Introduction to Quantum Computing
Lecture 22: Variational Quantum Algorithms II

Petros Wallden

School of Informatics, University of Edinburgh

4th November 2025

This Lecture

1. Step 2: How to Measure the Energy/Cost
2. Step 3: The Ansatz (Family of Quantum States)
3. Step 4: Classical Optimisation & VQA Summary

Step 2: How to Measure the Energy/Cost

The Mathematical Task

Given a Hermitian matrix $\mathcal{H}$ (typically called Hamiltonian), compute its smallest eigenvalue (called "ground state energy")

Previously in VQA

The Mathematical Task

Given a Hermitian matrix $\mathcal{H}$ (typically called Hamiltonian), compute its smallest eigenvalue (called "ground state energy")

Why is it relevant?

- Can solve it using VQA that is suitable for NISQ devices
- k-local Hamiltonian problem is QMA-complete

Step 1 Hamiltonian Encoding (Previous Lecture)

Express your desired problem as the ground state of a suitable qubit-Hamiltonian $\mathcal{H}$

Previously in VQA: the four steps

Step 1 Hamiltonian Encoding (Previous Lecture)

Express your desired problem as the ground state of a suitable qubit-Hamiltonian $\mathcal{H}$

Step 2 Energy estimation (the only quantum part)

Given copies of a state $|\psi\rangle$, estimate its energy $\langle\psi|\,\mathcal{H}\,|\psi\rangle$

Step 3 Choice of Ansatz

A family of parametrised quantum states $|\psi(\vec{\theta})\rangle$ where one of its members approximates best the ground state

Step 4 Classical optimiser

A classical optimiser that finds the values $\vec{\theta^*}$ that minimise the cost $C(\vec{\theta}) := \langle\psi(\vec{\theta})|\,\mathcal{H}\,|\psi(\vec{\theta})\rangle$, ie $\vec{\theta^*} := \arg\min_{\vec{\theta}} C(\vec{\theta})$

Step 2: Energy Estimation

Given:

- An efficient, NISQ compatible, description to generate an n-qubit quantum state $|\psi\rangle$
- An n-qubit Hamiltonian $\mathcal{H}$

Step 2: Energy Estimation

Given:

- An efficient, NISQ compatible, description to generate an n-qubit quantum state $|\psi\rangle$
- An n-qubit Hamiltonian $\mathcal{H}$

Task: Estimate the energy $E(\psi) := \langle\psi| \mathcal{H} |\psi\rangle$

Given:

- An efficient, NISQ compatible, description to generate an n-qubit quantum state $|\psi\rangle$
- An n-qubit Hamiltonian $\mathcal{H}$

Task: Estimate the energy $E(\psi) := \langle\psi| \mathcal{H} |\psi\rangle$

1. Decompose the Hamiltonian to sum of Pauli observables
2. Generate multiple copies of $|\psi\rangle$
3. Measure each Pauli suff times to get desired accuracy
4. Combine above to get an estimate for the energy $E(\psi)$

Decompose Hamiltonian to local Pauli observables

- Pauli observables can be measured **locally** and **easily**
- Frequently the Hamiltonian is already as sum of Pauli's

  E.g. Ising Hamiltonians

  $\mathcal{H} = -\sum_{(i,j)} J_{ij} Z_i \otimes Z_j - \mu \sum_i h_i Z_i$

- Other decompositions of the Hamiltonian to simple local obervables can and have been considered (not here)

Decompose Hamiltonian to local Pauli observables

- Any n-qubit Hermitian operator can be written as sum of products of Pauli matrices $P_i \in \{I, X, Y, Z\}$
(is orthonormal basis – Pauli observ: $\{+1, -1\}$ eigenvalues)

$$\mathcal{H} = \sum c_{i_1,\cdots,i_n} P_1^{i_1} \otimes \cdots \otimes P_n^{i_n}$$

Decompose Hamiltonian to local Pauli observables

- Any n-qubit Hermitian operator can be written as sum of products of Pauli matrices $P_i \in \{I, X, Y, Z\}$ (is orthonormal basis – Pauli observ: $\{+1, -1\}$ eigenvalues)

$$\mathcal{H} = \sum c_{i_1,\cdots,i_n} P_1^{i_1} \otimes \cdots \otimes P_n^{i_n}$$

