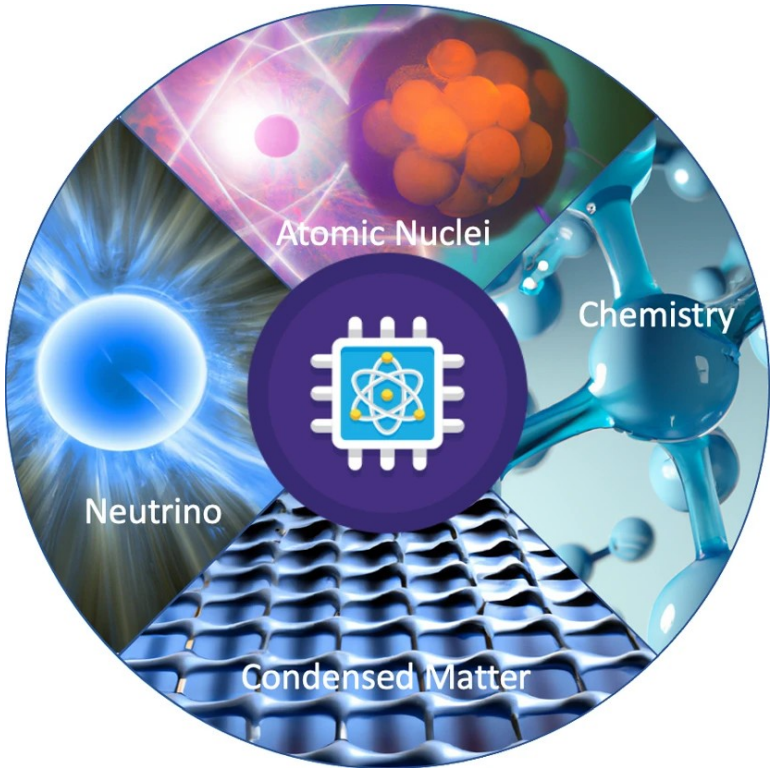
Calculation of Green's Functions using quantum computers for small superfluid systems

Samuel Aychet-Claisse
CEA Saclay & IJCLab Orsay

Supervisors : Vittorio Somà (CEA Saclay)
Denis Lacroix (IJCLab Orsay)

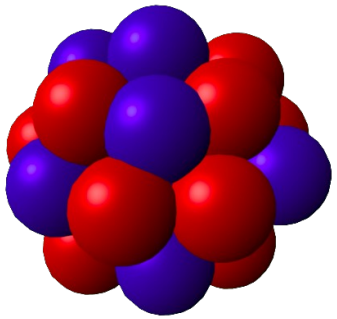


Picture taken from : Ayral, T., Besserve, P., Lacroix, D. et al. Quantum computing with and for many-body physics. Eur. Phys. J. A 59, 227 (2023). https://doi.org/10.1140/epja/s10050-023-01141-1

Contents

Contents

1.1 – Why quantum computing ?



A quantum many-body system :
the atomic nucleus

- Made of A = Z + N nucleons ($2 \leq A \leq 400$)

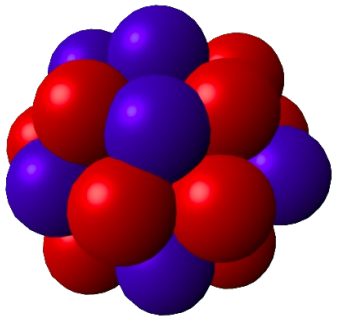
- Mesoscopic system
  - ↳ Rich phenomenology
  - ↳ Theoretical description challenging

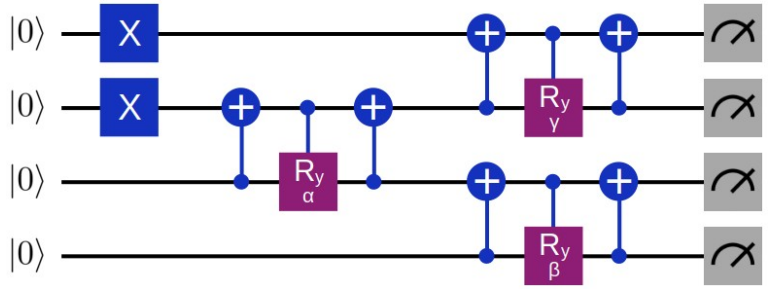
- Strong interaction in a non-perturbative regime
  - ↳ Complicated Hamiltonian
  - ↳ Quick growth of the (needed) Hilbert space

- Exact methods (diag of H) can be applied up to A~20.
Approximate methods exist but very costly.

→ Motivation for a quantum computing approach.

1.1 – Why quantum computing ?





A quantum many-body system :
the atomic nucleus

Not easy to model with a classical computer
Not easy to manipulate, to measure

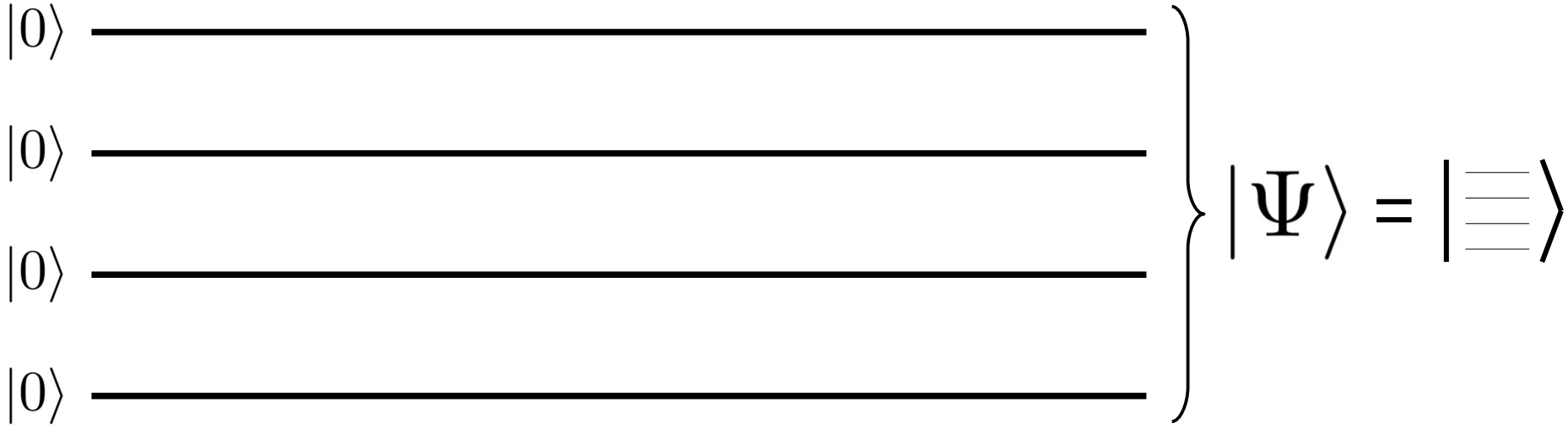
Another quantum many-body system

Not easy to model with a classical computer
More easy to manipulate, to measure (in principle)

Contents

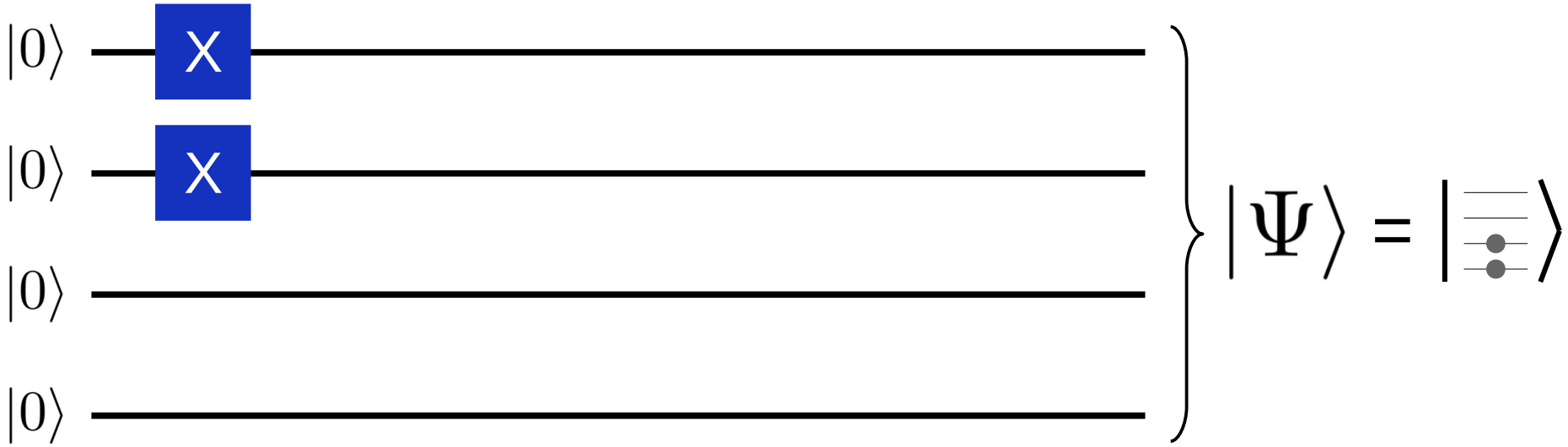
1.2 – Building a state

- Qubits provide an efficient representation of a many-body wavefunction.
- Operations on them allow to add the correlations needed to approximate the exact wavefunction.



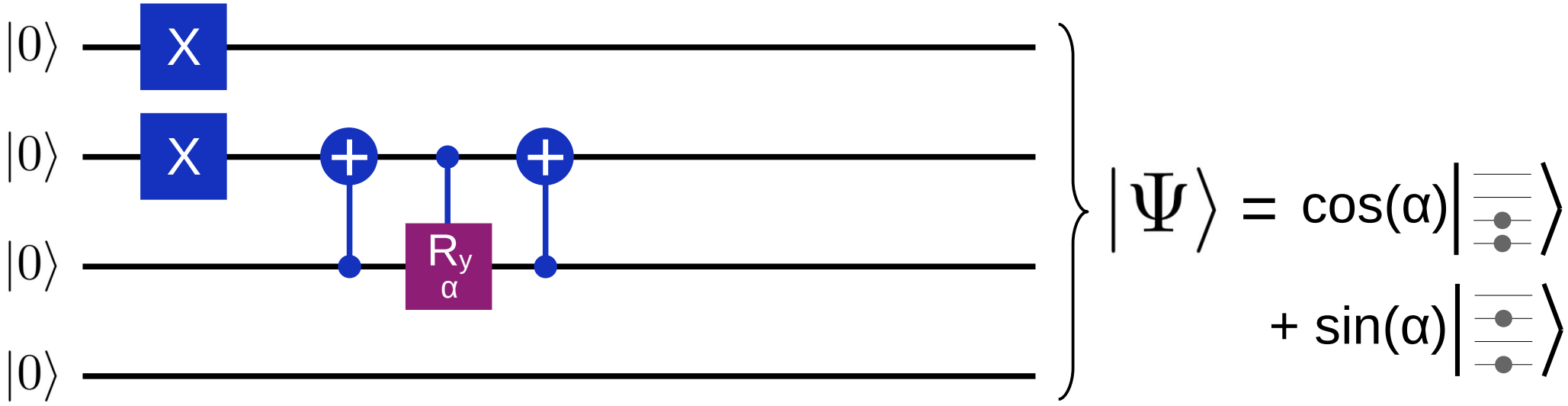
1.2 – Building a state

- Qubits provide an efficient representation of a many-body wavefunction.
- Operations on them allow to add the correlations needed to approximate the exact wavefunction.



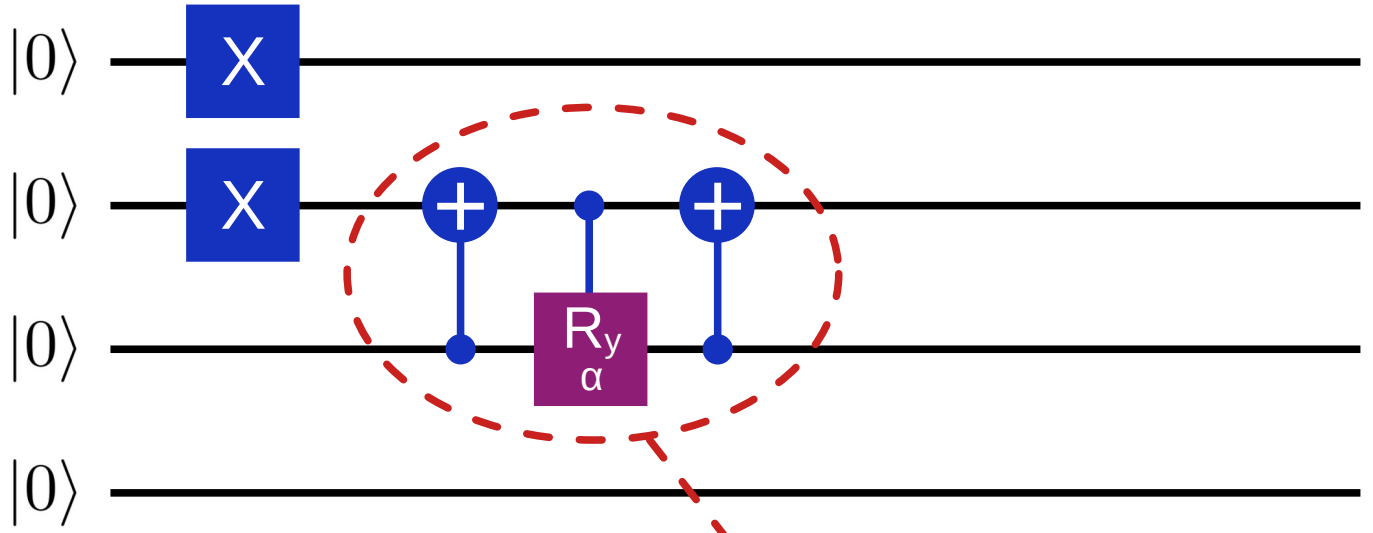
1.2 – Building a state

- Qubits provide an efficient representation of a many-body wavefunction.
- Operations on them allow to add the correlations needed to approximate the exact wavefunction.



