Solving systems of equations through quantum computing

Sara Galatro

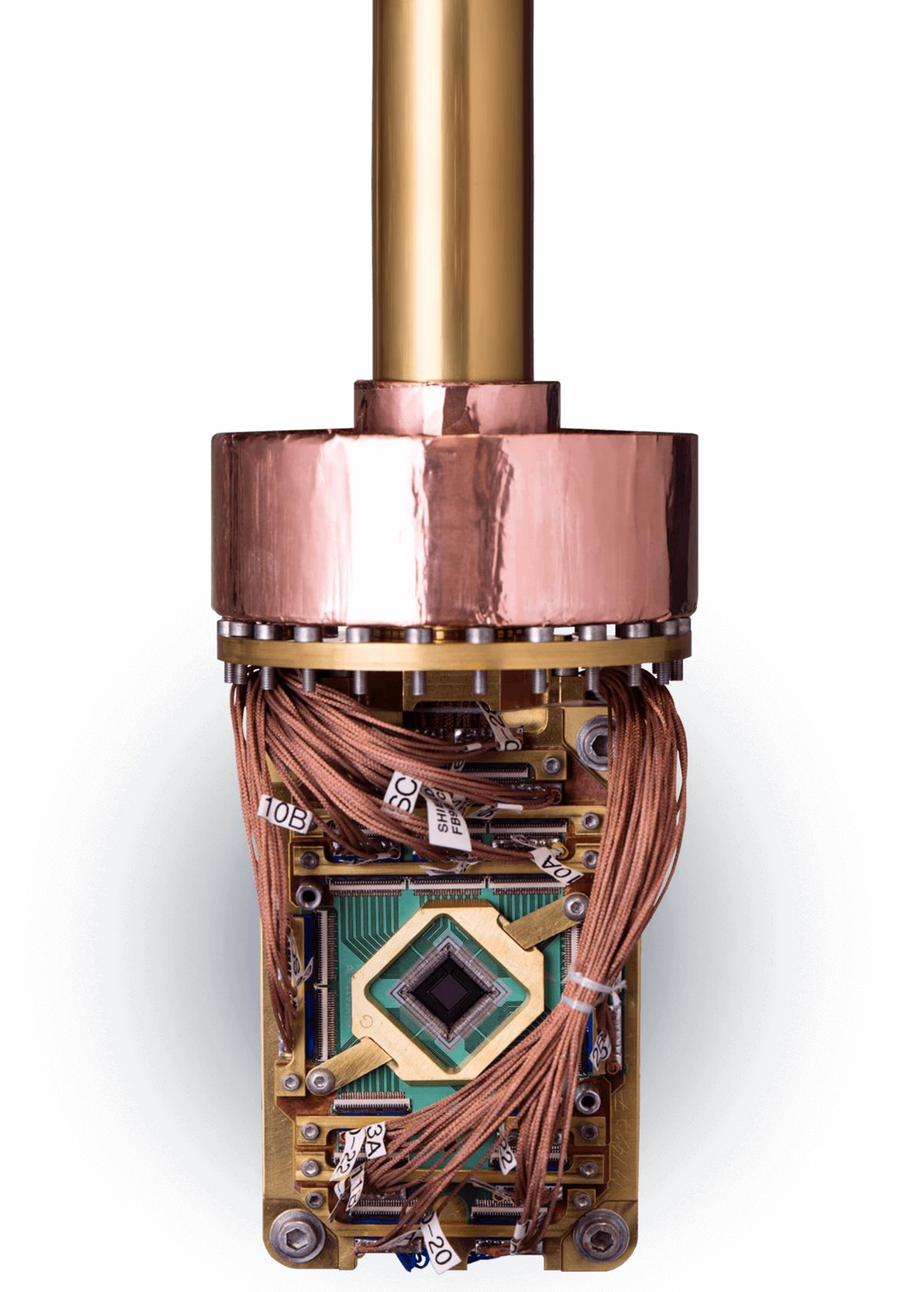
Master 's Thesis in Computational Sciences

Department of Mathematics and Physics

ROMA TRE
UNIVERSITÀ DEGLI STUDI

Introduction

- Quantum computing promises to become a powerful tool for solving classically expensive problems, but the current devices have limited capacity and service due to **noise** and **decoherence**.
- Two approaches, however, seem to help bridge this computational gap - **Variational Quantum Algorithms** and **Quantum Annealing**.
- We test both these methods to solve **systems of multivariate equations**.
- Results show that a **synergy** between quantum and classical algorithms is required to achieve interesting results.



