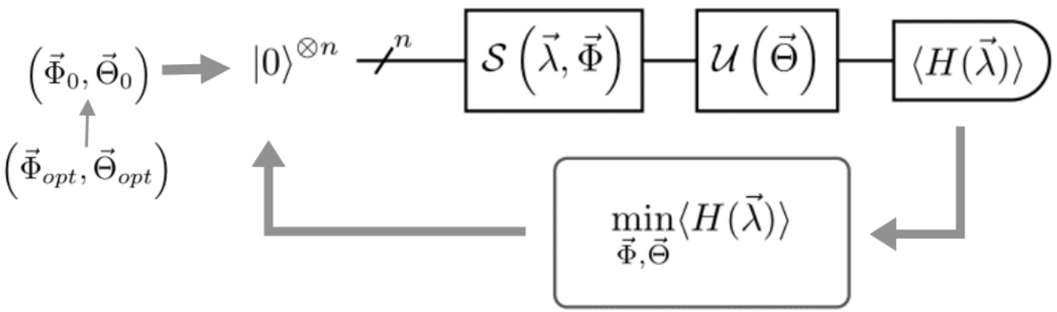
Output: $\vec{\Phi}_{opt}$ and $\vec{\Theta}_{opt}$

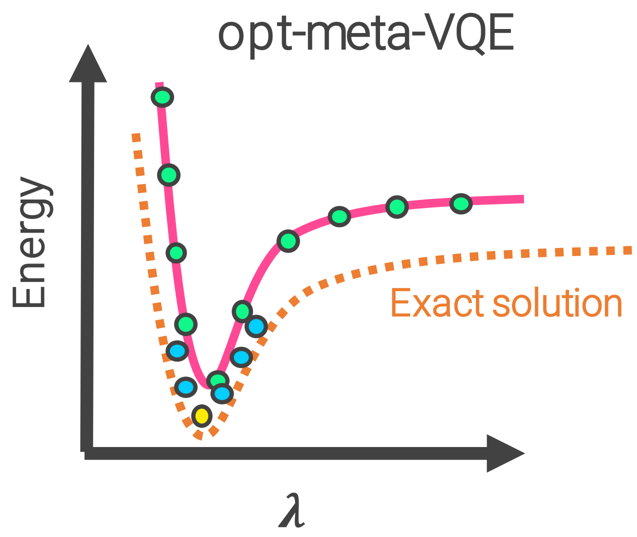
Option 1: run the circuit with test $\vec{\lambda}$ and obtain the g.s. energy profile.





Option 2: use $\vec{\Phi}_{opt}$ and $\vec{\Theta}_{opt}$ as starting point of a standard VQE optimization (*opt-meta-VQE*)





1D XXZ spin chain

Check the Tequila tutorial!