1.2 – Building a state

- Qubits provide an efficient representation of a many-body wavefunction.
- Operations on them allow to add the correlations needed to approximate the exact wavefunction.



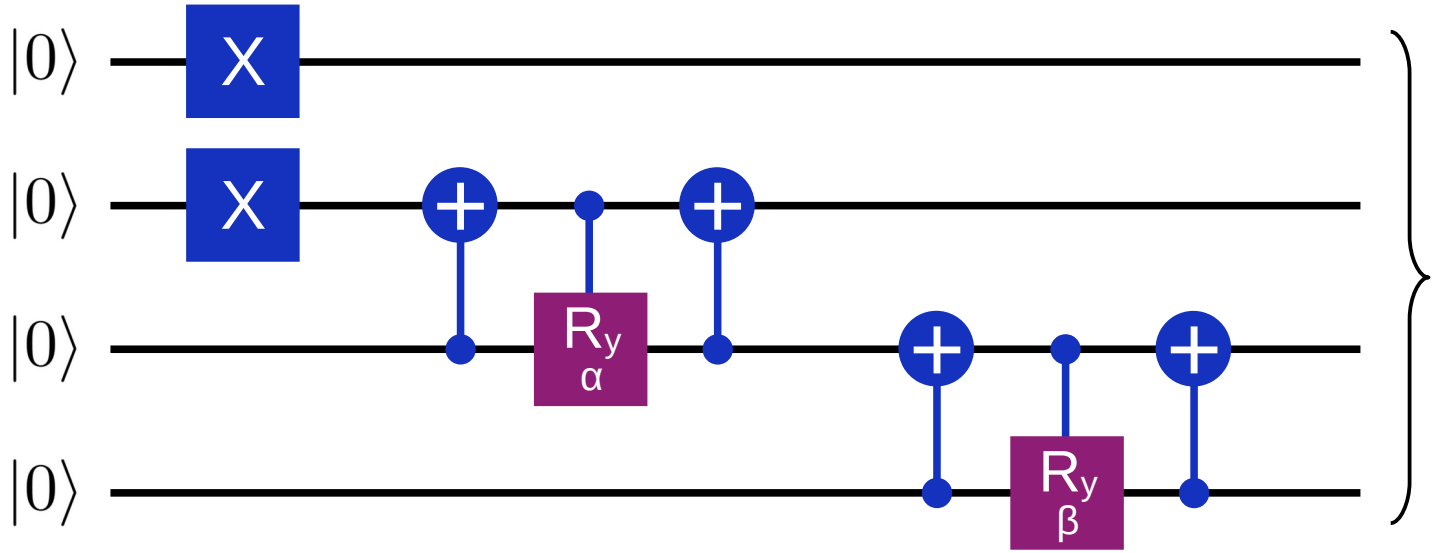
$$|\Psi\rangle = \cos(\alpha)\,|01\rangle + \sin(\alpha)\,|10\rangle$$

$$\begin{array}{c} |00\rangle \quad |01\rangle \quad |10\rangle \quad |11\rangle \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

Creates entanglement in the wavefunction

1.2 – Building a state

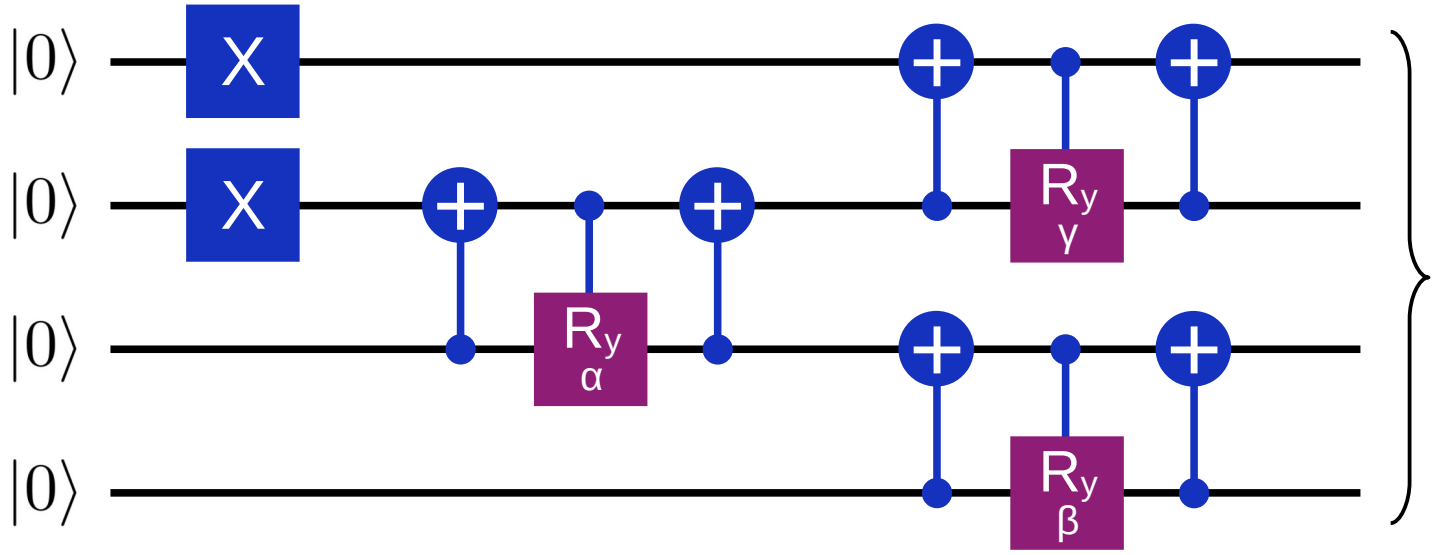
- Qubits provide an efficient representation of a many-body wavefunction.
- Operations on them allow to add the correlations needed to approximate the exact wavefunction.



$$|\Psi\rangle = \cos(\alpha)\left|\begin{matrix}\\ \bullet\\ \bullet\end{matrix}\right\rangle + \sin(\alpha)\cos(\beta)\left|\begin{matrix}\bullet\\ \bullet\\ \end{matrix}\right\rangle + \sin(\alpha)\sin(\beta)\left|\begin{matrix}\bullet\\ \\ \bullet\end{matrix}\right\rangle$$

1.2 – Building a state

- Qubits provide an efficient representation of a many-body wavefunction.
- Operations on them allow to add the correlations needed to approximate the exact wavefunction.

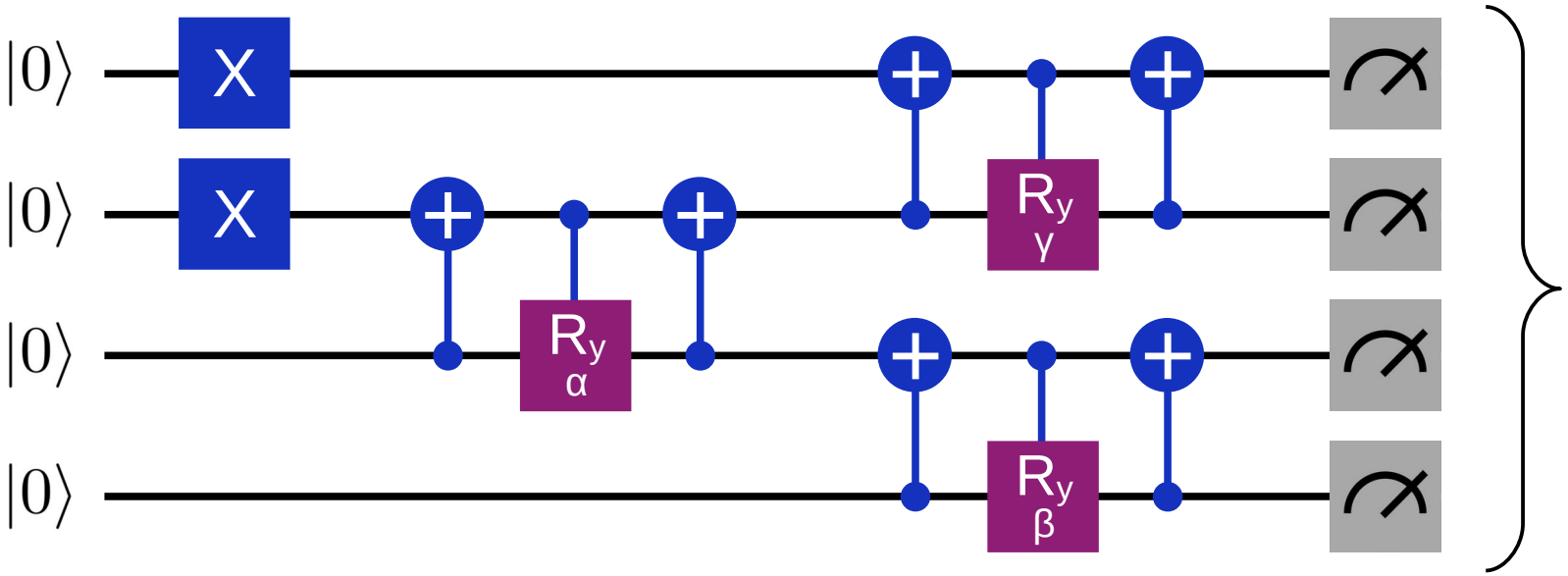


$$|\Psi\rangle = \cos(\alpha)|\,\rangle + \sin(\alpha)\cos(\beta)\cos(\gamma)|\,\rangle + \sin(\alpha)\cos(\beta)\sin(\gamma)|\,\rangle + \sin(\alpha)\sin(\beta)\cos(\gamma)|\,\rangle + \sin(\alpha)\sin(\beta)\sin(\gamma)|\,\rangle$$

- Ideally, n qubits are sufficient to represent an n-level many-body state.
- Classically, it requires (without symmetries) to store 2^n amplitudes (nb of bits = 2^n × nb of precision digits).

Contents

1.3 – Measuring expectation values



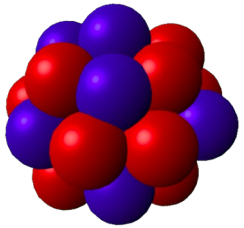
One can measure all Pauli string expectation values

$$\langle \Psi | \sigma_0 \sigma_1 \sigma_2 \sigma_3 | \Psi \rangle$$

with $\forall i, \sigma_i \in \{I, X, Y, Z\}$

1.3 – Measuring expectation values

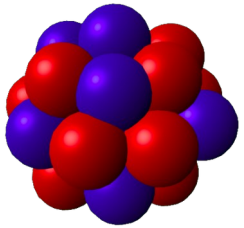
$$\hat{H} = \sum_{k=1}^{N} \epsilon_k (\hat{a}_k^\dagger \hat{a}_k + \hat{a}_{\bar{k}}^\dagger \hat{a}_{\bar{k}}) - g \sum_{k \neq l} \hat{a}_k^\dagger \hat{a}_{\bar{k}}^\dagger \hat{a}_{\bar{l}} \hat{a}_l$$



Hamiltonian in second quantization form (here : Pairing Hamiltonian)

1.3 – Measuring expectation values

$$\hat{H} = \sum_{k=1}^{N} \epsilon_k (\hat{a}_k^\dagger \hat{a}_k + \hat{a}_{\bar{k}}^\dagger \hat{a}_{\bar{k}}) - g \sum_{k \neq l} \hat{a}_k^\dagger \hat{a}_{\bar{k}}^\dagger \hat{a}_{\bar{l}} \hat{a}_l$$



Hamiltonian in second quantization form (here : Pairing Hamiltonian)

Map the second-quantized operators into Pauli operators

Ex : Jordan-Wigner Transformation (JWT) : $a_\nu^\dagger \longrightarrow \left(\prod_{k=1}^{\nu-1}(-Z_k)\right)\left(X_\nu - iY_\nu\right)$

$a_\nu \longrightarrow \left(\prod_{k=1}^{\nu-1}(-Z_k)\right)\left(X_\nu + iY_\nu\right)$

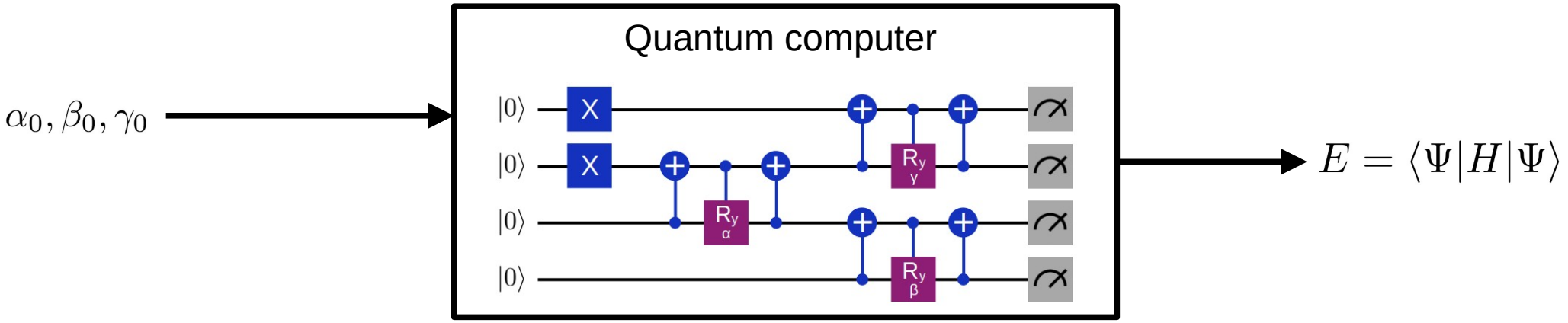
$$JWT[\hat{H}] = \sum_{\sigma_1 \ldots \sigma_N \in \{I,X,Y,Z\}} \lambda_{\sigma_1 \ldots \sigma_N} \hat{\sigma}_1 \ldots \hat{\sigma}_N$$