Contents

- Quantum mechanics
- Gate-Model Computing
- Quantum Annealing
- Experiments:
  - Variational Quantum Linear Solver (VQLS)
  - Quantum Approximate Optimization Algorithm (QAOA)
  - MQ with Quantum Annealing
- Conclusions and future work

Quantum Mechanics: Dirac Notation

- We assume to be working in a **complex vector** space $\mathbb{C}^N$.
- A **ket** $|\psi\rangle$ is a column vector and a **bra** $\langle\psi|$ is the associated row vector:

$$|\psi\rangle^\dagger = \begin{bmatrix} \psi_1 \\ \vdots \\ \psi_N \end{bmatrix}^\dagger \equiv [\psi_1^*, \dots, \psi_N^*] = \langle\psi|$$

- Suppose we have two complex vector spaces, V and W. The **tensor product** of two kets from these spaces, say $|\psi\rangle \in V$ and $|\varphi\rangle \in W$, is a ket in $V \otimes W$ and it can be written using various notations:

$$|\psi\rangle \otimes |\varphi\rangle \qquad |\psi\rangle|\varphi\rangle \qquad |\psi\phi\rangle$$

Quantum Mechanics

First Postulate

Associated to any isolated quantum system is a complex Hilbert space known as the **state space** of the system. The system is completely described by its **state vector**, a *unit* vector in its state space.

- The **qubit** is the information unit of quantum information and quantum computing, and it is associated with a two-dimensional Hilbert space.
- The most used base to describe a qubit is the **computational basis**, defined as:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

- An arbitrary state $|\psi\rangle$ can be written as a **superposition** of basis states:

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$$

where $\alpha_0, \alpha_1 \in \mathbb{C}$ are such that

$$|\alpha_0|^2 + |\alpha_1|^2 = 1$$

Quantum Mechanics

First Postulate

Associated to any isolated quantum system is a complex Hilbert space known as the **state space** of the system. The system is completely described by its **state vector**, a *unit* vector in its state space.

Second Postulate

The evolution of a closed quantum system is described by a **unitary operator** that only depends on the times t_1, t_2:

$$|\psi(t_2)\rangle = \mathbf{U}|\psi(t_1)\rangle$$

- Examples:

$$\sigma^x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad \sigma^y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$\sigma^z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad \mathbf{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

- An alternative statement can be given using continuous time and the **Schrödinger equation**:

$$i\hbar \frac{d|\psi\rangle}{dt} = \mathcal{H}|\psi\rangle$$

Quantum Mechanics

Third Postulate

Quantum **measurements** are described by a collection $\{\mathbf{M}_m\}$ of measurement operators, which act on the state space of the system being measured. After the measurement, the system **collapses** to the measured classical state.

- We can measure a state

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$$

with respect to the computational basis using the **projectors $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$**, obtaining the corresponding outputs with probability $|\alpha_0|^2$ and $|\alpha_1|^2$ respectively.

- We can measure with respect to a quantum observable $\mathbf{M}$ using its spectral decomposition

$$\mathbf{M} = \sum_{\lambda} \lambda \mathbf{P}_{\lambda}$$

Hence $\mathbb{E}[\mathbf{M}] \equiv \langle \mathbf{M} \rangle = \langle \psi|\mathbf{M}|\psi\rangle$, for any $|\psi\rangle$.

Quantum Mechanics

Third Postulate

Quantum **measurements** are described by a collection $\{\mathbf{M}_m\}$ of measurement operators, which act on the state space of the system being measured. After the measurement, the system **collapses** to the measured classical state.

Fourth Postulate

The state space of a **composite physical system** is the tensor product of the state spaces of the component physical systems. Moreover, if we have systems numbered 1 through n, and system number i is prepared in the state $|\psi\rangle$,then the joint state of the total system is

$$|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle$$

- If unitary operators act independently on different subsystem, we can write the overall operator as

$$\mathbf{U} = \mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_k$$

- It is useful to define **controlled operations**, a core term in quantum computation:

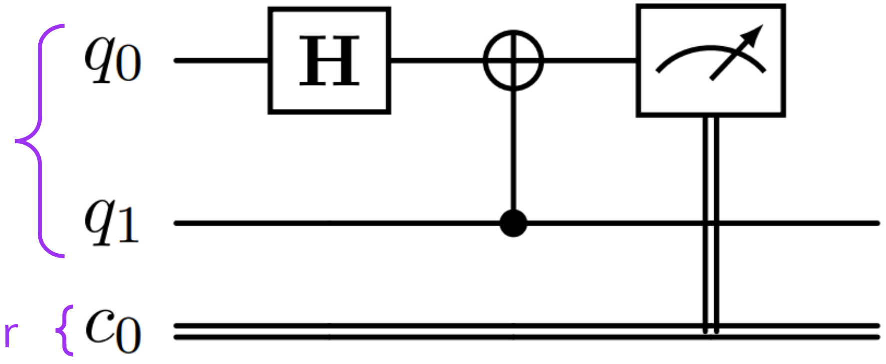
$$|0\rangle\langle 0| \otimes \mathbb{I}_T + |1\rangle\langle 1| \otimes \mathbf{U}_\mathrm{T} = \begin{bmatrix} \mathbb{I}_T & 0 \\ 0 & \mathbf{U}_\mathrm{T} \end{bmatrix}$$

Quantum Circuits

- Quantum circuits are a **universal quantum model** used to devise and analyze quantum algorithms.
- Quantum analogue of logical circuits, they are defined starting from **registers**, both quantum and classical.