- Practically, in many cases it is given in this form or in a similar form where one needs to decompose some fixed two-qubit gates ($\wedge X, \wedge Z,$ etc) in Pauli's

Decompose Hamiltonian to local Pauli observables

- Any n-qubit Hermitian operator can be written as sum of products of Pauli matrices $P_i \in \{I, X, Y, Z\}$ (is orthonormal basis – Pauli observ: $\{+1, -1\}$ eigenvalues)

$$\mathcal{H} = \sum c_{i_1,\cdots,i_n} P_1^{i_1} \otimes \cdots \otimes P_n^{i_n}$$

- Practically, in many cases it is given in this form or in a similar form where one needs to decompose some fixed two-qubit gates ($\wedge X, \wedge Z,$ etc) in Pauli's

- To compute coefficients $c_{i_1,\cdots,i_n}$ use the inner product

$$\langle A, B \rangle := \frac{\mathrm{Tr}(A^\dagger B)}{2^n}$$

$$c_{i_1,\cdots,i_n} = \langle P_1^{i_1} \otimes \cdots \otimes P_n^{i_n}, \mathcal{H} \rangle$$

Decompose Hamiltonian to local Pauli observables

Example:

Decompose $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ in Pauli's

$$H = \alpha I + \beta X + \gamma Y + \delta Z$$

Decompose Hamiltonian to local Pauli observables

Example:

Decompose $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ in Pauli's

$$H = \alpha I + \beta X + \gamma Y + \delta Z$$

$$\langle I, H \rangle = \alpha = 0 \quad ; \quad \langle X, H \rangle = \beta = \sqrt{2}/2$$
$$\langle Y, H \rangle = \gamma = 0 \quad ; \quad \langle Z, H \rangle = \delta = \sqrt{2}/2$$

Decompose Hamiltonian to local Pauli observables

Example:

Decompose $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ in Pauli's

$$H = \alpha I + \beta X + \gamma Y + \delta Z$$

$$\langle I, H\rangle = \alpha = 0 \quad ; \quad \langle X, H\rangle = \beta = \sqrt{2}/2$$
$$\langle Y, H\rangle = \gamma = 0 \quad ; \quad \langle Z, H\rangle = \delta = \sqrt{2}/2$$

As expected

$$H = \frac{1}{\sqrt{2}}(X + Z) = \frac{1}{\sqrt{2}}\left(\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\right)$$

Decompose Hamiltonian to local Pauli observables

Example:

Decompose $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ in Pauli's

$$H = \alpha I + \beta X + \gamma Y + \delta Z$$

$$\langle I, H\rangle = \alpha = 0 \quad ; \quad \langle X, H\rangle = \beta = \sqrt{2}/2$$
$$\langle Y, H\rangle = \gamma = 0 \quad ; \quad \langle Z, H\rangle = \delta = \sqrt{2}/2$$

As expected

$$H = \frac{1}{\sqrt{2}}(X + Z) = \frac{1}{\sqrt{2}}\left(\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\right)$$

- Other example (check)

$$\wedge Z = \frac{1}{2}(I \otimes I + I \otimes Z + Z \otimes I - Z \otimes Z)$$

Estimating the value of Pauli observables $\langle\psi| P |\psi\rangle$

- Prepare-and-measure the state N times
- Each outcome gives a value $O_i \in \{+1, -1\}$
- Output $\langle\mathcal{O}\rangle = \sum_i O_i/N$ for the value of the observable

Estimating the value of Pauli observables $\langle\psi| P |\psi\rangle$

- Prepare-and-measure the state N times
- Each outcome gives a value $O_i \in \{+1, -1\}$
- Output $\langle\mathcal{O}\rangle = \sum_i O_i/N$ for the value of the observable
- If we require accuracy of our estimate ϵ with δ confidence:

$$\boxed{N \approx O(\frac{1}{\epsilon^2}\log\left(\frac{1}{\delta}\right))}$$

Estimating the value of Pauli observables $\langle\psi| P |\psi\rangle$

- Prepare-and-measure the state N times
- Each outcome gives a value $O_i \in \{+1, -1\}$
- Output $\langle\mathcal{O}\rangle = \sum_i O_i/N$ for the value of the observable
- If we require accuracy of our estimate ϵ with δ confidence:

$$\boxed{N \approx O(\frac{1}{\epsilon^2} \log\left(\frac{1}{\delta}\right))}$$

- By Hoeffding (and Chernoff) inequalities we know:

$$Pr(|\bar{O} - \langle O\rangle| \geq \epsilon) \leq e^{-N\epsilon^2}$$

Probability that the true expectation differs by ϵ or more from the measured one

- If we require that this probability is also bounded by $\delta = e^{-N\epsilon^2}$ our confidence, we get above expression