Transformed Hamiltonian as combination of Pauli terms

Contents

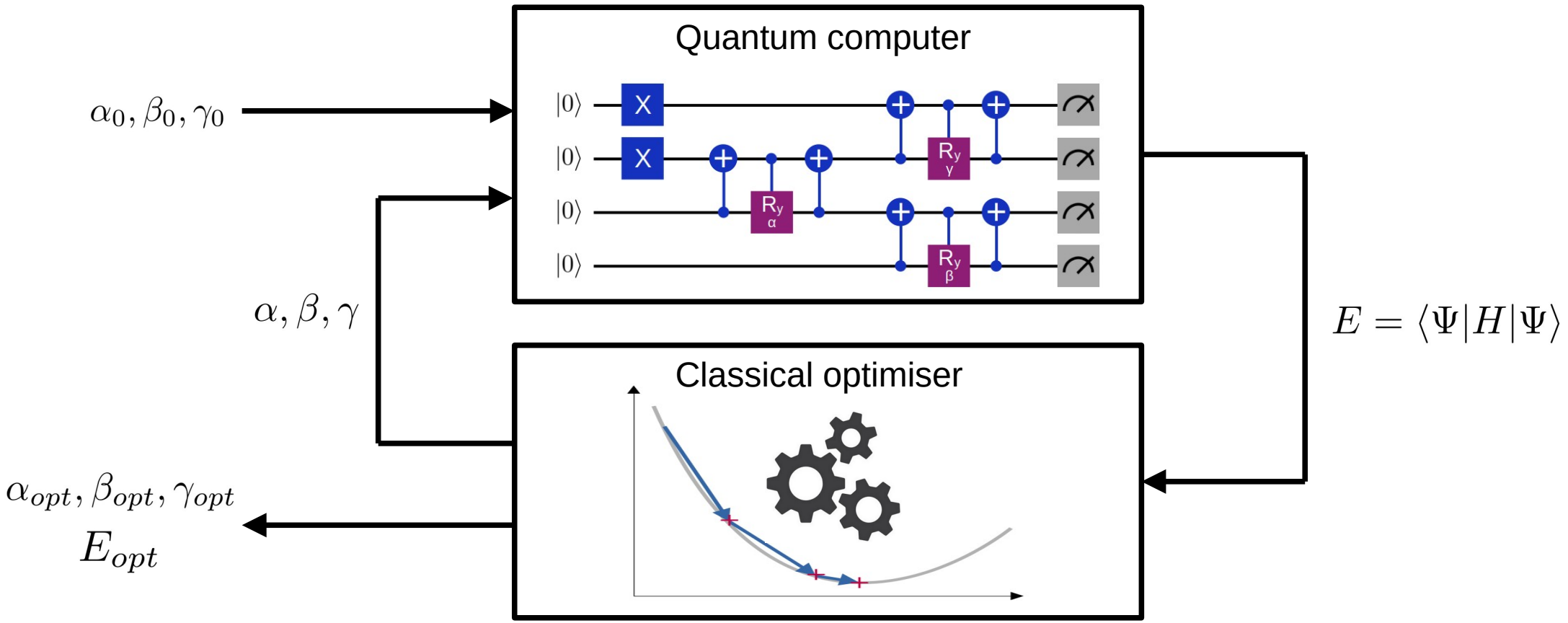
1.4 – Approaching the ground state

Variational Quantum Eigensolver (VQE) :
a hybrid quantum-classical algorithm

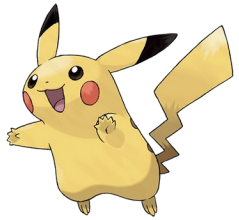


1.4 – Approaching the ground state

Variational Quantum Eigensolver (VQE) :
a hybrid quantum-classical algorithm



1.4 – Approaching the ground state



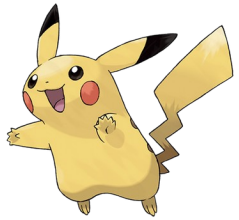
VQE

ADAPT-VQE

The ADAPT-VQE procedure requires :

- A set of well chosen operators $\{ R_i(\theta) \}_{1 \leq i \leq n}$.
- A criterion to select one of them on their ability to reduce the energy of a given state.

1.4 – Approaching the ground state



VQE

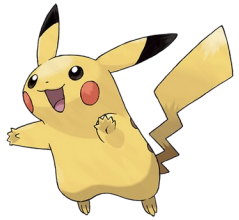
ADAPT-VQE

Input state

The ADAPT-VQE procedure requires :

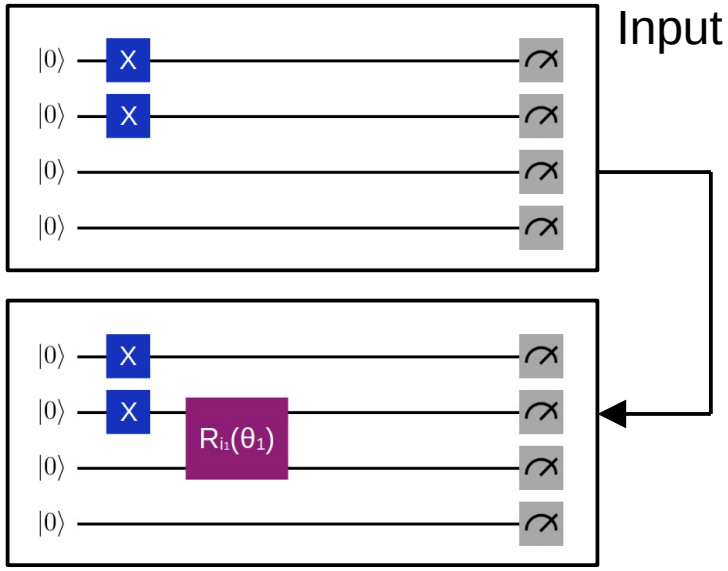
- A set of well chosen operators $\{ R_i(\theta) \}_{1 \leq i \leq n}$.
- A criterion to select one of them on their ability to reduce the energy of a given state.

1.4 – Approaching the ground state



VQE

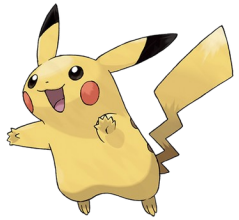
ADAPT-VQE



The ADAPT-VQE procedure requires :

- A set of well chosen operators $\{ R_i(\theta) \}_{1 \leq i \leq n}$.
- A criterion to select one of them on their ability to reduce the energy of a given state.

1.4 – Approaching the ground state

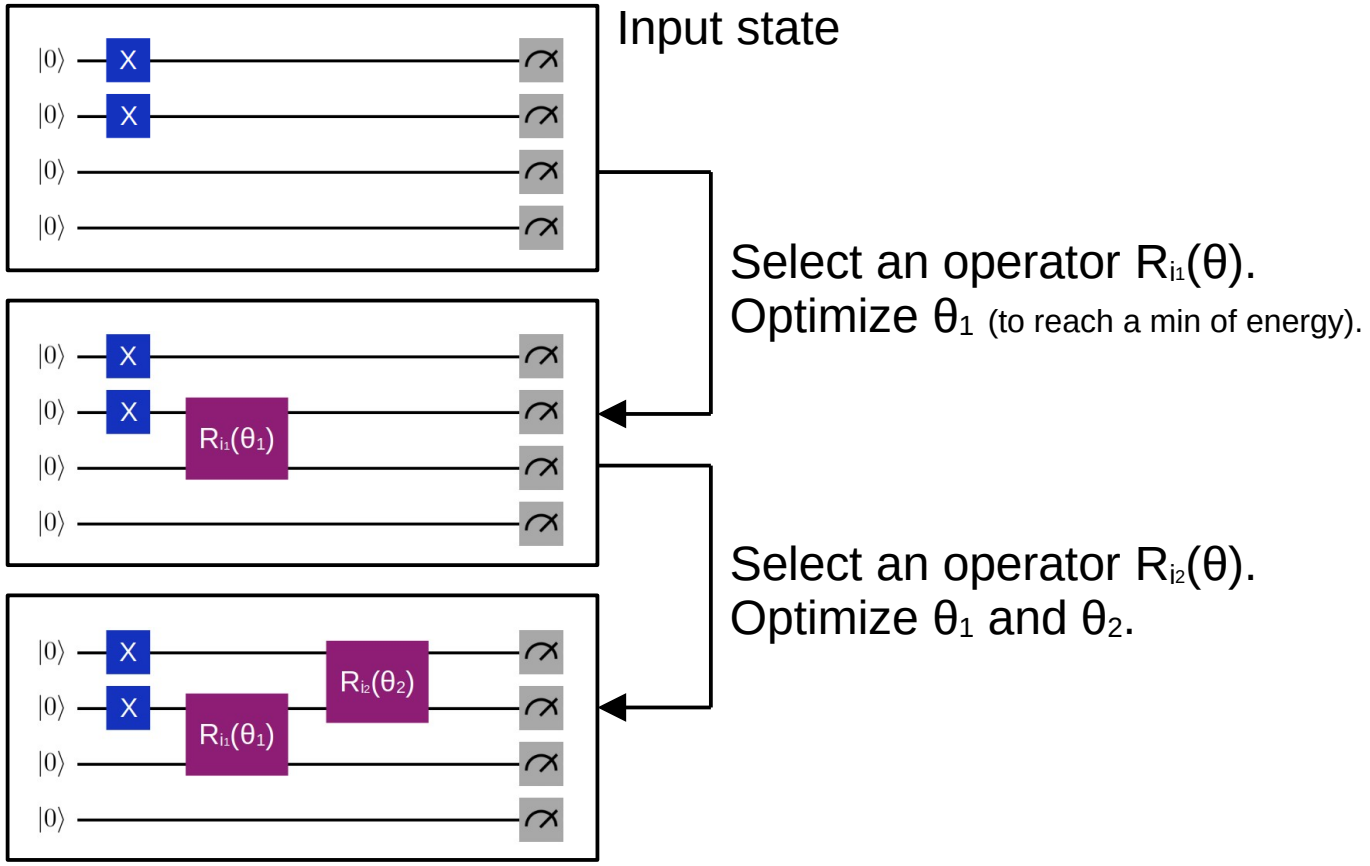


VQE

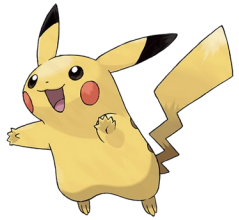
ADAPT-VQE

The ADAPT-VQE procedure requires :

- A set of well chosen operators $\{ R_i(\theta) \}_{1 \leq i \leq n}$.
- A criterion to select one of them on their ability to reduce the energy of a given state.



1.4 – Approaching the ground state

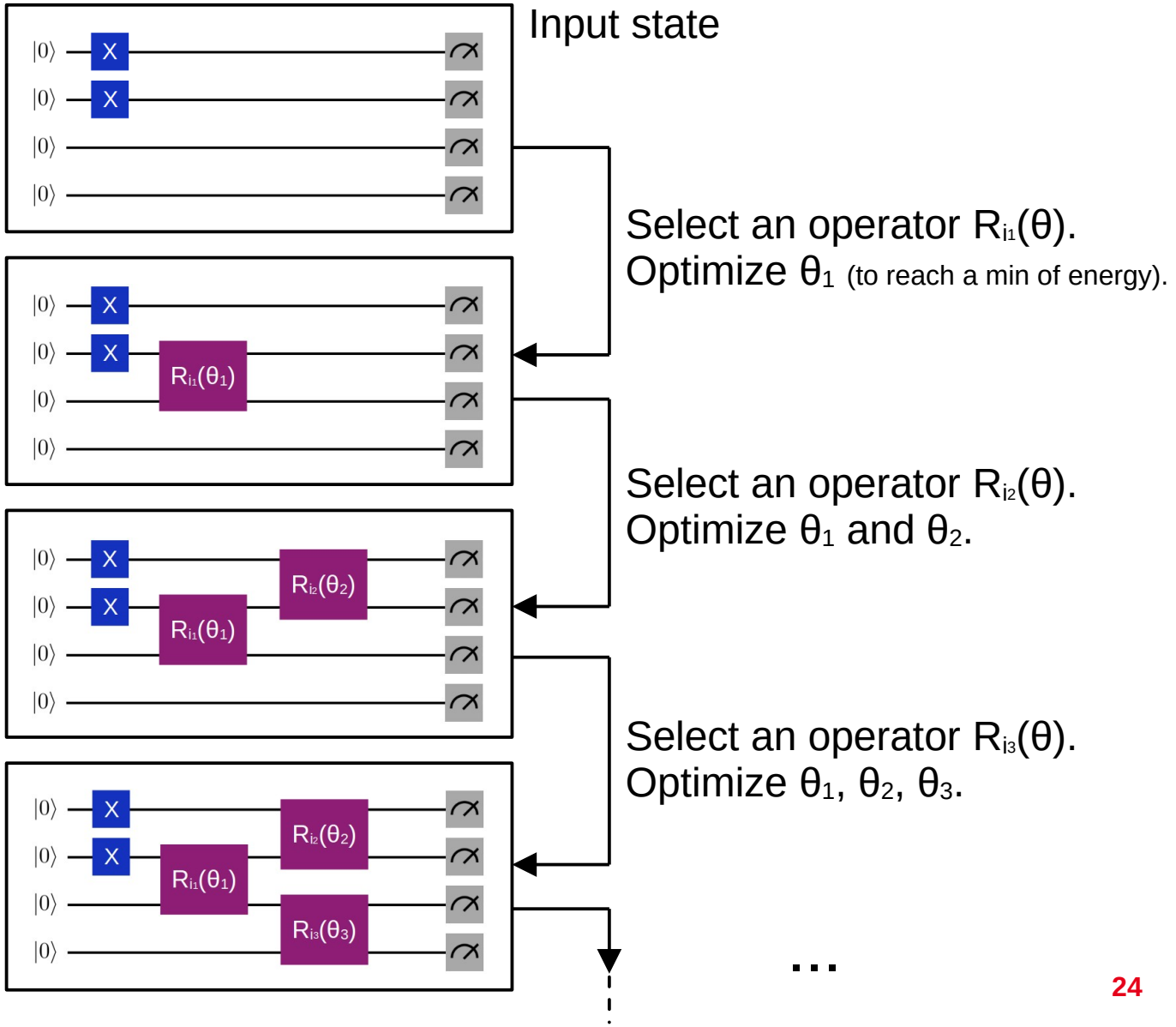


VQE

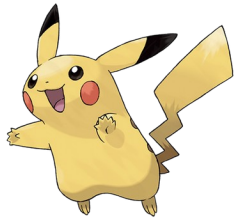
ADAPT-VQE

The ADAPT-VQE procedure requires :

- A set of well chosen operators $\{ R_i(\theta) \}_{1 \leq i \leq n}$.
- A criterion to select one of them on their ability to reduce the energy of a given state.