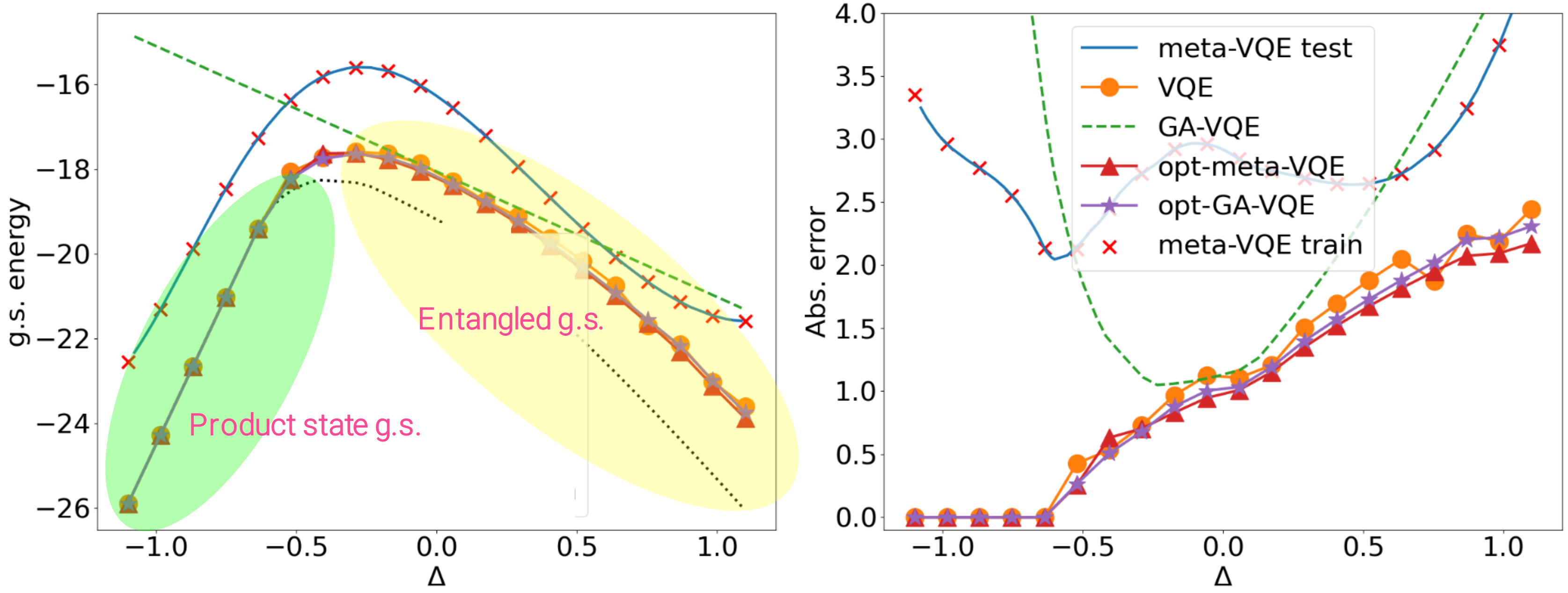
14 qubits simulation, $\lambda = 0.75$

Linear encoding: $R_z(w_1\Delta + \phi_1)R_y(w_2\Delta + \phi_2)\otimes$ Alternating CNOT

Processing layer: $R_z(\theta_1)R_y(\theta_2)\otimes$ Alternating CNOT

Results 2 encoding + 2 processing layers

$$H = \sum_{i=1}^{n} \sigma_i^x\sigma_{i+1}^x + \sigma_i^y\sigma_{i+1}^y + \Delta\sigma_i^z\sigma_{i+1}^z + \lambda\sigma_i^z$$



ACL, J. Kottmann, A. Aspuru-Guzik, PRX Quantum 2, 020329 (2021)

H_4 molecule

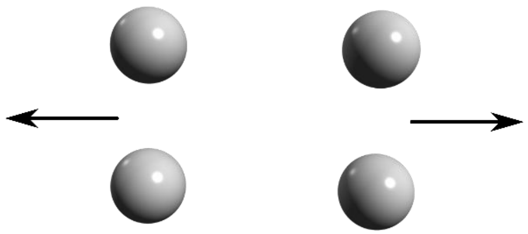
H_4 molecule in 8 spin-orbitals (STO-3G basis set)

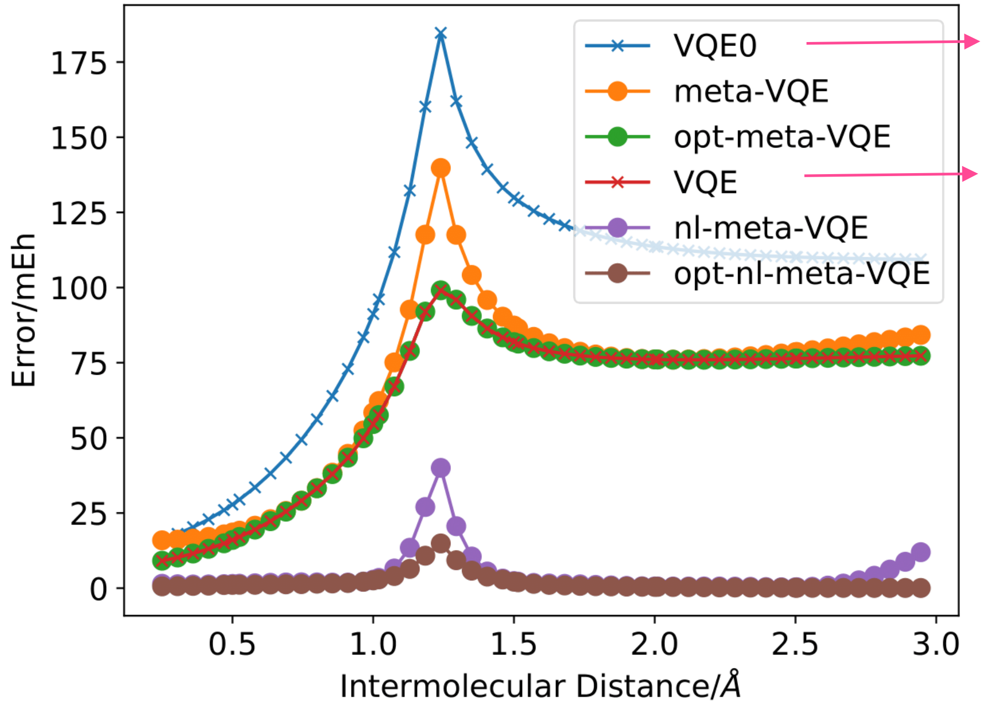
Ansatz: k-UpCCGSD (k=2 for these results)

Linear encoding: $\theta = \alpha + d\beta$

Non-linear encoding: $\theta = \alpha e^{\beta(\gamma - d)} + \delta$ (floating Gaussians)

Hamiltonian Parameter
(intermolecular distance)





Initial state: |0>

Initial state: |HF>