Quantum registers

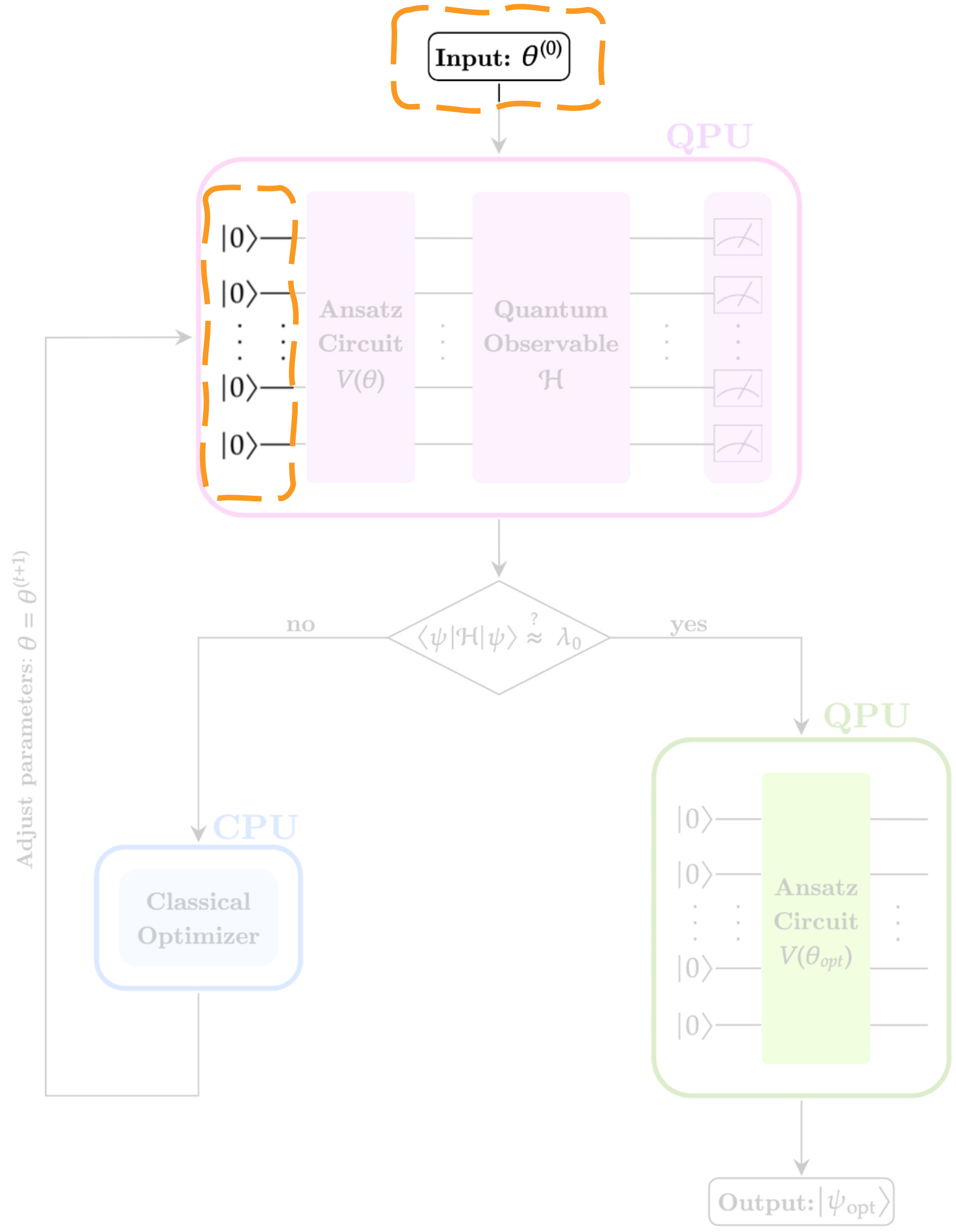
Classical register



$$\longrightarrow \quad \mathbf{CX}_{10} \cdot (\mathbf{H} \otimes \mathbb{I})|00\rangle$$

Transpiling Process

- Each Quantum Processor Unit (QPU) has a distinctive topology and native set of universal gates.
- If not native, every unitary operator must be **simulated**.
- Even if all gates are native to the QPU, a circuit must still be **embedded** in the QPU's layout.
- The combination of all these processes is called **Transpilation**.
- Transpiling a circuit may cause an increase in **size** and **depth**, making it **unfeasible**.