1.4 – Approaching the ground state

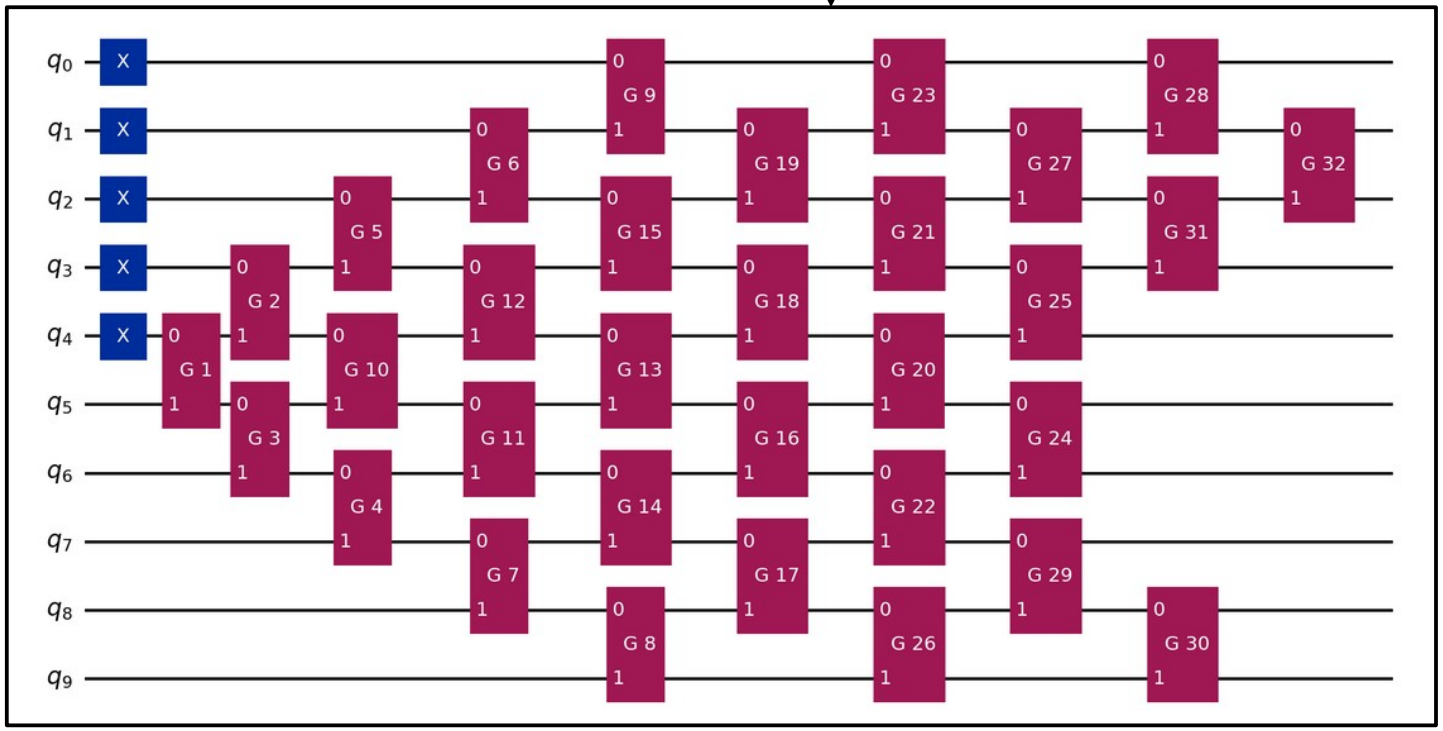


VQE

ADAPT-VQE

The ADAPT-VQE procedure requires :

- A set of well chosen operators $\{ R_i(\theta) \}_{1 \leq i \leq n}$.
- A criterion to select one of them on their ability to reduce the energy of a given state.



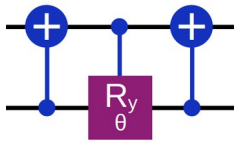
1.4 – Approaching the ground state

3 implementations of ADAPT-VQE

ADAPT-St

Set of operators : $R_{pq}(\theta) = e^{i\frac{\theta}{2}(X_pY_q\text{-}Y_pX_q)}$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{pq}$$

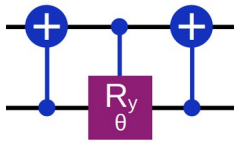


(single qubit excitation-based pool)

ADAPT-Min

Set of operators : $R_{pq}(\theta) = e^{i\frac{\theta}{2}(X_pY_q\text{-}Y_pX_q)}$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{pq}$$

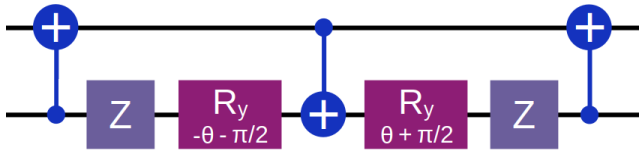


(single qubit excitation-based pool)

ADAPT-Fix

Set of operators :

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) & 0 \\ 0 & \sin(\theta) & -\cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{p,p+1}$$



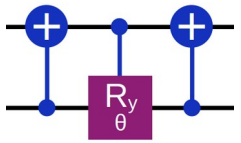
1.4 – Approaching the ground state

3 implementations of ADAPT-VQE

ADAPT-St

Set of operators : $R_{pq}(\theta) = e^{i\frac{\theta}{2}(X_pY_q - Y_pX_q)}$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{pq}$$



(single qubit excitation-based pool)

Selection criterion : maximum absolute gradient at θ = 0, computed from 1 measurement.

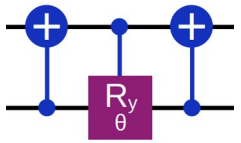
$$\left.\frac{\partial E}{\partial \theta}\right|_{\theta=0} = \langle \Psi | \left[H, \tfrac{1}{2}(X_pY_q - Y_pX_q)\right] | \Psi \rangle$$

See, for example : J. Zhang, D. Lacroix, Y. Beaujeault-Taudière, *Neutron-proton pairing correlations described on quantum computers*, Phys. Rev. C 110, 064320 (2024).

ADAPT-Min

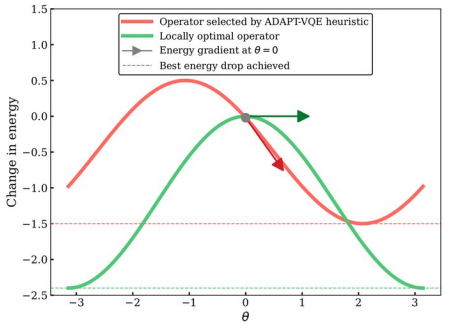
Set of operators : $R_{pq}(\theta) = e^{i\frac{\theta}{2}(X_pY_q - Y_pX_q)}$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{pq}$$



(single qubit excitation-based pool)

Selection criterion : minimum of the energy E(θ), computed from 5 measurements.

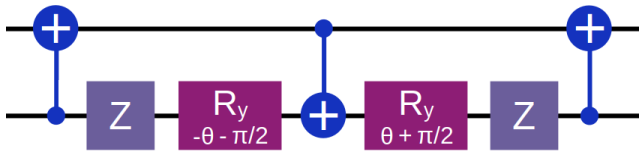


Picture taken from : C. Feniou et al, *Greedy Gradient-free Adaptive Variational Quantum Algorithms on a Noisy Intermediate Scale Quantum Computer*, arXiv:2306.17159 [quant-ph] (2023).

ADAPT-Fix

Set of operators :

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) & 0 \\ 0 & \sin(\theta) & -\cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{p,p+1}$$



Selection criterion : predefined order

Suggested by : B. T. Gard, L. Zhu, G. S. Barron, N. J. Mayhall, S. E. Economou and E. Barnes, *Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm*, npj Quantum Inf. 6, 10 (2020).

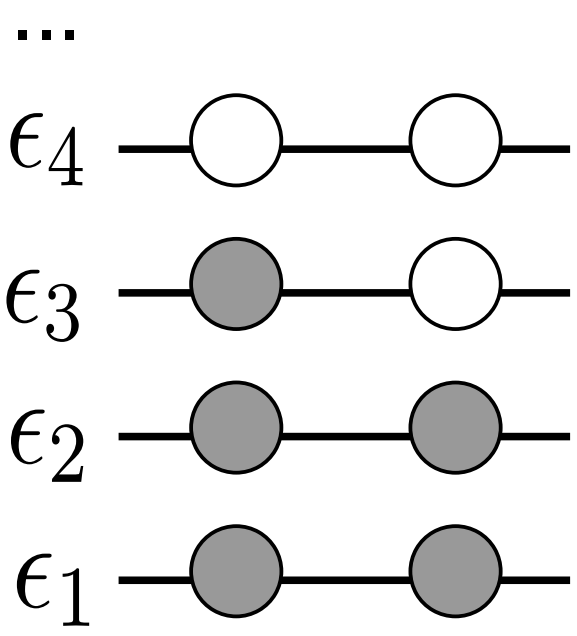
Contents

Contents

2.1 – Pairing model

Pairing Hamiltonian :

$$\hat{H}(g) = \underbrace{\sum_{k=1}^{N} \epsilon_k (\hat{a}_k^\dagger \hat{a}_k + \hat{a}_{\bar{k}}^\dagger \hat{a}_{\bar{k}})}_{\text{Energy of each particle, independently}} \underbrace{- g \sum_{k \neq l} \hat{a}_k^\dagger \hat{a}_{\bar{k}}^\dagger \hat{a}_{\bar{l}} \hat{a}_l}_{\text{Interaction between pairs}}$$



- Models pairing effects in nuclei
- Generates strongly-correlated wave functions
- Exact solutions can be accessed ⟶ benchmarks
- Reference approximation : projected BCS (VAP or PAV)

k particles in N levels :

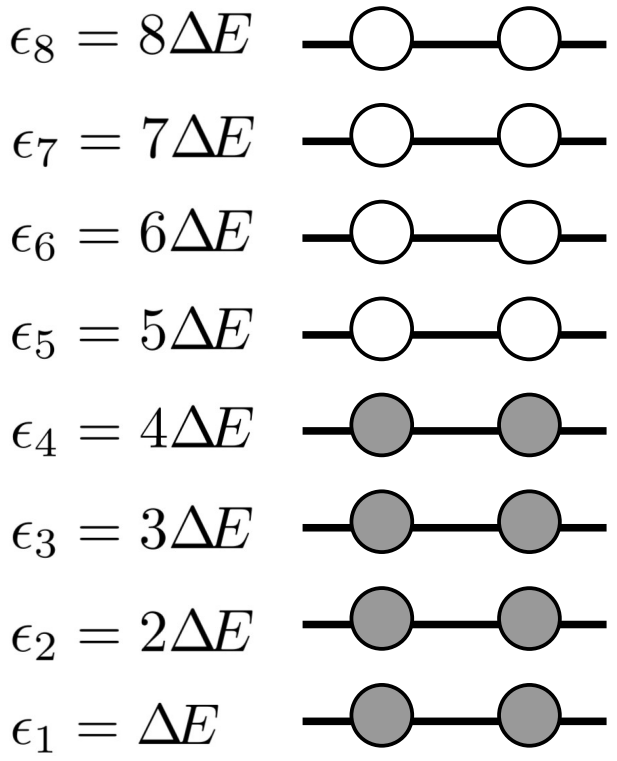
$\binom{2N}{k}$ - dimensional space

2.1 – Pairing model

Pairing Hamiltonian :

$$\hat{H}(g) = \underbrace{\sum_{k=1}^{N} \epsilon_k (\hat{a}_k^\dagger \hat{a}_k + \hat{a}_{\bar{k}}^\dagger \hat{a}_{\bar{k}})}_{\text{Energy of each particle, independently}} \underbrace{- g \sum_{k \neq l} \hat{a}_k^\dagger \hat{a}_{\bar{k}}^\dagger \hat{a}_{\bar{l}} \hat{a}_l}_{\text{Interaction between pairs}}$$

- Models pairing effects in nuclei
- Generates strongly-correlated wave functions
- Exact solutions can be accessed ⟶ benchmarks
- Reference approximation : projected BCS (VAP or PAV)



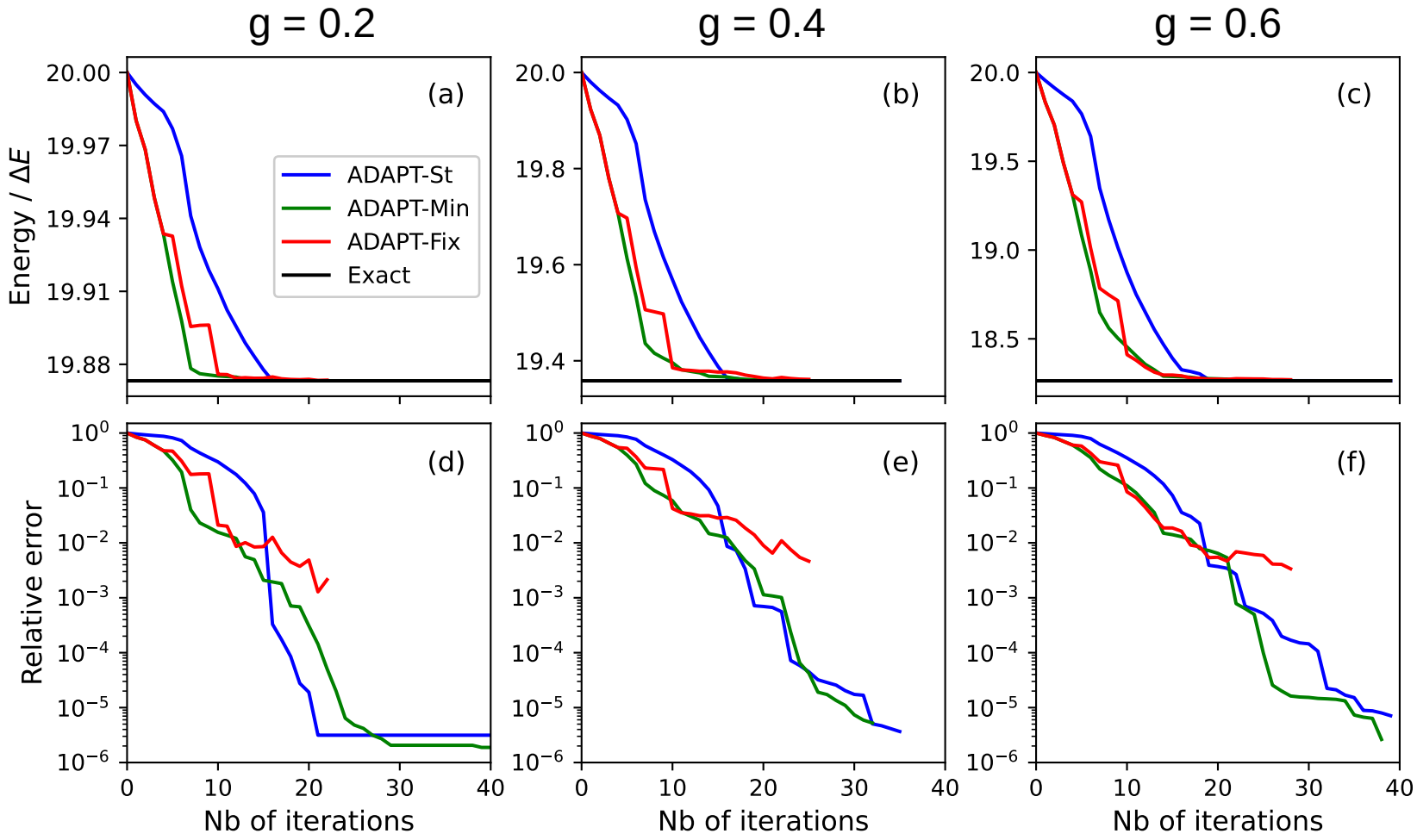
8 particles in 8 levels.

