Introduction to Quantum Computing
Lecture 15: Variational Quantum Algorithms I

Petros Wallden
School of Informatics, University of Edinburgh
14th November 2023
Noisy Intermediate Scale Quantum Devices and Near-Term Quantum Algorithms

Variational Quantum Algorithms: What & How (4 steps)

Step 1: Hamiltonian Problem with an Example (Max-Cut)
Noisy Intermediate Scale Quantum Devices and

Near-Term Quantum Algorithms
Noisy Intermediate-Scale Quantum (NISQ) Devices

- **Qubit Number**

  Number of qubits a processor have (width of computation)
Noisy Intermediate-Scale Quantum (NISQ) Devices

- **Qubit Number**
  Number of qubits a processor have (width of computation)

- **Gate Fidelity**
  Quality of quantum gates
Noisy Intermediate-Scale Quantum (NISQ) Devices

- **Qubit Number**
  Number of qubits a processor have (width of computation)

- **Gate Fidelity**
  Quality of quantum gates

- **Coherence Time**
  Time that quantum information can be stored
NISQ Devices: Limitations

- Quantum Error Correction **not possible** (too few qubits)

Fault Tolerant Quantum Computation not in “Near-Term”
NISQ Devices: Limitations

- Quantum Error Correction **not possible** (too few qubits)
  
  Fault Tolerant Quantum Computation not in “Near-Term”

- Quality of output deteriorates with the size (width), number of gates applied, depth of computation (time taken)
NISQ Devices: Limitations

- Quantum Error Correction **not possible** (too few qubits)
- Fault Tolerant Quantum Computation not in "Near-Term"
- Quality of output deteriorates with the size (width), number of gates applied, depth of computation (time taken)
- Architecture **topology** is important
  - Nearest-neighbour interaction leads to more physical gates to implement a computation than all-to-all connectivity
NISQ Devices: Limitations

- Quantum Error Correction **not possible** (too few qubits)

  Fault Tolerant Quantum Computation not in “Near-Term”

- Quality of output deteriorates with the size (width), number of gates applied, depth of computation (time taken)

  **Architecture topology** is important

  Nearest-neighbour interaction leads to more physical gates to implement a computation than all-to-all connectivity

  Beyond a point **output is random**

  Even before that point output offers **no longer an advantage** to classical methods
NISQ Devices: Limitations

- Quantum Error Correction **not possible** (too few qubits)
- Fault Tolerant Quantum Computation not in “Near-Term”
- Quality of output deteriorates with the size (width), number of gates applied, depth of computation (time taken)
- **Architecture topology** is important

  Nearest-neighbour interaction leads to more physical gates to implement a computation than all-to-all connectivity

  Beyond a point **output is random**

  Even before that point output offers **no longer an advantage** to classical methods

**Main Question:**

Can NISQ devices offer **computational advantage** and how?
NISQ Devices: Where we are

Superconducting hardware

- Number of Qubits: $\approx 100$ (IBM’s “Osprey” has 433 and plans to announce by the end of the year “Condor” with 1121 qubits)

- Circuit depth: $\approx 100 : 20$ cycles of 5 gates

- Quality of gates (a bit outdated):
  - 1-qubit gate error: $1.6 \times 10^{-3}$
  - 2-qubit gate error: $6.2 \times 10^{-3}$
  - Measurement error: $3.2 \times 10^{-2}$

From “Quantum supremacy using a programmable superconducting processor”, Frank Arute, Kunal Arya, · · ·, John M. Martinis, Nature volume 574, 505 (2019)
Use of Hybrid Quantum - Classical Algorithms

- Move big part of the computation to the classical devices
- Use of QC for specific subroutine that is (classically) computationally expensive
- Quantum part can be completed with NISQ devices
  - Possibly using multiple repetitions, each requiring small coherence time
- Can find a “quantum” solution to any problem:
  - Take a classical algorithm for the problem and replace expensive subroutines with quantum ones
  - Heuristics with potential speed-ups (to be examined case-by-case)
NISQ Devices: An approach towards quantum advantage

- Use of Hybrid Quantum - Classical Algorithms
- Move big part of the computation to the classical devices
NISQ Devices: An approach towards quantum advantage

- Use of Hybrid Quantum - Classical Algorithms
- Move big part of the computation to the classical devices
- Use of QC for specific subroutine that is (classically) computationally expensive
NISQ Devices: An approach towards quantum advantage