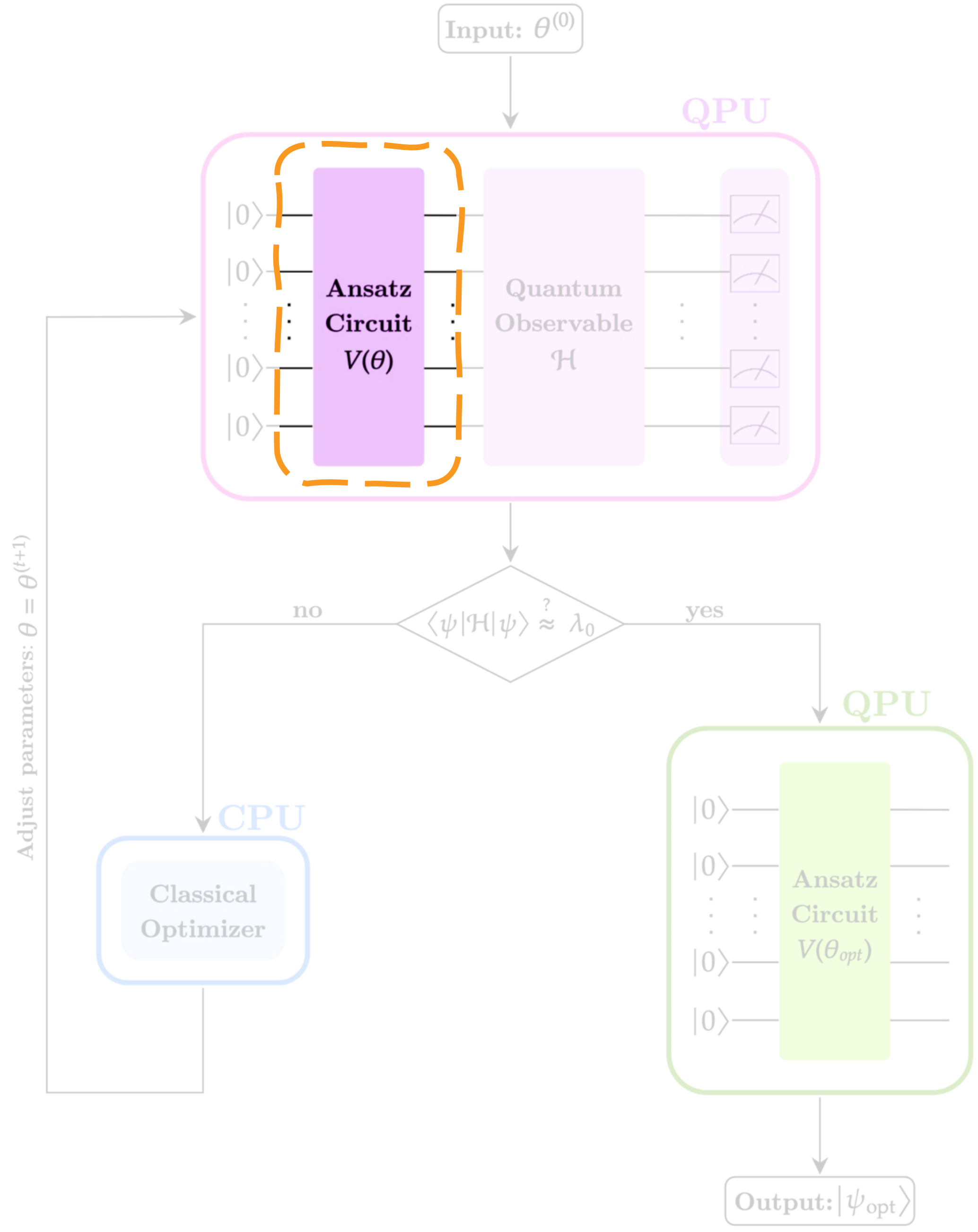
VQAs

- VQAs use a classical optimizer to minimize a cost function by training a **parameterized quantum circuit**.
- The cost function is defined as

$$\min_{\boldsymbol{\theta}} C(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \langle \psi(\boldsymbol{\theta})|\mathcal{H}|\psi(\boldsymbol{\theta})\rangle \geq \lambda_0$$

- Any VQA can be decomposed in **five submodules**:

1) Initialization
2) Parameterized circuit
3) Cost evaluation
4) Classical optimizer
5) Adjust ansatz parameters and re-run