Implications of the accuracy

- Note that the **resources** required **depend on the problem**
- NP-**complete** problems (or even worse QMA-complete) **cannot be solved in poly-time** with a quantum computer
- Problems outside BQP have negligible “energy gap”, i.e. the ground state differs from the next eigenvalue (1st excited state) by a very small amount
- To achieve **accuracy that distinguishes** between the two, one needs **super-polynomial repetitions**

Implications of the accuracy

- Note that the **resources** required **depend on the problem**
- NP-**complete** problems (or even worse QMA-complete) **cannot be solved in poly-time** with a quantum computer
- Problems outside BQP have negligible "energy gap", i.e. the ground state differs from the next eigenvalue (1st excited state) by a very small amount
- To achieve **accuracy that distinguishes** between the two, one needs **super-polynomial repetitions**
- **High accuracy** is also required:

- To move in a hyper-parametrised space

  (where gradients are negligible)

- To overcome the effects of noise and determine truly the direction in the parameter space to move.

Computing the Energy $E(\psi)$

- Using the Pauli decomposition we have:

  $$E(\psi) = \langle\psi|\,\mathcal{H}\,|\psi\rangle = \sum c_{i_1,\cdots,i_n}\langle\psi|\,P_1^{i_1}\otimes\cdots\otimes P_n^{i_n}\,|\psi\rangle$$

  where $\langle\psi|\,P_1^{i_1}\otimes\cdots\otimes P_n^{i_n}\,|\psi\rangle$ are the Pauli observables we estimated earlier

Computing the Energy $E(\psi)$

- Using the Pauli decomposition we have:

  $$E(\psi) = \langle\psi| \mathcal{H} |\psi\rangle = \sum c_{i_1,\cdots,i_n} \langle\psi| P_1^{i_1} \otimes \cdots \otimes P_n^{i_n} |\psi\rangle$$

  where $\langle\psi| P_1^{i_1} \otimes \cdots \otimes P_n^{i_n} |\psi\rangle$ are the Pauli observables we estimated earlier

- The accuracy of the energy estimate depends on the accuracy of individual terms, and the number of terms in the sum

- We will use this value as the "cost" of the state $|\psi\rangle$

Computing the Energy $E(\psi)$

- Using the Pauli decomposition we have:

  $E(\psi) = \langle\psi|\,\mathcal{H}\,|\psi\rangle = \sum c_{i_1,\cdots,i_n}\langle\psi|\,P_1^{i_1}\otimes\cdots\otimes P_n^{i_n}\,|\psi\rangle$

  where $\langle\psi|\,P_1^{i_1}\otimes\cdots\otimes P_n^{i_n}\,|\psi\rangle$ are the Pauli observables we estimated earlier

- The accuracy of the energy estimate depends on the accuracy of individual terms, and the number of terms in the sum
- We will use this value as the "cost" of the state $|\psi\rangle$
- For the earlier example: $H = \frac{1}{\sqrt{2}}(X+Z)$ we need to estimate two observables:

  $\langle\psi|\,X\,|\psi\rangle = O_1$; $\langle\psi|\,Z\,|\psi\rangle = O_2$

  Resulting to $E(\psi) = \frac{1}{\sqrt{2}}(O_1+O_2)$

Step 3: The Ansatz (Family of Quantum States)

Ansatz: The space we optimise

- To solve the Hamiltonian problem we need to find the quantum state that has **minimum energy** from *all* the states of the **Hilbert space**
- This is **infeasible**. Instead, we select a family of (parametrised) quantum states, and we hope that one **member of the family approx. well the ground state**.