- Use of Hybrid **Quantum** - **Classical** Algorithms
- Move **big part of the computation** to the classical devices
- Use of QC for **specific subroutine** that is (classically) computationally expensive
- **Quantum part can be completed with NISQ devices**

Possibly using **multiple repetitions**, each requiring **small coherence time**
NISQ Devices: An approach towards quantum advantage

- Use of Hybrid Quantum - Classical Algorithms
- Move big part of the computation to the classical devices
- Use of QC for specific subroutine that is (classically) computationally expensive
- Quantum part can be completed with NISQ devices

Possibly using multiple repetitions, each requiring small coherence time

- Can find a “quantum” solution to any problem:

  Take a classical algorithm for the problem and replace expensive subroutines with quantum ones
NISQ Devices: An approach towards quantum advantage

- Use of Hybrid Quantum - Classical Algorithms
- Move big part of the computation to the classical devices
- Use of QC for specific subroutine that is (classically) computationally expensive
- Quantum part can be completed with NISQ devices
  Possibly using multiple repetitions, each requiring small coherence time
- Can find a “quantum” solution to any problem:
  Take a classical algorithm for the problem and replace expensive subroutines with quantum ones
- Heuristics with potential speed-ups
  (to be examined case-by-case)
Variational Quantum Algorithms: What & How (4 steps)
Given a Hermitian matrix $\mathcal{H}$ (typically called Hamiltonian), compute its smallest eigenvalue (called “ground state energy”).
Given a Hermitian matrix $\mathcal{H}$ (typically called Hamiltonian), compute its smallest eigenvalue (called “ground state energy”)

There exist variations:
- Find the minimum eigenvector (called “ground state”)
- Find other eigenvalues or eigenvectors
- Find the expectation value (“energy”) of a quantum state $|\psi\rangle$

$$\langle\psi| \mathcal{H} |\psi\rangle$$
Given a \textbf{Hermitian matrix} $\mathcal{H}$ (typically called Hamiltonian), compute its \textbf{smallest eigenvalue} (called “ground state energy”)

There exist variations:

- Find the \textbf{minimum eigenvector} (called “ground state”)

- Find other \textbf{eigenvalues} or \textbf{eigenvectors}

- Find the \textbf{expectation value} ("energy") of a quantum state $|\psi\rangle$

$$\langle \psi | \mathcal{H} | \psi \rangle$$

\textbf{How to use this to solve everyday problems?}
The $k$-local Hamiltonian problem is QMA-complete.

- **QMA**: class of problems that they can be verified in poly-time by a quantum computer

  QMA is to BQP, what NP is to P

- **QMA** contains both BQP and NP

  We can use VQA to solve all problems in NP and BQP! But is it really practical? (not always: time, prob of success)
$k$-local Hamiltonian problem is QMA-complete

- **QMA**: class of problems that they can be verified in poly-time by a quantum computer

  QMA is to BQP, what NP is to P

- QMA contains both BQP and NP

- The $k$-local Hamiltonian problem is:

  Find the **ground state energy** of a Hamiltonian $\mathcal{H} = \sum_i \mathcal{H}_i$ where each $\mathcal{H}_i$ acts on at most $k$-qubits.

  This is QMA-complete! (similar to $k -$ SAT)
The $k$-local Hamiltonian problem is QMA-complete

- **QMA**: class of problems that they can be verified in poly-time by a quantum computer

  QMA is to BQP, what NP is to P

- **QMA** contains both BQP and NP

- The $k$-local Hamiltonian problem is:

  Find the **ground state energy** of a Hamiltonian $\mathcal{H} = \sum_i \mathcal{H}_i$ where each $\mathcal{H}_i$ acts on at most $k$-qubits.

  This is **QMA-complete**! (similar to $k-SAT$)

- We can use VQA to solve all problems in **NP** and **BQP**!

- But is it really practical? (not always: time, prob of success)
Applications: Why is this task useful

- Optimisation
- Quantum Chemistry
- Quantum Simulation
- Many-body Physics
- Quantum Machine Learning
VQA: four steps

Step 1 Hamiltonian Encoding

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 $\langle \psi | \mathcal{H} | \psi \rangle$
VQA: four steps

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 $C(\vec{\theta}) := \langle \psi(\vec{\theta}) | H | \psi(\vec{\theta}) \rangle$, ie $\vec{\theta}^* := \arg \min_{\vec{\theta}} C(\vec{\theta})$. 