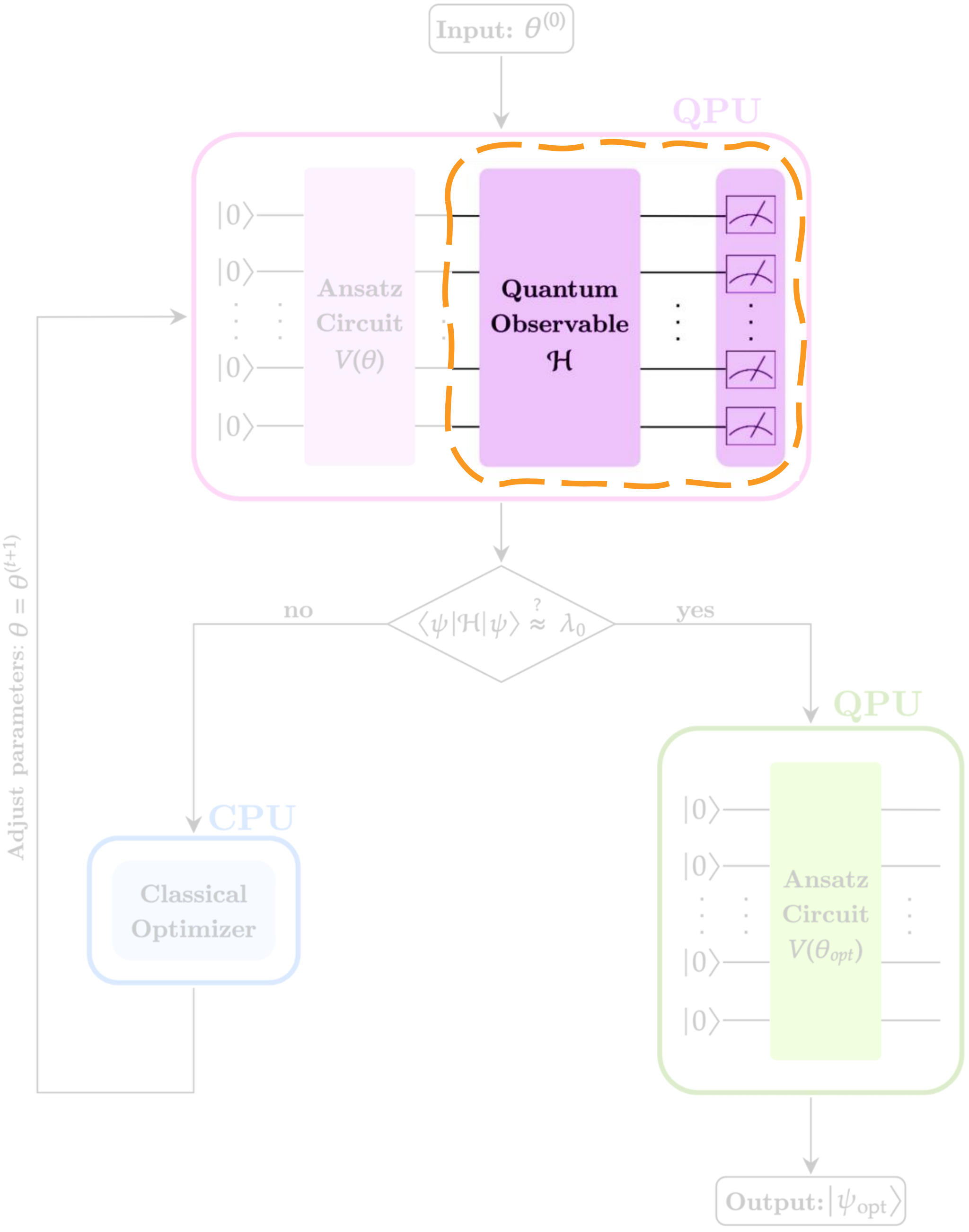
VQAs

- VQAs use a classical optimizer to minimize a cost function by training a **parameterized quantum circuit**.
- The cost function is defined as

$$\min_{\boldsymbol{\theta}} C(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \langle \psi(\boldsymbol{\theta}) | \mathcal{H} | \psi(\boldsymbol{\theta}) \rangle \geq \lambda_0$$

- Any VQA can be decomposed in **five submodules**:

1) Initialization
2) Parameterized circuit
3) Cost evaluation
4) Classical optimizer
5) Adjust ansatz parameters and re-run