Ansatz: The space we optimise

- To solve the Hamiltonian problem we need to find the quantum state that has **minimum energy** from *all* the states of the **Hilbert space**
- This is **infeasible**. Instead, we select a family of (parametrised) quantum states, and we hope that one **member of the family approx. well the ground state**.
- Can view the ansatz as a **family of parametrised quantum circuits**: $|\psi(\vec{\theta})\rangle = U(\vec{\theta})\,|0\rangle$
- The circuits $U(\vec{\theta})$ should be NISQ devices compatible (short depth, limited width)

Ansatz: The space we optimise

- To solve the Hamiltonian problem we need to find the quantum state that has **minimum energy** from *all* the states of the **Hilbert space**
- This is **infeasible**. Instead, we select a family of (parametrised) quantum states, and we hope that one **member of the family approx. well the ground state**.
- Can view the ansatz as a **family of parametrised quantum circuits**: $|\psi(\vec{\theta})\rangle = U(\vec{\theta})\,|0\rangle$
- The circuits $U(\vec{\theta})$ should be NISQ devices compatible (short depth, limited width)
- Two approaches: (i) Hardware Efficient, (ii) Problem Specific

(i) Hardware Efficient Ansatz

- Generate a family of states that spans evenly the Hilbert space
- Needs to be able to produced high entanglement
- Should choose circuits that are easy to implement with a given NISQ device
- Generic Ansatz that can be used for *any* Hamiltonian problem

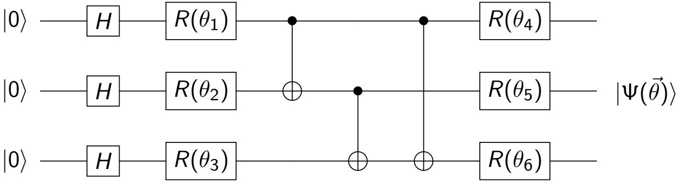
(i) Hardware Efficient Ansatz

- Generate a family of states that spans evenly the Hilbert space
- Needs to be able to produced high entanglement
- Should choose circuits that are easy to implement with a given NISQ device
- Generic Ansatz that can be used for *any* Hamiltonian problem
- **Typical Ansatz:**
  1. A number of layers repeating the same circuit with different parameters
  2. Single-qubit rotations parametrised by the rotation angle
  3. Entangling gates (non-commuting with the rotations)

     Hardware architecture determines entanglement topology (all-to-all Vs nearest-neighbour)

An Example: 3-qubit, 1-layer, all-to-all entanglement

$$\vec{\theta} := (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6)$$



(ii) Problem Specific Ansatz

- Generate a family of states that **uses the problem's Hamiltonian**
- Does not span evenly the Hilbert space – hopefully is **more dense around** the region we expect to have **the ground state**
- Not designed with the hardware in mind

(ii) Problem Specific Ansatz

- Generate a family of states that **uses the problem's Hamiltonian**
- Does not span evenly the Hilbert space – hopefully is **more dense around** the region we expect to have **the ground state**
- Not designed with the hardware in mind
- Is theoretically **more promising**, but in practice may lead to **more noisy results**
- Many families exist (e.g. unitary coupled cluster, adiabatic, etc).

(ii) Problem Specific Ansatz

- Generate a family of states that **uses the problem's Hamiltonian**
- Does not span evenly the Hilbert space – hopefully is **more dense around** the region we expect to have **the ground state**
- Not designed with the hardware in mind
- Is theoretically **more promising**, but in practice may lead to **more noisy results**
- Many families exist (e.g. unitary coupled cluster, adiabatic, etc).
- We give an important type suitable for optimisation problems:

  Quantum Approximate Optimisation Algorithm (QAOA)

- Let H_C be the **problem's Hamiltonian** of Ising type (only Pauli-Z, up to quadratic terms)

  Recall Max-Cut: $\mathcal{H}_C = \sum_{(i,j)\in E} Z_i \otimes Z_j$

- Let $H_B = \sum_i X_i$ be the **"mixer" Hamiltonian** (non-commuting with H_C)

Quantum Approximate Optimisation Algorithm

- Let H_C be the **problem's Hamiltonian** of Ising type (only Pauli-Z, up to quadratic terms)

  Recall Max-Cut: $\mathcal{H}_C = \sum_{(i,j)\in E} Z_i \otimes Z_j$

- Let $H_B = \sum_i X_i$ be the **"mixer" Hamiltonian** (non-commuting with H_C)

- A 1-layer QAOA ansatz is given by

  $e^{-i\beta H_B} e^{-i\gamma H_C} H^{\otimes n} |0\rangle^{\otimes n}$

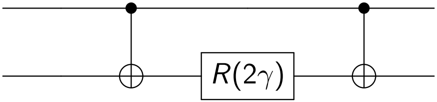
- A single layer has only **two parameters** (β, γ), irrespective of the number of qubits/width of computation