One can go to the subspace with no unpaired particle, which is of dimension

$$\binom{8}{4} = 70$$

Contents

2.2 – Convergence to the ground state



4 pairs in 8 level : $\binom{8}{4}$ – dim Hilbert space

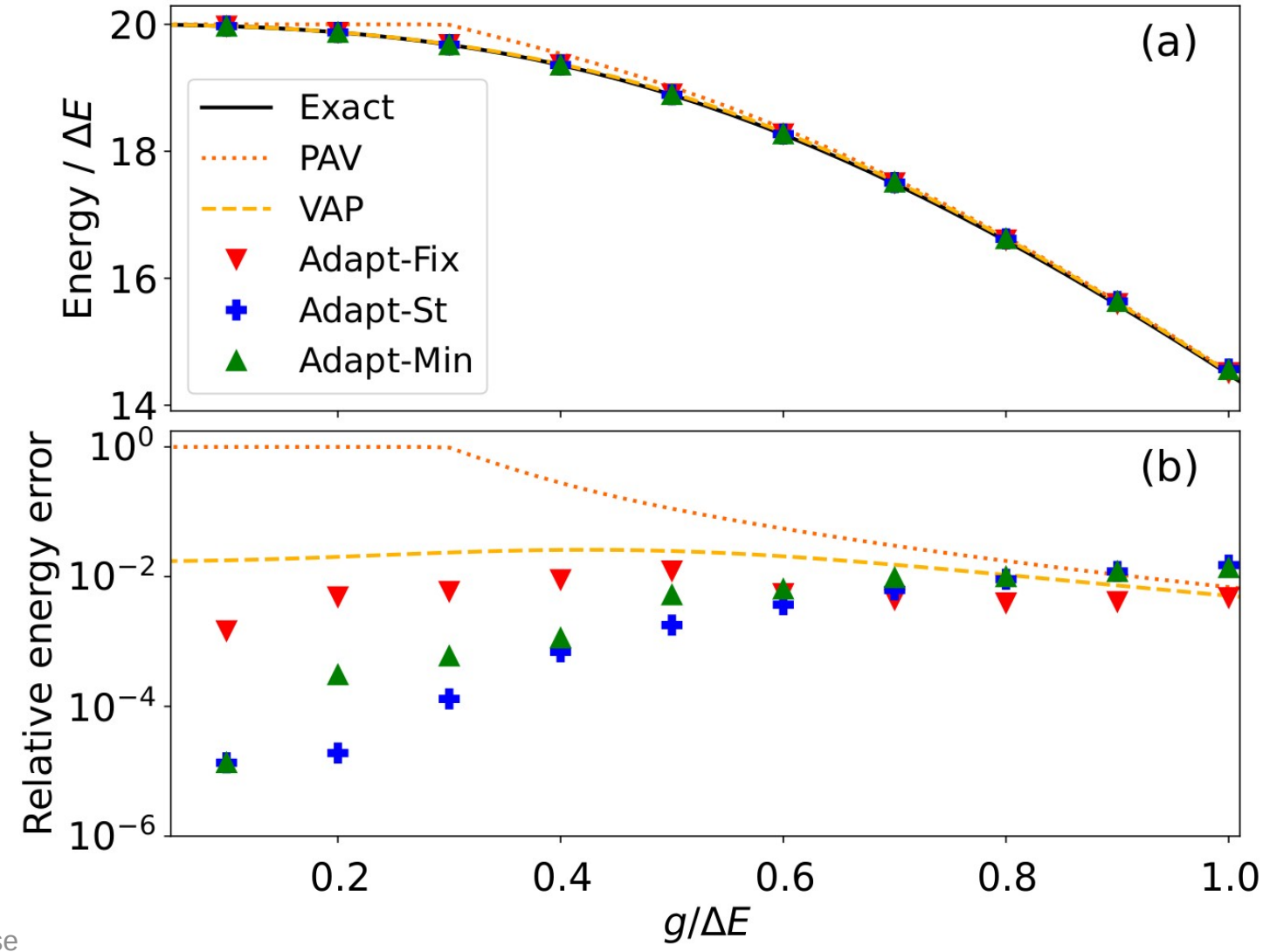
Due to symmetries and normalisation, one need $\binom{8}{4}$ – 1 = 69 real parameters to fully explore the space.

However, **20 parameters are sufficient** to reach the ground state with less than 1% relative error in energy and beat BCS states (PAV and VAP).

(Not obtained from a real quantum computer, but from an emulator)

2.2 – Convergence to the ground state

If one stops at 20 iterations :



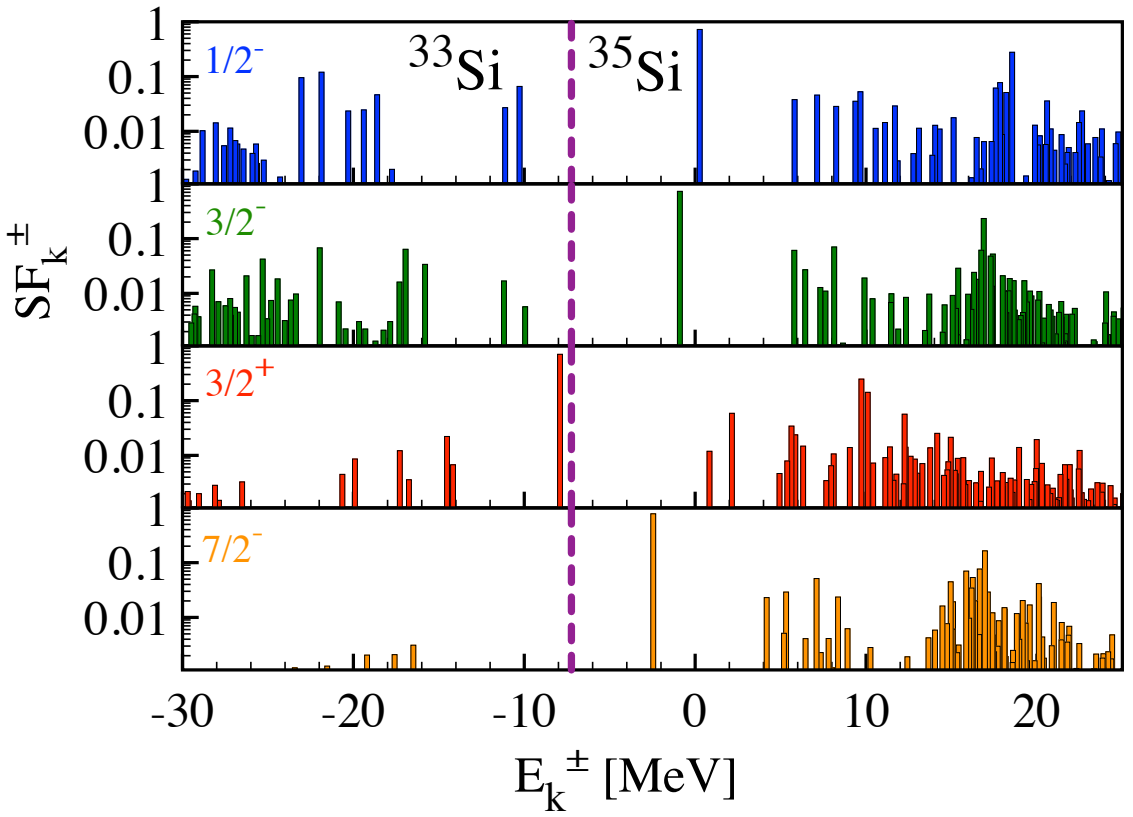
Contents

2.3 – Green's functions

Green's function in Lehmann representation :

$$g_{ab}(\omega) = \sum_k \frac{\langle \Psi_0^A | \hat{a}_a | \Psi_k^{A+1} \rangle \langle \Psi_k^{A+1} | \hat{a}_b^\dagger | \Psi_0^A \rangle}{\omega - (E_k^{A+1} - E_0) + i\eta} + \sum_k \frac{\langle \Psi_0^A | \hat{a}_b^\dagger | \Psi_k^{A-1} \rangle \langle \Psi_k^{A-1} | \hat{a}_a | \Psi_0^A \rangle}{\omega - (E_0 - E_k^{A-1}) - i\eta}$$

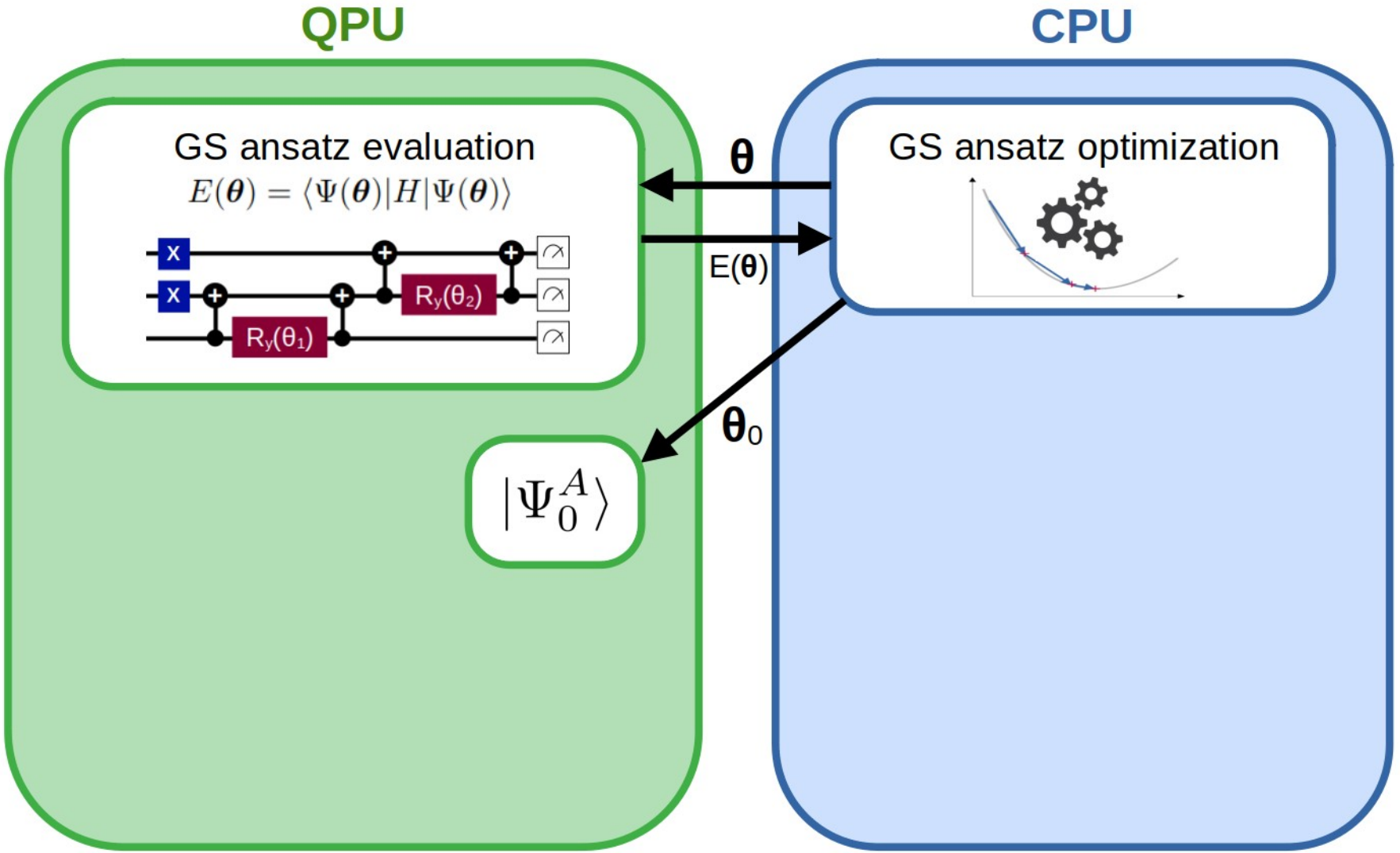
It contains information about excited states in the spaces with A±1 particles.