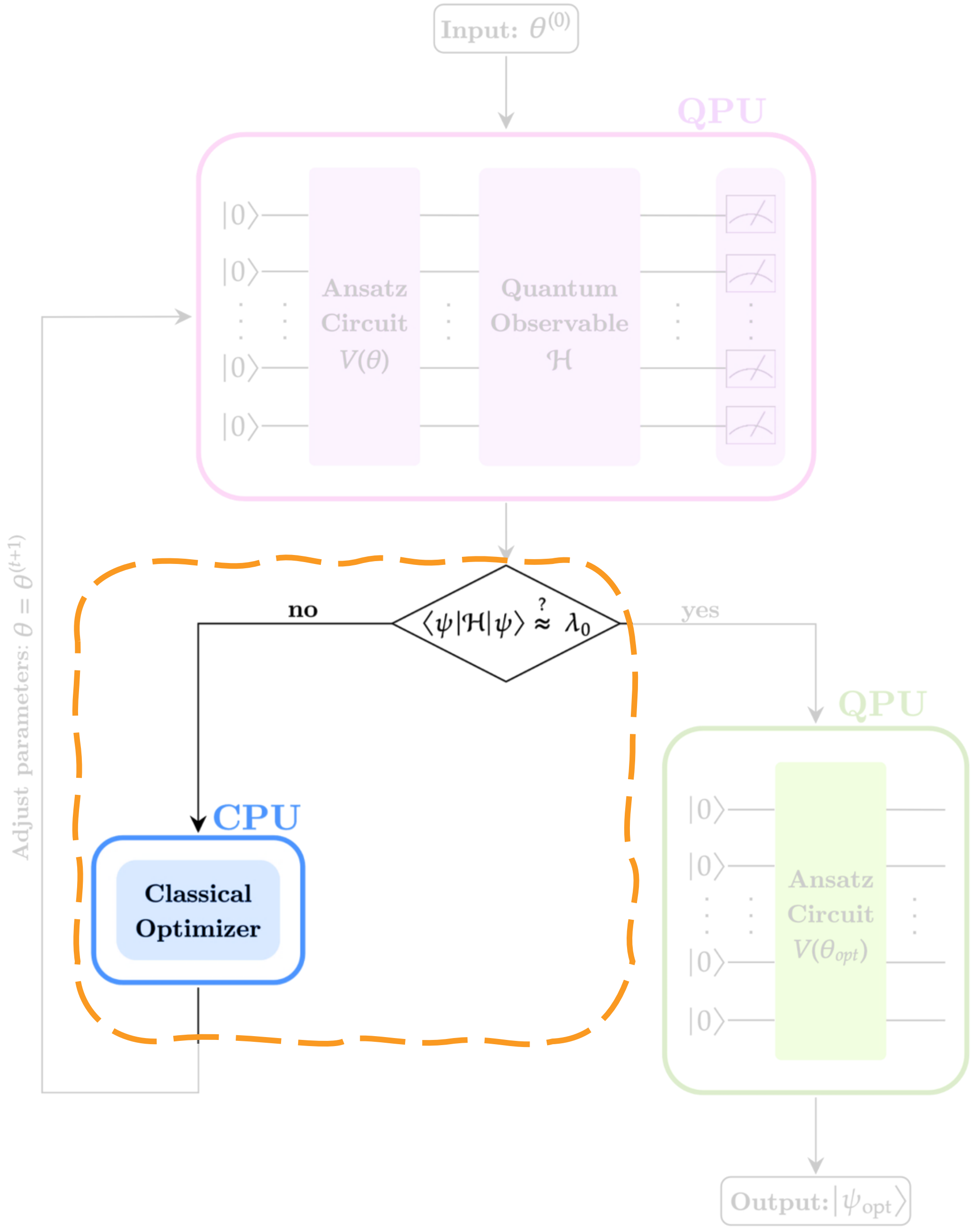
VQAs

- VQAs use a classical optimizer to minimize a cost function by training a **parameterized quantum circuit**.
- The cost function is defined as

$$\min_{\boldsymbol{\theta}} C(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \langle\psi(\boldsymbol{\theta})|\mathcal{H}|\psi(\boldsymbol{\theta})\rangle \geq \lambda_0$$

- Any VQA can be decomposed in **five submodules**:

  1) Initialization
  2) Parameterized circuit
  3) Cost evaluation
  4) Classical optimizer
  5) Adjust ansatz parameters and re-run