Quantum Approximate Optimisation Algorithm

- The mixer Hamiltonian leads to single-qubit rotations $e^{-i\beta X_i}$

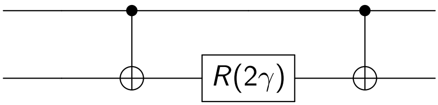
Quantum Approximate Optimisation Algorithm

- The mixer Hamiltonian leads to single-qubit rotations $e^{-i\beta X_i}$
- The problem Hamiltonian has terms $Z_i \otimes Z_j$ leading to unitaries $e^{-i\gamma Z_i Z_j}$:



Quantum Approximate Optimisation Algorithm

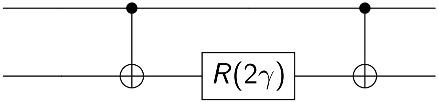
- The mixer Hamiltonian leads to single-qubit rotations $e^{-i\beta X_i}$
- The problem Hamiltonian has terms $Z_i \otimes Z_j$ leading to unitaries $e^{-i\gamma Z_i Z_j}$:



- Depending on interaction terms of H_C entangling gates can act on distant qubits
- Problem has to be Ising (i.e. only Pauli Z, and at most quadratic terms). Other cases can exist but ansatz becomes much harder to implement.

Quantum Approximate Optimisation Algorithm

- The mixer Hamiltonian leads to single-qubit rotations $e^{-i\beta X_i}$
- The problem Hamiltonian has terms $Z_i \otimes Z_j$ leading to unitaries $e^{-i\gamma Z_i Z_j}$:



- Depending on interaction terms of H_C entangling gates can act on distant qubits
- Problem has to be Ising (i.e. only Pauli Z, and at most quadratic terms). Other cases can exist but ansatz becomes much harder to implement.
- **More layers repeat the above** with fresh parameters (β_2, γ_2)

Step 4: Classical Optimisation & VQA Summary

Moving Through Parameter Space

Given:

- Ansatz: set of q-states parametrised by classical parameters $\vec{\theta}$
- How to compute the "cost-function" $E(\vec{\theta}) = \langle\Psi(\vec{\theta})|\,\mathcal{H}\,|\Psi(\vec{\theta})\rangle$

Moving Through Parameter Space

Given:

- Ansatz: set of q-states parametrised by classical parameters $\vec{\theta}$
- How to compute the "cost-function" $E(\vec{\theta}) = \langle\Psi(\vec{\theta})|\,\mathcal{H}\,|\Psi(\vec{\theta})\rangle$

Task:

- **Find the minimum** (local or global).

  Can use any **classical optimisation** technique

Moving Through Parameter Space

Given:

- Ansatz: set of q-states parametrised by classical parameters $\vec{\theta}$
- How to compute the "cost-function" $E(\vec{\theta}) = \langle \Psi(\vec{\theta}) | \mathcal{H} | \Psi(\vec{\theta}) \rangle$

Task:

- **Find the minimum** (local or global).

  Can use any **classical optimisation** technique

Examples of Techniques Used:

- Gradient Descent
- Monte Carlo - based
- Nelder-Mead method
- COBYLA (Constr. Optimiz. by Linear Approximation)
- Any deterministic or stochastic global optimisation method

Gradient Descent

- Trick to compute the gradient (generalises).
  Parametrised gates: $U_P = e^{-i\theta P}$ with P a Pauli (recall $P^2 = \mathrm{I}$)

$$U_P(\theta) = I\cos(\theta) - iP\sin(\theta) \; ; \; \frac{\partial}{\partial\theta} U_P(\theta) = -iPe^{-i\theta P}$$

leading to

$$\boxed{\frac{\partial}{\partial\theta} E(\theta) = E(\theta + \frac{\pi}{4}) - E(\theta - \frac{\pi}{4})}$$

Note: This difference is NOT infinitesimal!