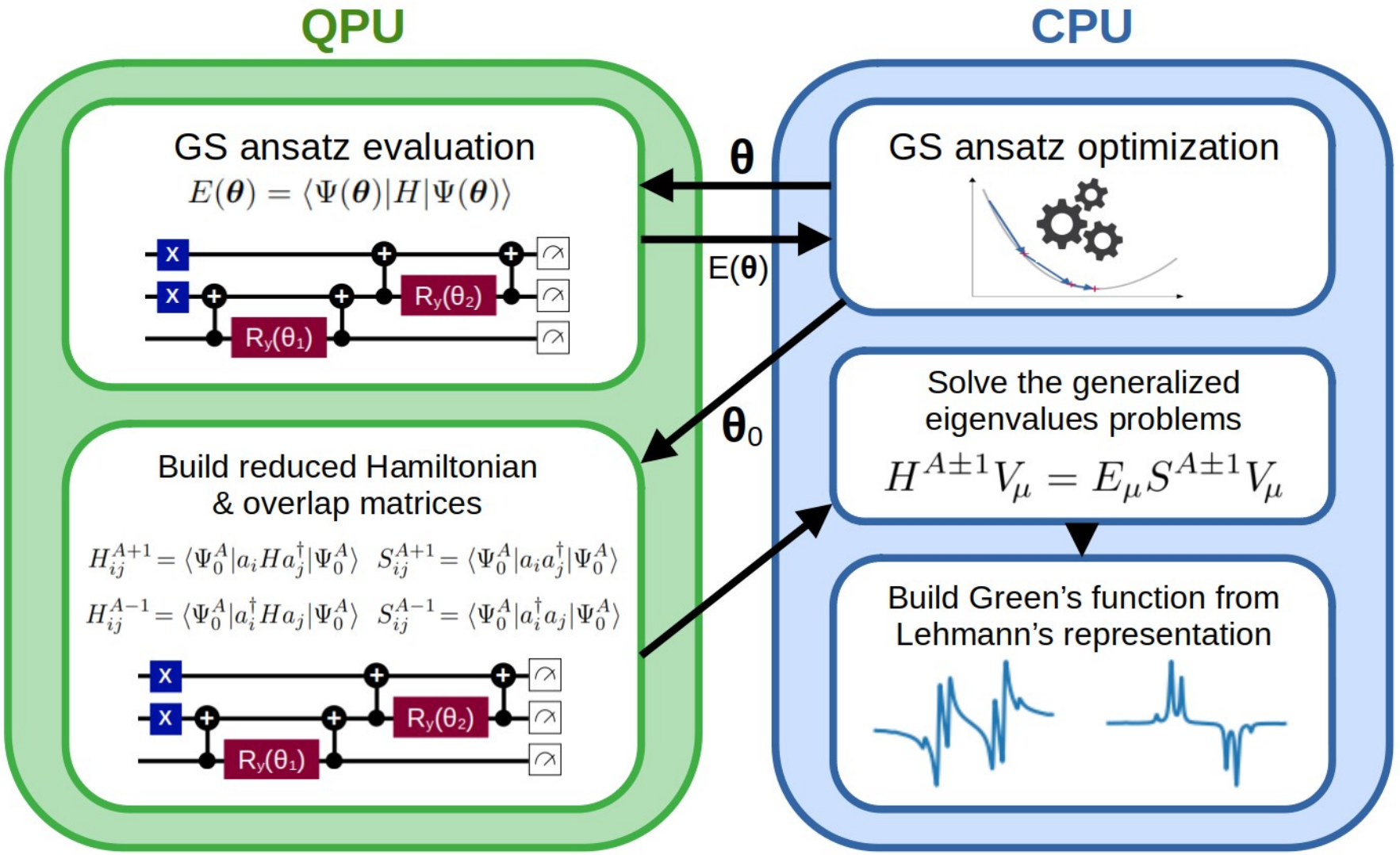
An application of Green's functions : spectral function of ^{34}Si, giving the energies of ^{33}Si and ^{35}Si.

(from V. Somà, *Self-consistent Green's function theory for atomic nuclei*, Front. Phys. 8, 340 (2020)).

2.3 – Green’s functions



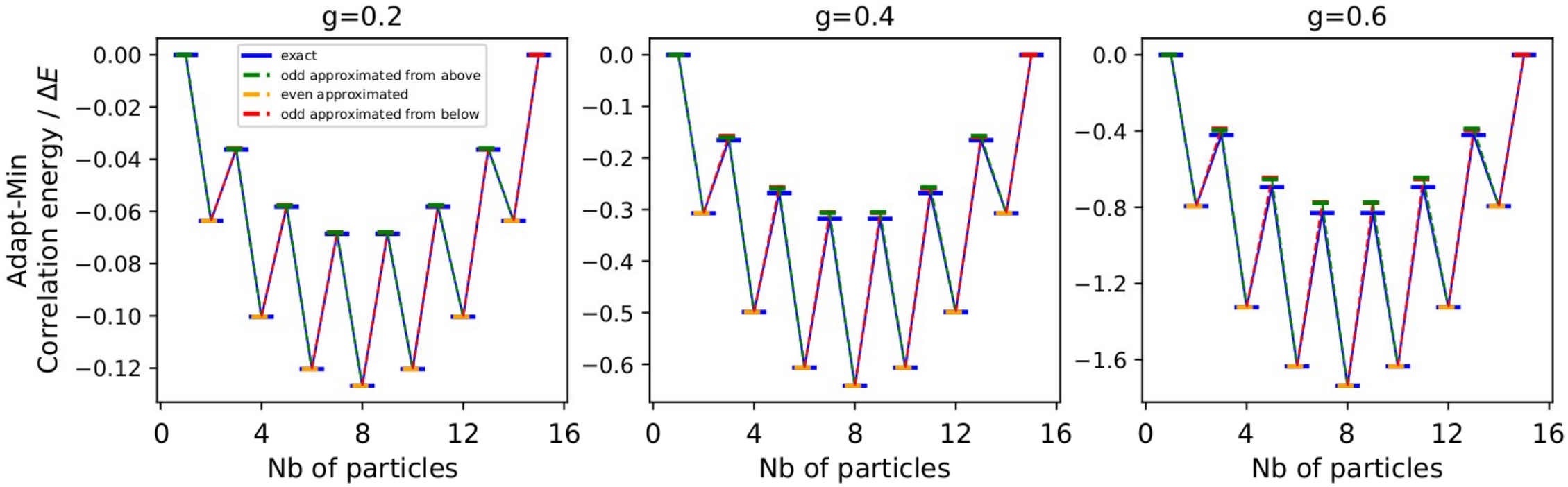
2.3 – Green's functions



Based on a method proposed in :
Diksha Dhawan, Dominika Zgid, Mario Motta, *Quantum algorithm for imaginary-time Green's functions*, Journal of Chemical Theory and Computation 20 (11), 4629 (2024)

2.3 – Green’s functions

Bonus : access the ground state of odd systems

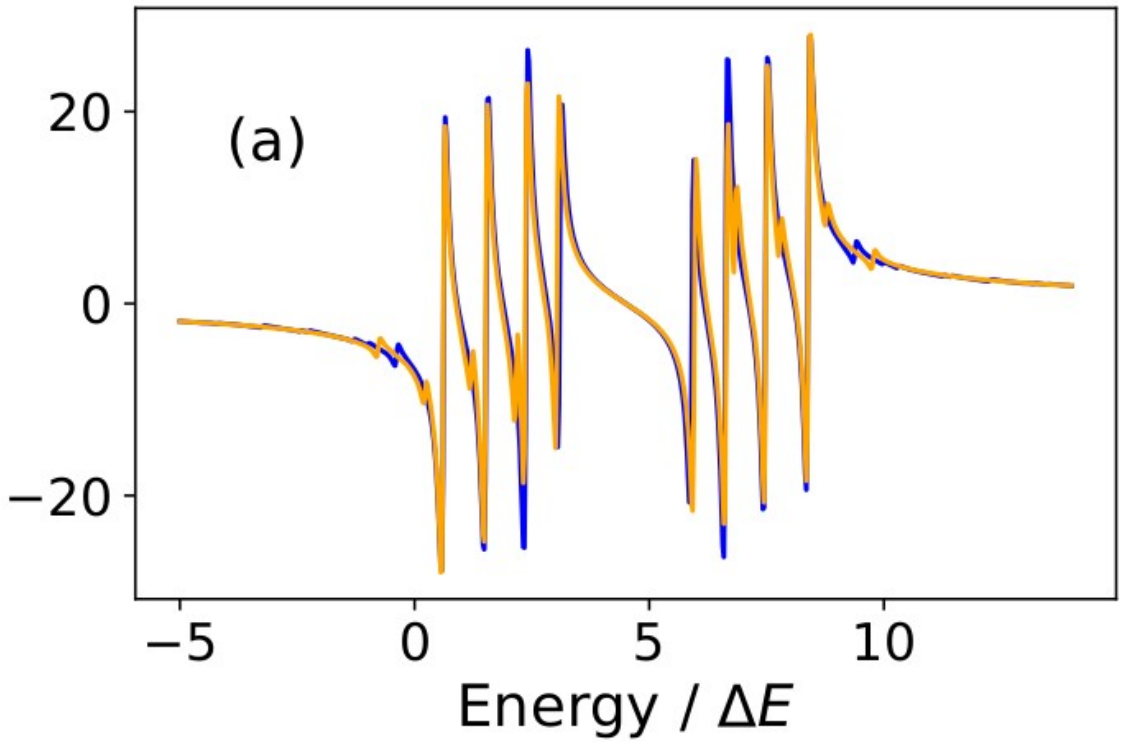


Odd-even staggering well reproduced

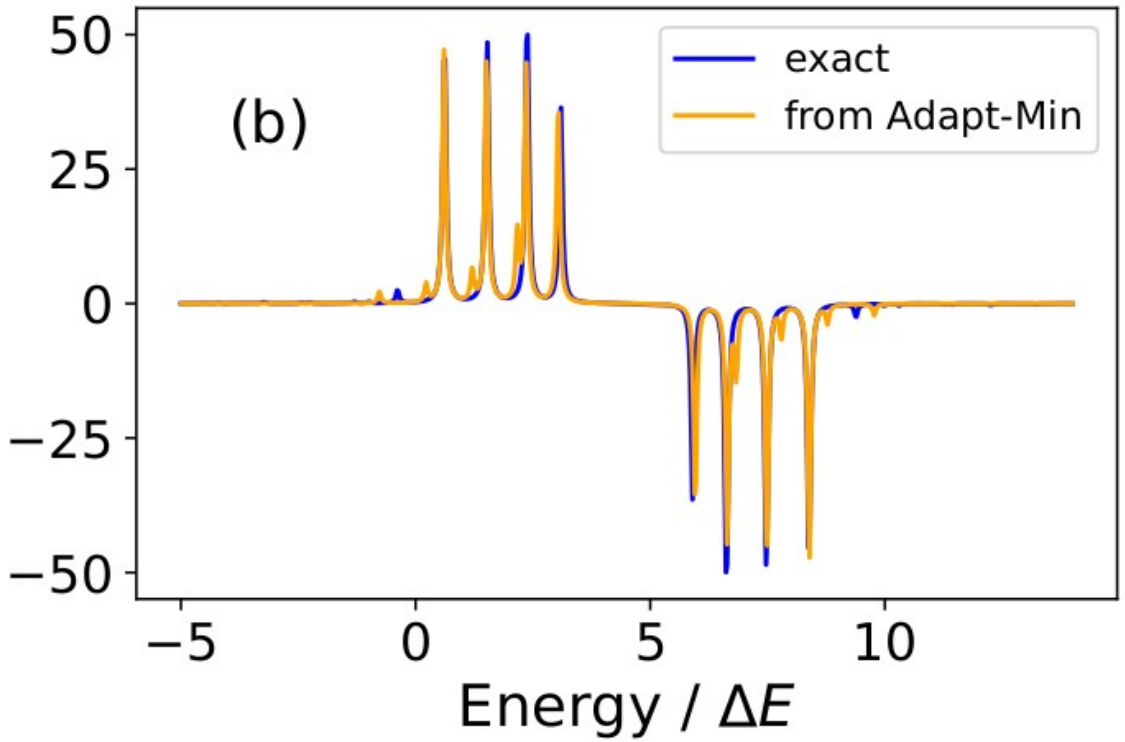
2.3 – Green’s functions

Trace of the Green’s function obtained for a
system of 4 pairs in 8 levels with coupling g = 0,6.