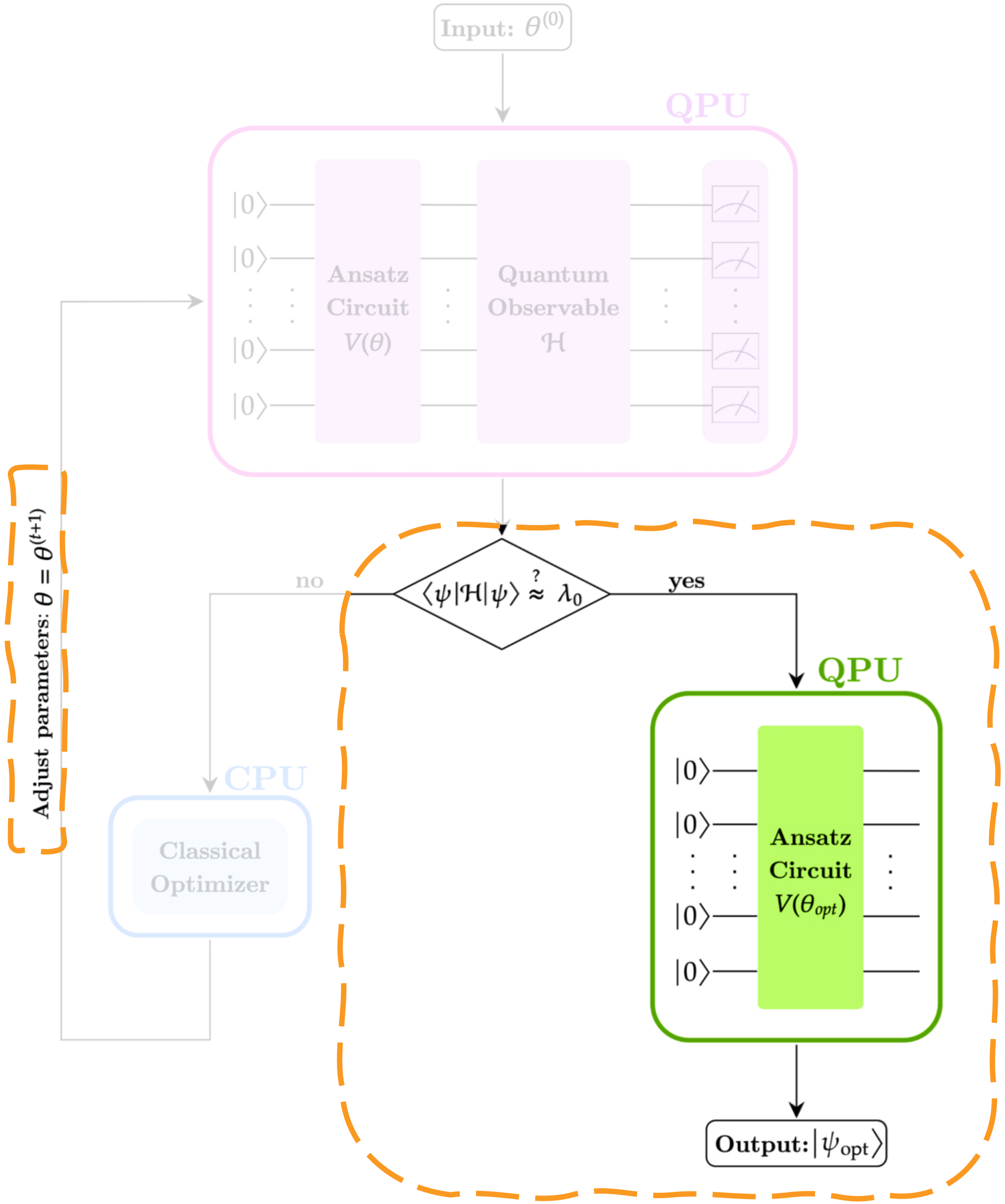
VQAs

- VQAs use a classical optimizer to minimize a cost function by training a **parameterized quantum circuit**.
- The cost function is defined as

$$\min_{\boldsymbol{\theta}} C(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \langle\psi(\boldsymbol{\theta})|\mathcal{H}|\psi(\boldsymbol{\theta})\rangle \geq \lambda_0$$

- Any VQA can be decomposed in **five submodules**:

  1) Initialization
  2) Parameterized circuit
  3) Cost evaluation
  4) Classical optimizer
  5) Adjust ansatz parameters and re-run



VQAs

- VQAs use a classical optimizer to minimize a cost function by training a **parameterized quantum circuit**.
- The cost function is defined as

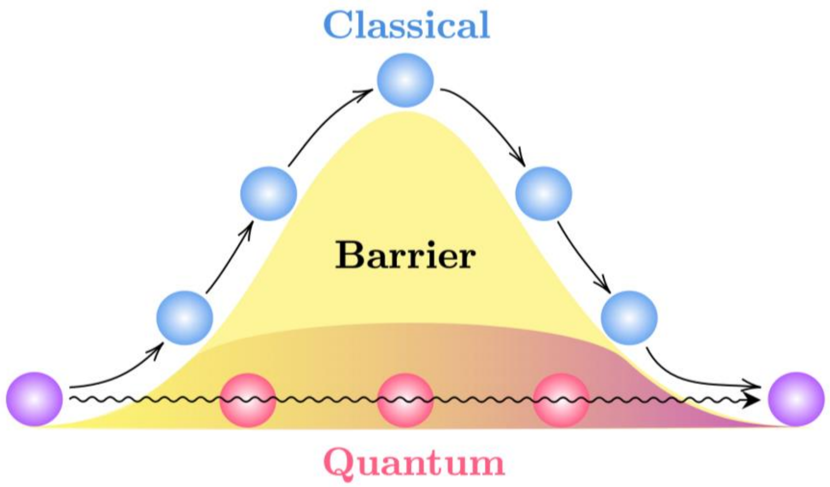
$$\min_{\boldsymbol{\theta}} C(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \langle\psi(\boldsymbol{\theta})|\mathcal{H}|\psi(\boldsymbol{\theta})\rangle \geq \lambda_0$$

- Any VQA can be decomposed in **five submodules**:

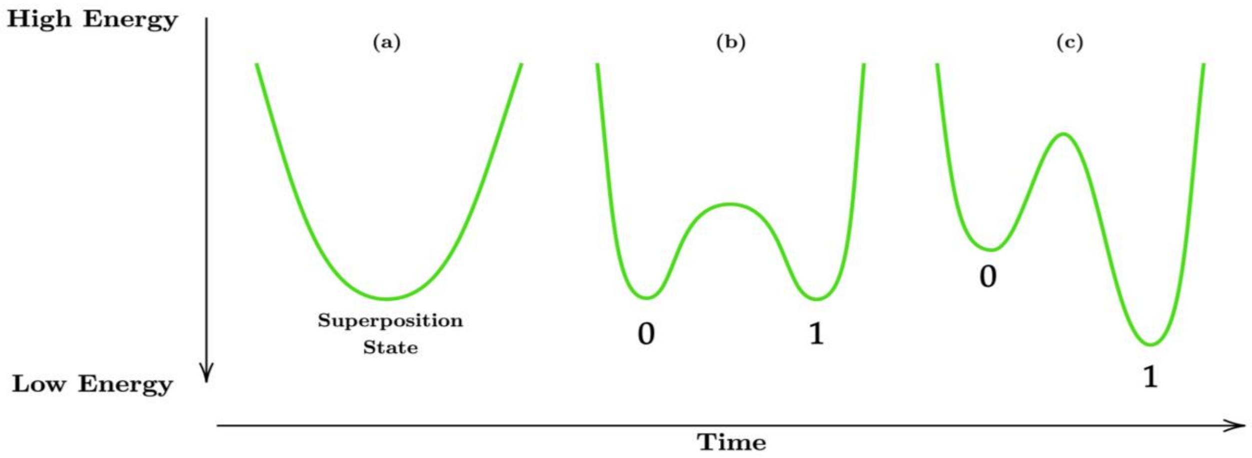
  1) Initialization
  2) Parameterized circuit
  3) Cost evaluation
  4) Classical optimizer
  5) Adjust ansatz parameters (and re-run)

Quantum Tunneling

- Simulated annealing (SA) is a classical heuristic to explore the landscape of a cost function, searching for the global minimum. It uses a **temperature** parameter to move stochastically in the landscape.
- If the landscape has wells too deep, the SA may **get stuck** in a local minimum.
- Using **quantum tunnelling**, it is possible to define a quantum version of the SA to have a positive probability to escape local minima regardless of the barrier height.