Gradient Descent

- Trick to compute the gradient (generalises). Parametrised gates: $U_P = e^{-i\theta P}$ with P a Pauli (recall $P^2 = \mathbb{I}$)

$$U_P(\theta) = I\cos(\theta) - iP\sin(\theta)\ ;\ \frac{\partial}{\partial\theta}U_P(\theta) = -iPe^{-i\theta P}$$

leading to

$$\boxed{\frac{\partial}{\partial\theta}E(\theta) = E(\theta + \frac{\pi}{4}) - E(\theta - \frac{\pi}{4})}$$

Note: This difference is NOT infinitesimal!

- Move towards negative direction of the gradient: $\vec{\theta}_{i+1} = \vec{\theta}_i - \gamma\nabla E(\vec{\theta}_i)\ ;\ \nabla E(\vec{\theta}_i) = \left(\partial_{\theta^1}E(\vec{\theta}), \partial_{\theta^2}E(\vec{\theta}), \partial_{\theta^3}E(\vec{\theta})\right)$
- Can find **local minima** (not global)

Monte Carlo - Based

- Initial Guess $\vec{\theta}_0$ and compute $E(\vec{\theta}_0)$
- Generate a new guess $\vec{\theta'}$ (using any method, e.g. by varying a single parameter) and compute $E(\vec{\theta'})$

Monte Carlo - Based

- Initial Guess $\vec{\theta}_0$ and compute $E(\vec{\theta}_0)$
- Generate a new guess $\vec{\theta'}$ (using any method, e.g. by varying a single parameter) and compute $E(\vec{\theta'})$
- If, $E(\vec{\theta'}) \leq E(\vec{\theta}_0)$, **keep guess** and continue with $\vec{\theta}_1 := \vec{\theta'}$

Monte Carlo - Based

- Initial Guess $\vec{\theta}_0$ and compute $E(\vec{\theta}_0)$
- Generate a new guess $\vec{\theta'}$ (using any method, e.g. by varying a single parameter) and compute $E(\vec{\theta'})$
- If, $E(\vec{\theta'}) \leq E(\vec{\theta}_0)$, **keep guess** and continue with $\vec{\theta}_1 := \vec{\theta'}$
- Else, with probability $p_r = 1 - e^{-\beta(E(\vec{\theta'}) - E(\vec{\theta}_0))}$, **reject guess** (make a fresh guess starting again from $\vec{\theta}_0$)
- With the remaining probability $p_a = e^{-\beta(E(\vec{\theta'}) - E(\vec{\theta}_0))}$, **keep guess** and continue with $\vec{\theta}_1 := \vec{\theta'}$

Monte Carlo - Based

- Initial Guess $\vec{\theta}_0$ and compute $E(\vec{\theta}_0)$
- Generate a new guess $\vec{\theta'}$ (using any method, e.g. by varying a single parameter) and compute $E(\vec{\theta'})$
- If, $E(\vec{\theta'}) \leq E(\vec{\theta}_0)$, **keep guess** and continue with $\vec{\theta}_1 := \vec{\theta'}$
- Else, with probability $p_r = 1 - e^{-\beta(E(\vec{\theta'}) - E(\vec{\theta}_0))}$, **reject guess** (make a fresh guess starting again from $\vec{\theta}_0$)
- With the remaining probability $p_a = e^{-\beta(E(\vec{\theta'}) - E(\vec{\theta}_0))}$, **keep guess** and continue with $\vec{\theta}_1 := \vec{\theta'}$

Observations

- May keep moving with greater new Energy (esc local min)
- Keeping probability reduces with Energy difference
- β is "inverse temperature". Can increase value with iteration steps $\beta(i)$ (so it stabilises in time)

Limitations

- **Resources:**

- Repetitions per round (accuracy desired)

- Optimisation rounds (classical optimisation problem)

Limitations

- **Resources:**
  - Repetitions per round (accuracy desired)
  - Optimisation rounds (classical optimisation problem)
- **Effect of noise:**
  - Greater repetitions to achieve desired accuracy
  - Potentially systematic errors in estimating cost