Real part



Imaginary part

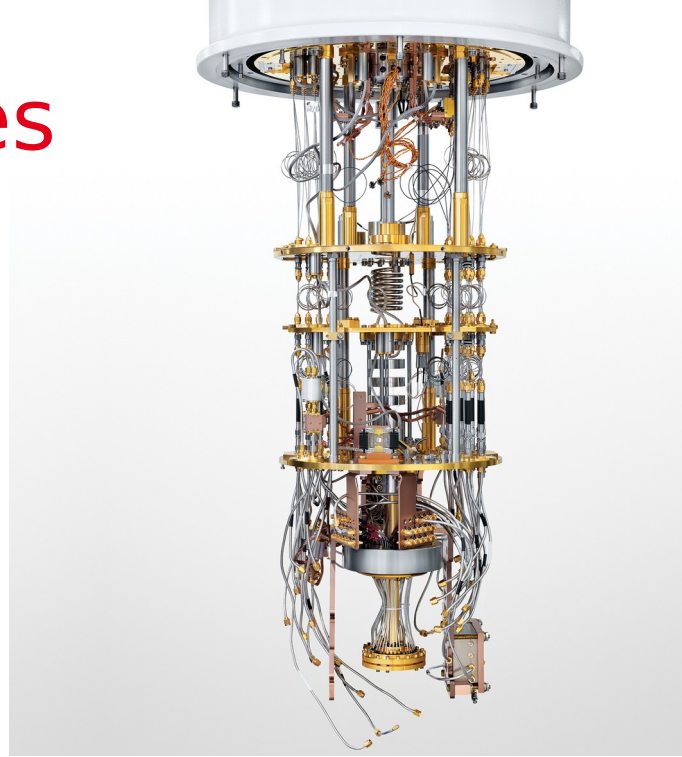


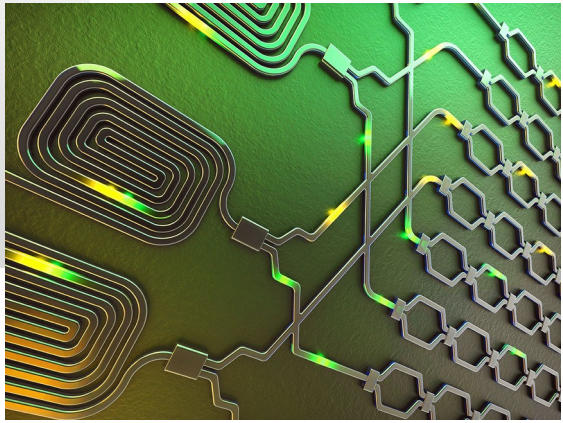
Contents

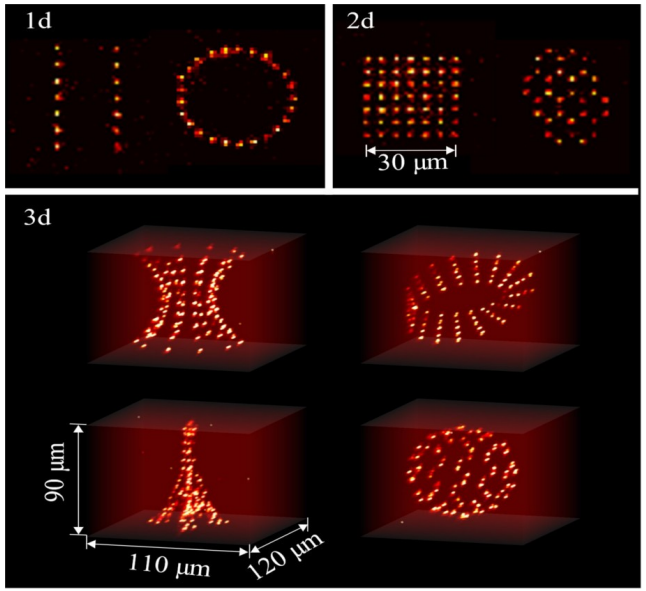
Conclusion and perspectives

Applications on quantum emulator

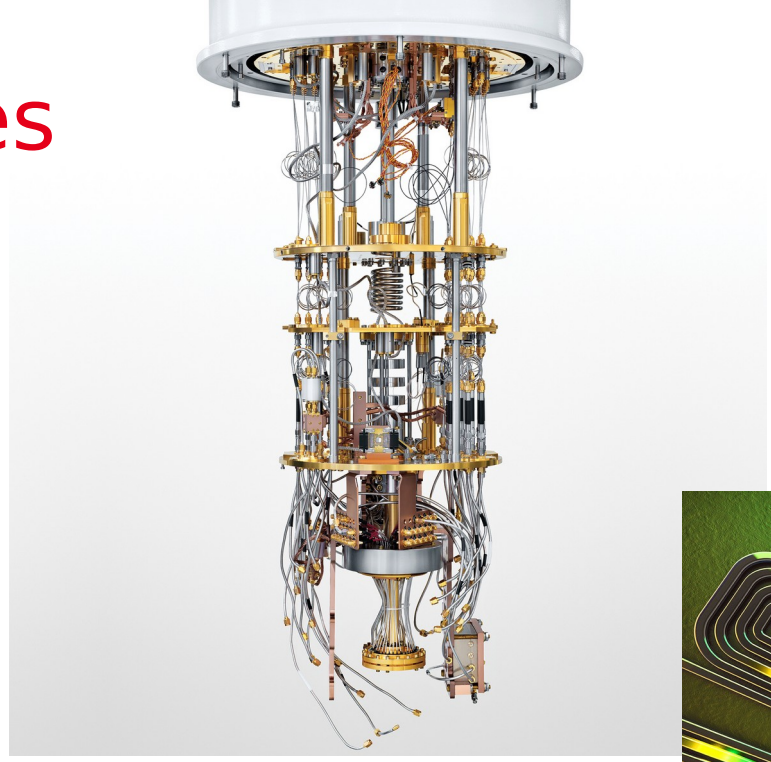
- Quantum algorithm (ADAPT-VQE) achieves good reproduction of ground state.
- Promising results for odd systems and Green's function







Conclusion and perspectives

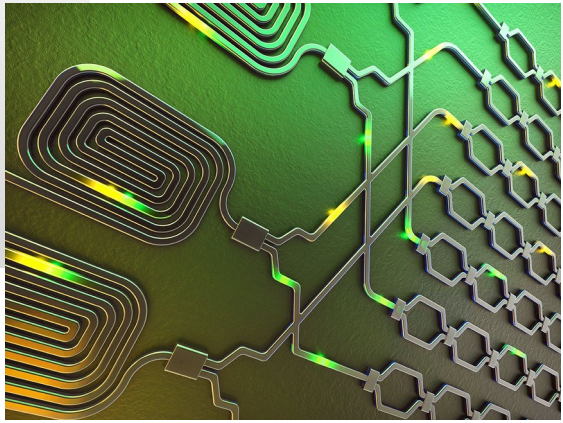


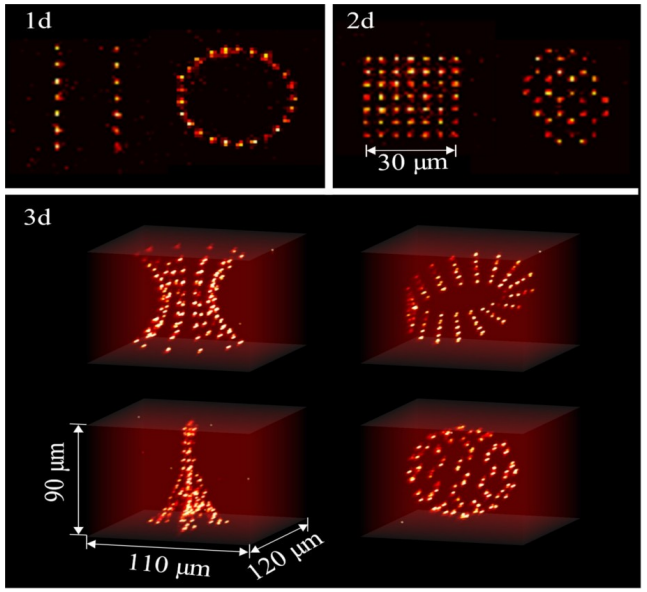
Applications on quantum emulator

- Quantum algorithm (ADAPT-VQE) achieves good reproduction of ground state.
- Promising results for odd systems and Green's function

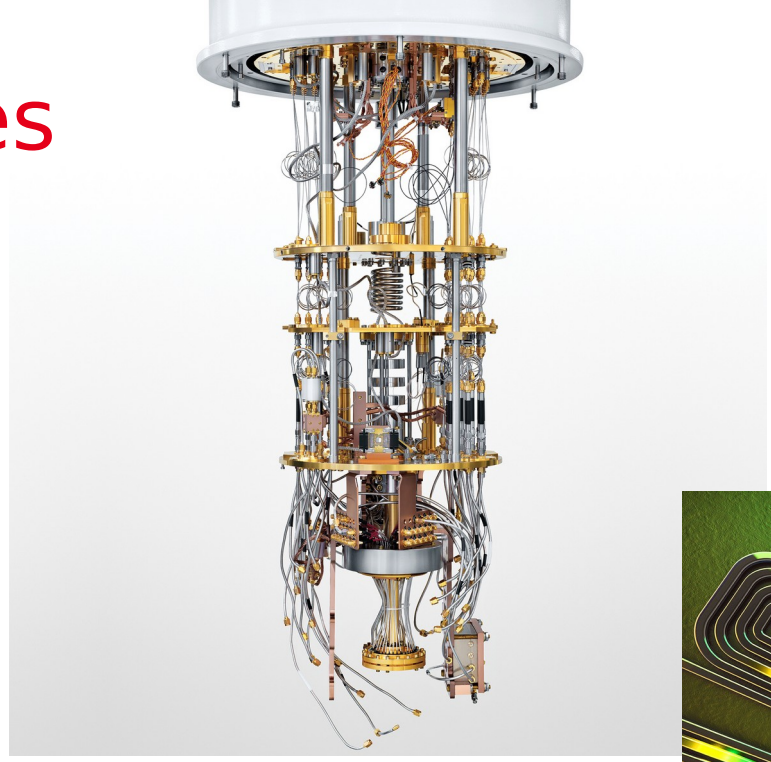
Next : application to real quantum device

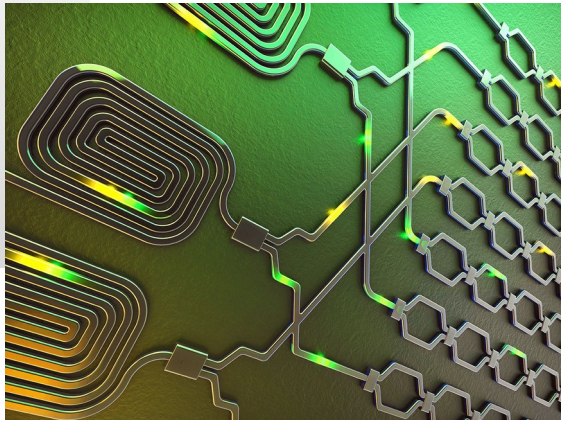
- Pasqual platform (Rydberg atoms) currently being installed at CEA





Conclusion and perspectives



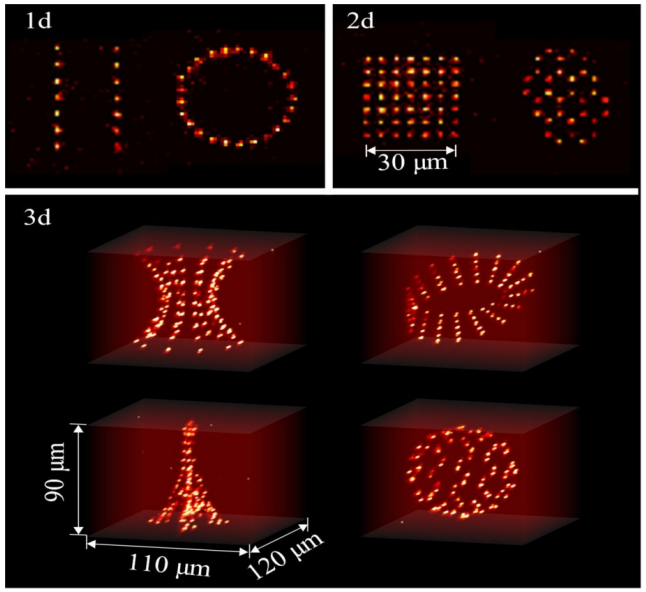


Applications on quantum emulator

- Quantum algorithm (ADAPT-VQE) achieves good reproduction of ground state.
- Promising results for odd systems and Green's function

Next : application to real quantum device

- Pasqual platform (Rydberg atoms) currently being installed at CEA



Status and challenges of quantum computers

- Several types of hardware (superconducting circuits, trapped ions, Rydberg atoms, photons, …)
- Noisy qubits (decoherence, error correction remains hard)
- Emulators are essential to test algorithms

Thanks for your attention

Appendix

What is quantum computing ?

Classical computing

Unit of information : bit

$$b \in \{0, 1\}$$

Operations on a bit :

$$I_d : \begin{matrix} 0 \mapsto 0 \\ 1 \mapsto 1 \end{matrix}$$

$$NOT : \begin{matrix} 0 \mapsto 1 \\ 1 \mapsto 0 \end{matrix}$$

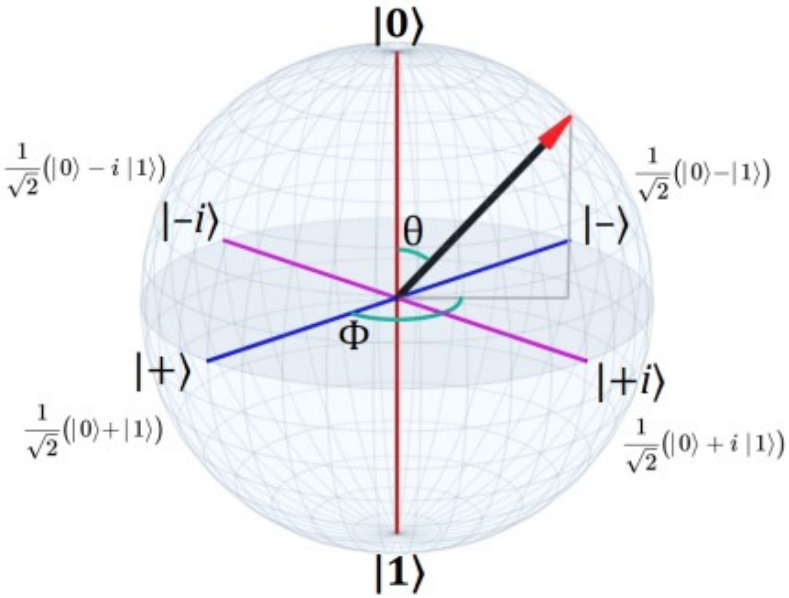
Quantum computing

Unit of information : qubit

$$|q\rangle = \alpha|0\rangle + \beta|1\rangle \quad \text{with } \alpha, \beta \in \mathbb{C}$$

Normalisation
Definition up to a global phase

$$|q\rangle = \cos(\tfrac{\theta}{2})|0\rangle + e^{i\phi}\sin(\tfrac{\theta}{2})|1\rangle \quad \text{with } \theta \in [0, \pi] \text{ and } \phi \in [0, 2\pi[$$

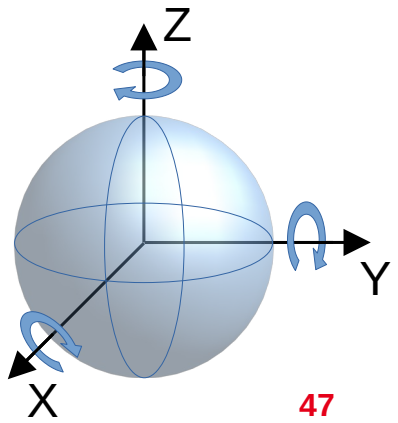


Operations on a qubit :

$$R_x(\theta) = \begin{pmatrix} \cos(\frac{\theta}{2}) & -i\sin(\frac{\theta}{2}) \\ -i\sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{pmatrix} \qquad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$ (→ corresponds to NOT gate)

$$R_y(\theta) = \begin{pmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{pmatrix} \qquad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$R_z(\theta) = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \qquad \dots$$



What is quantum computing ?