Petros Wallden
Lecture 15: Variational Quantum Algorithms I
Step 1: Hamiltonian Problem with an Example (Max-Cut)
The Max-Cut Problem

- Given Graph \( G = (V, E) \)
  - with vertices \( v \in V \) and edges \( e = (v_1, v_2) \in E \)
- Partition vertices to two sets \( S, T \)
  - where \( S \cup T = V \) and \( S \cap T = \emptyset \)
- **Cut** is the number of edges between the two sets \( S, T \)
  - (\# of red edges)
The Max-Cut Problem

**Task:** Select $S, T$ such that the Cut is maximised

$$\max_{(S,T)} \#(s,t) \in E \mid s \in S \land t \in T$$

- Decision version of Max-Cut is **NP-complete**
- Max(Min)-Cut has applications in Flow Networks including circuit optimisation (VLSI design), computer vision and others
- Version that edges have a weight $w_e$ and one maximises the total weight of the cut edges exists (similar analysis):

$$\max_{(S,T)} \sum_{(s,t)} w(s,t) \mid (s,t) \in E \land s \in S \land t \in T$$
Need to use our tool (ground state energy of a Hamiltonian)

In general one can take any classical algorithm that solves Max-Cut and replace an expensive sub-routine with a Hamiltonian problem

Natural map of this problem to a (simple) Hamiltonian
Towards a Quantum Solution for Max-Cut

- Need to use our tool (ground state energy of a Hamiltonian)

- In general one can take any classical algorithm that solves Max-Cut and replace an expensive sub-routine with a Hamiltonian problem

- Natural map of this problem to a (simple) Hamiltonian

- Assign to each vertex $v$ a spin $s_v \in \{+1, -1\}$

- Those with $s_i = +1$ define the one set (say $S$) those with $s_i = -1$ define the other set (say $T$)
Consider the cost $\mathcal{H}(\vec{s})$ (energy) of a configuration $\vec{s} := (s_1, \cdots, s_n)$

Split the edges to three sets:

- $E^{+1}$ edges between vertices that both have $s = +1$
- $E^{-1}$ edges between vertices that both have $s = -1$
- $E^C$ edges between vertices with different spins (the “cut”)
Consider the cost $\mathcal{H}(\vec{s})$ (energy) of a configuration $\vec{s} := (s_1, \cdots, s_n)$

Split the edges to three sets:

$E^{+1}$ edges between vertices that both have $s = +1$

$E^{-1}$ edges between vertices that both have $s = -1$

$E^C$ edges between vertices with different spins (the “cut”)

$$\mathcal{H}(\vec{s}) = \sum_{(i, j) \in E(G)} s_i s_j$$

$$= \sum_{(i, j) \in E^{+1}(G)} s_i s_j + \sum_{(i, j) \in E^{-1}(G)} s_i s_j + \sum_{(i, j) \in E^C(G)} s_i s_j$$
Towards a Quantum Solution for Max-Cut

Note that $s_is_j = 1$ for $E^{+1}, E^{-1}$ while $s_is_j = -1$ for $E^C$:

$$\mathcal{H}(\vec{s}) = \sum_{(i,j) \in E^{+1}(G)} 1 \quad \sum_{(i,j) \in E^{-1}(G)} 1 \quad \sum_{(i,j) \in E^C(G)} 1$$

$$= \sum_{(i,j) \in E^{+1}(G)} + \sum_{(i,j) \in E^{-1}(G)} + \sum_{(i,j) \in E^C(G)} 1 - 2 \sum_{(i,j) \in E^C(G)} 1$$

$$= \sum_{(i,j) \in E(G)} -2 \sum_{(i,j) \in E^C(G)}$$

$$= |E| - 2\text{Cut}(G) \quad (2)$$
Towards a Quantum Solution for Max-Cut

- Note that \( s_i s_j = 1 \) for \( E^+ \), \( E^- \) while \( s_i s_j = -1 \) for \( E^C \):

\[
\mathcal{H}(\vec{s}) = \sum_{(i,j) \in E^+} 1 + \sum_{(i,j) \in E^-} 1 - \sum_{(i,j) \in E^C} 1
\]

\[
= \sum_{(i,j) \in E^+} + \sum_{(i,j) \in E^-} + \sum_{(i,j) \in E^C} 1 - 2 \sum_{(i,j) \in E^C} 1
\]

\[
= \sum_{(i,j) \in E} -2 \sum_{(i,j) \in E^C} 1
\]

\[
= |E| - 2\text{Cut}(G)
\]