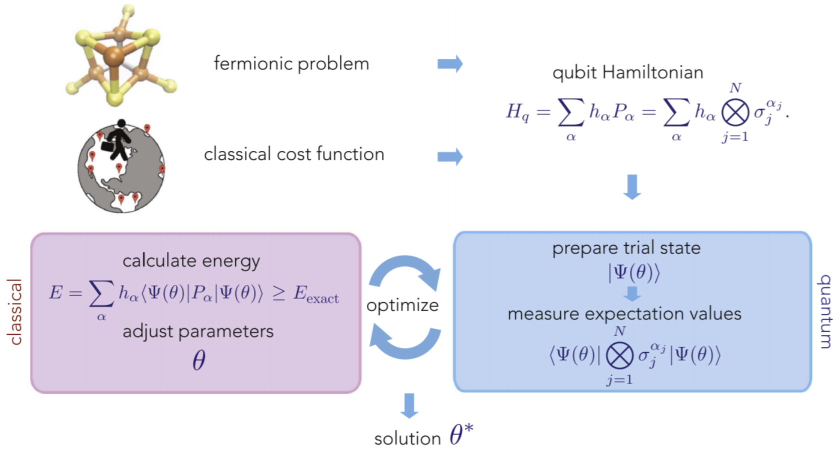
Limitations

- **Resources:**
  - Repetitions per round (accuracy desired)
  - Optimisation rounds (classical optimisation problem)
- **Effect of noise:**
  - Greater repetitions to achieve desired accuracy
  - Potentially systematic errors in estimating cost
- **Possible Failures:**
  - Converge to local minimum
  - Fail to conv. (Flat Landscape a.e.) "Barren Plateau"

Limitations

- **Resources:**
  - Repetitions per round (accuracy desired)
  - Optimisation rounds (classical optimisation problem)
- **Effect of noise:**
  - Greater repetitions to achieve desired accuracy
  - Potentially systematic errors in estimating cost
- **Possible Failures:**
  - Converge to local minimum
  - Fail to conv. (Flat Landscape a.e.) "Barren Plateau"
- **Overall limitations (heuristics):**
  - Not exact complexity
  - No guarantee for obtaining solution

VQA Pictorially



Taken from: Nikolaj Moll et al 2018 Quantum Sci. Technol. **3** 030503

VQA: Summary

- Hybrid Quantum-Classical Algorithms
- Quantum Part needs small coherence time, works without QECC and can be optimised for given hardware:

  Best candidate for quantum speed-up at NISQ devices

VQA: Summary

- Hybrid Quantum-Classical Algorithms
- Quantum Part needs small coherence time, works without QECC and can be optimised for given hardware:

  Best candidate for quantum speed-up at NISQ devices

1. Map problem to ground state energy of some Hamiltonian $\mathcal{H}$ (classical)

VQA: Summary

- Hybrid Quantum-Classical Algorithms

- Quantum Part needs small coherence time, works without QECC and can be optimised for given hardware:

  Best candidate for quantum speed-up at NISQ devices

1. Map problem to ground state energy of some Hamiltonian $\mathcal{H}$ (classical)

2. Can generate (efficiently) a family of states $|\Psi(\vec{\theta})\rangle$ (quantum)

3. Can compute expectation value $E_{\vec{\theta}} = \langle\Psi(\vec{\theta})|\,\mathcal{H}\,|\Psi(\vec{\theta})\rangle$ with local Pauli measurement for any guess $\vec{\theta}$ (quantum)

VQA: Summary

- Hybrid Quantum-Classical Algorithms
- Quantum Part needs small coherence time, works without QECC and can be optimised for given hardware:

  Best candidate for quantum speed-up at NISQ devices

1. Map problem to ground state energy of some Hamiltonian $\mathcal{H}$ (classical)
2. Can generate (efficiently) a family of states $|\Psi(\vec{\theta})\rangle$ (quantum)
3. Can compute expectation value $E_{\vec{\theta}} = \langle\Psi(\vec{\theta})|\,\mathcal{H}\,|\Psi(\vec{\theta})\rangle$ with local Pauli measurement for any guess $\vec{\theta}$ (quantum)
4. Comparing $E(\vec{\theta})$ of existing points, evaluate a new guess $\vec{\theta}'$ using standard techniques.
   Feedback to step 2 to evaluate $E(\vec{\theta}')$ (classical)

Variational Quantum Algorithms Reviews

1. *Variational quantum algorithms*, Cerezo, Marco, et al. Nature Reviews Physics (2021): 1-20.
2. *Noisy intermediate-scale quantum (NISQ) algorithms*, Bharti, Kishor, et al. arXiv preprint arXiv:2101.08448 (2021).
3. *Quantum optimization using variational algorithms on near-term quantum devices*, Moll, Nikolaj, et al. Quantum Science and Technology 3.3 (2018): 030503.