Classical computing

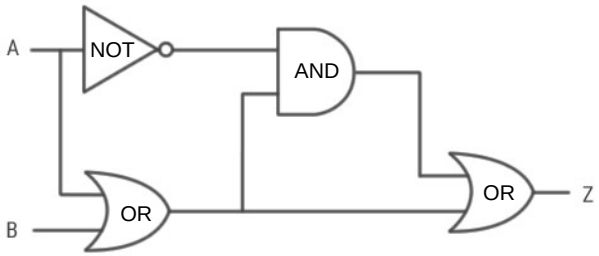
State of n bits :

$$\mathbf{b} = \{b_1, b_2, ..., b_n\} \in \{0,1\}^n$$

Operations on n bits :

$$\text{All } f : \{0,1\}^n \rightarrow \{0,1\}^n$$

Logical circuit :



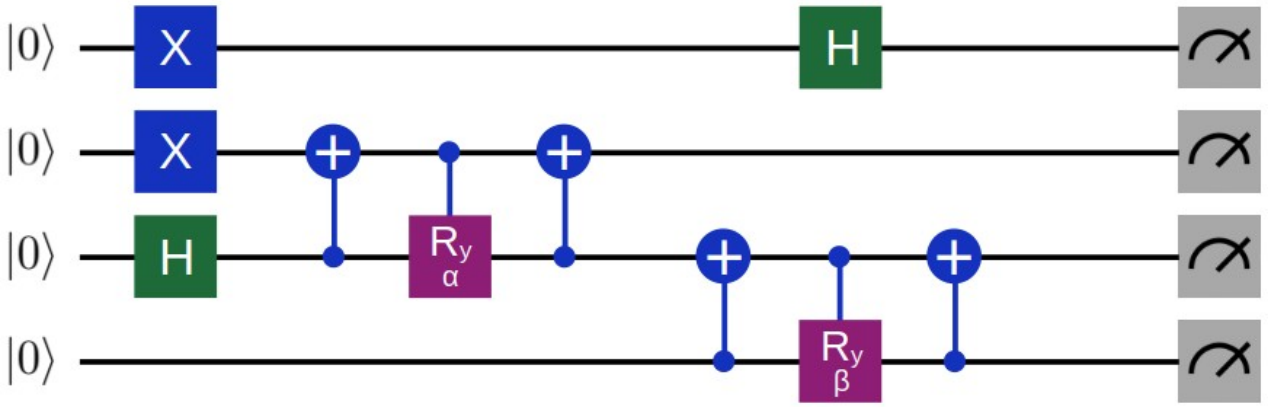
Quantum computing

State of n qubits :

$$|\Psi\rangle = c_1|00...0\rangle + c_2|00...1\rangle + ... + c_{2^n}|11...1\rangle$$

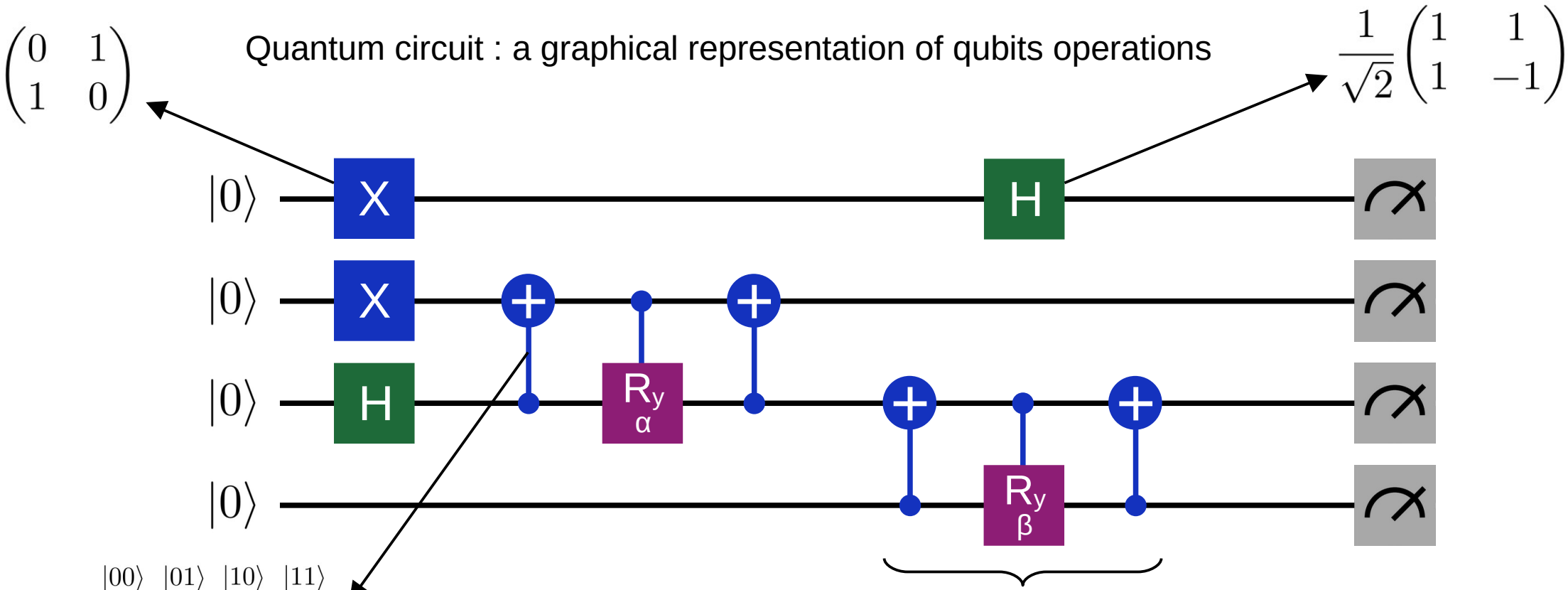
Operations on n qubits : unitary operators

Quantum circuit :



What is quantum computing ?

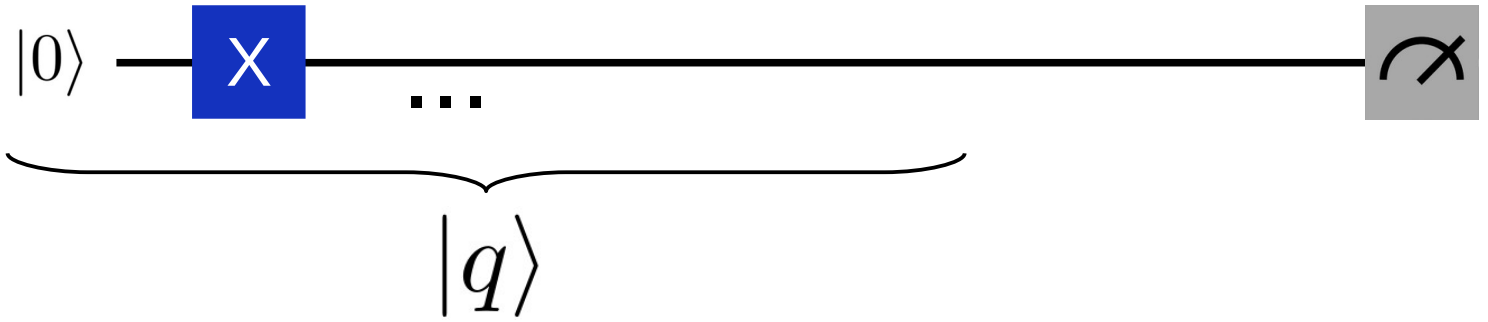
Quantum circuit : a graphical representation of qubits operations



$$\begin{matrix} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{matrix}$$
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

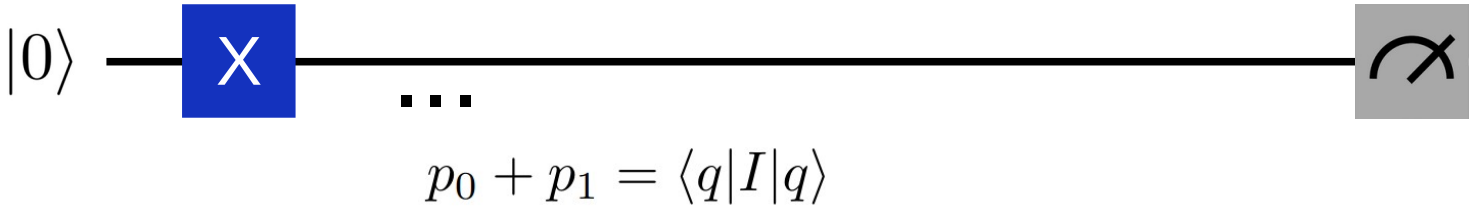
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos(\theta) & -\sin(\theta) \\ 0 & 0 & \sin(\theta) & \cos(\theta) \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Measuring expectation values



$$p_0 = |\langle 0|q\rangle|^2$$
$$p_1 = |\langle 1|q\rangle|^2$$

Measuring expectation values



$$p_0 + p_1 = \langle q|I|q\rangle$$

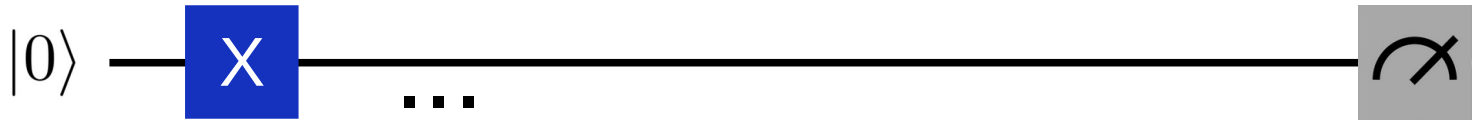
$$p_0 = |\langle 0|q\rangle|^2$$
$$p_1 = |\langle 1|q\rangle|^2$$

Measuring expectation values

$$p_0 = |\langle 0|q\rangle|^2$$
$$p_1 = |\langle 1|q\rangle|^2$$

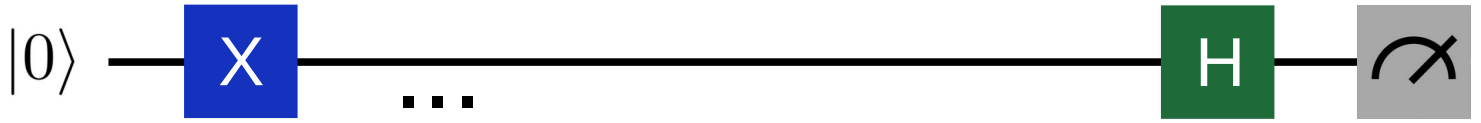
$$\begin{aligned}
p_0 + p_1 &= \langle q|I|q\rangle \\
p_0 - p_1 &= \langle q|0\rangle\langle 0|q\rangle - \langle q|1\rangle\langle 1|q\rangle \\
&= \langle q|\Big(|0\rangle\langle 0| - |1\rangle\langle 1|\Big)|q\rangle \\
&= \langle q| \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} |q\rangle \\
&= \langle q|Z|q\rangle
\end{aligned}$$

Measuring expectation values



$$p_0 + p_1 = \langle q|I|q\rangle$$
$$p_0 - p_1 = \langle q|Z|q\rangle$$

$$p_0 = |\langle 0|q\rangle|^2$$
$$p_1 = |\langle 1|q\rangle|^2$$



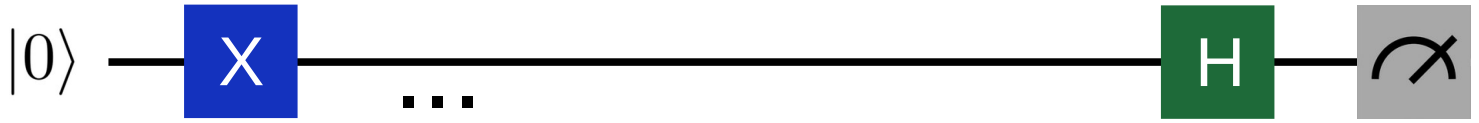
$$p_0 - p_1 = \langle q|H^\dagger Z H|q\rangle = \langle q|X|q\rangle$$

$$p_0 = |\langle 0|H|q\rangle|^2$$
$$p_1 = |\langle 1|H|q\rangle|^2$$

Measuring expectation values

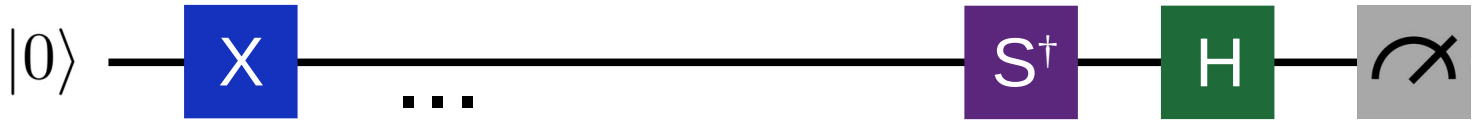
$$p_0 + p_1 = \langle q|I|q\rangle$$
$$p_0 - p_1 = \langle q|Z|q\rangle$$

$$p_0 = |\langle 0|q\rangle|^2$$
$$p_1 = |\langle 1|q\rangle|^2$$



$$p_0 - p_1 = \langle q|H^\dagger Z H|q\rangle = \langle q|X|q\rangle$$

$$p_0 = |\langle 0|H|q\rangle|^2$$
$$p_1 = |\langle 1|H|q\rangle|^2$$



$$p_0 - p_1 = \langle q|SH^\dagger Z H S^\dagger|q\rangle = \langle q|Y|q\rangle$$

$$p_0 = |\langle 0|HS^\dagger|q\rangle|^2$$
$$p_1 = |\langle 1|HS^\dagger|q\rangle|^2$$