(2)

- The greater the \( \text{Cut}(G) \) the smaller the energy \( \mathcal{H}(\vec{s}) \)

- Minimising Energy = Solving Max-Cut!
Towards a Quantum Solution for Max-Cut

- Map each spin $s_i$ to a qubit $|x_i\rangle$, where $+1 \to |0\rangle$; $-1 \to |1\rangle$

- The cost function (Hamiltonian) changes
  \[ H(\vec{s}) = \sum_{(i,j) \in E} s_is_j \to \hat{H}(\vec{x}) := \sum_{(i,j) \in E} (-1)^{x_i + x_j} \]
  \[ \to \hat{H}(\vec{x}) := \sum_{(i,j) \in E} Z_i \otimes Z_j \]
Towards a Quantum Solution for Max-Cut

- Map each spin $s_i$ to a qubit $|x_i\rangle$, where $+1 \rightarrow |0\rangle$; $-1 \rightarrow |1\rangle$

- The cost function (Hamiltonian) changes
  \[ H(\vec{s}) = \sum_{(i,j) \in E} s_i s_j \rightarrow H(\vec{x}) := \sum_{(i,j) \in E} (-1)^{x_i + x_j} \]

  \[ \rightarrow \hat{H}(\vec{x}) := \sum_{(i,j) \in E} Z_i \otimes Z_j \]

  **Check:** For each edge $(i, j) \in E$ we have
  \[ Z_i \otimes Z_j |x_i\rangle \otimes |x_j\rangle = (-1)^{x_i + x_j} |x_i\rangle \otimes |x_j\rangle \]

  As earlier, if edge of same type $\rightarrow$ even parity there a $+1$ contribution (comp states remain invariant)

  If edge of different type (i.e. counts in “cut”) $\rightarrow$ odd parity and contributes as $-1$ (comp states remain invariant)
Towards a Quantum Solution for Max-Cut

- Map each spin $s_i$ to a qubit $|x_i\rangle$, where $+1 \rightarrow |0\rangle$; $-1 \rightarrow |1\rangle$

- The cost function (Hamiltonian) changes

$$\mathcal{H}(\vec{s}) = \sum_{(i,j) \in E} s_i s_j \rightarrow \mathcal{H}(\vec{x}) := \sum_{(i,j) \in E} (-1)^{x_i + x_j}$$

$$\rightarrow \hat{\mathcal{H}}(\vec{x}) := \sum_{(i,j) \in E} Z_i \otimes Z_j$$

**Check:** For each edge $(i,j) \in E$ we have

$$Z_i \otimes Z_j |x_i\rangle \otimes |x_j\rangle = (-1)^{x_i + x_j} |x_i\rangle \otimes |x_j\rangle$$

As earlier, if edge of same type $\rightarrow$ even parity there a $+1$ contribution (comp states remain invariant)

If edge of different type (i.e. counts in “cut”) $\rightarrow$ odd parity and contributes as $-1$ (comp states remain invariant)

- Taking all terms together:

$$\sum_{(i,j) \in E} Z_i \otimes Z_j |x_1 \cdots x_n\rangle = (|E| - 2\text{Cut}(G)) |x_1 \cdots x_n\rangle$$
The smallest eigenvalue of $\hat{H}(\vec{x})$ gives the maximum $\text{Cut}(G)$. 

The smallest eigenvalue of $\hat{H}(\vec{x})$ gives the maximum $\text{Cut}(G)$

Special case of an Ising Hamiltonian (important class)

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

The smallest eigenvalue of $\hat{H}(\vec{x})$ gives the maximum $\text{Cut}(G)$.

Special case of an Ising Hamiltonian (important class)

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


We want to find $|\vec{x}\rangle$ that minimises this Hamiltonian.
The smallest eigenvalue of $\hat{\mathcal{H}}(\vec{x})$ gives the maximum $\text{Cut}(G)$

Special case of an Ising Hamiltonian (important class)

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


We want to find $|\vec{x}\rangle$ that minimises this Hamiltonian

Next Lecture:

- How to compute the cost/energy of a quantum state
  $$C(\psi) := \langle \psi | \mathcal{H} | \psi \rangle$$

- How to approximate the minimum without brute-forcing the full Hilbert space
Variational Quantum Algorithms Reviews