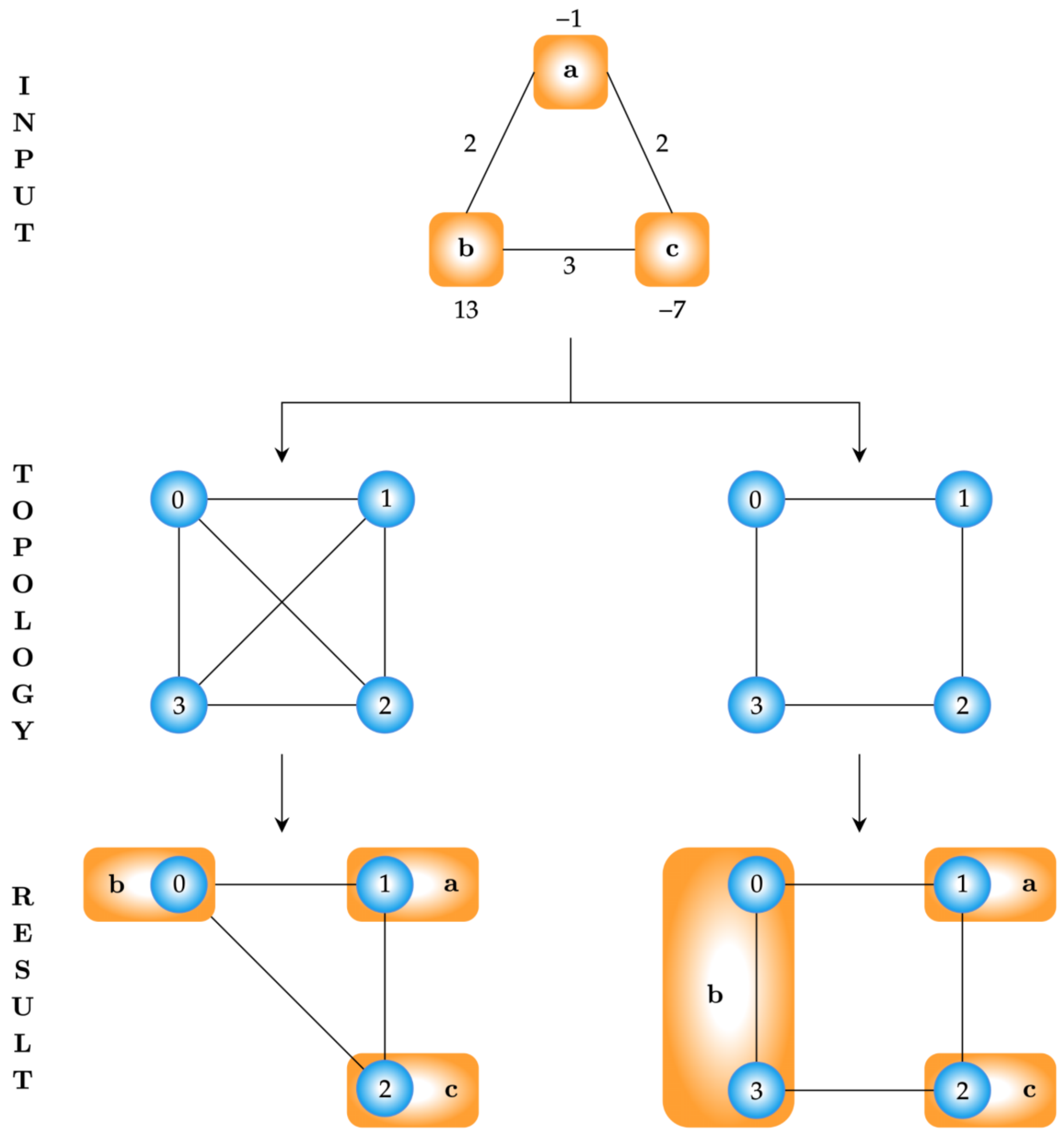
Quantum Annealing



- The idea is to introduce artificial **quantum fluctuations $\Gamma(t)\mathcal{H}_{kin}$** regulated through the parameter $\Gamma(t)$, that is slowly decreases in time.
- The **total Hamiltonian** of the system is

$$\mathcal{H} = \mathcal{H}_P + \Gamma(t)\mathcal{H}_{kin}$$

- If we initialize our system in the ground state of $\mathcal{H}_{kin}$, the **adiabatic theorem** guarantees the system will end in the ground state of $\mathcal{H}_P$.



Minor Embedding

- Given a problem Hamiltonian $\mathcal{H}_P$, the input to the QPU will be its associated **graph**.
- The input graph is then embedded into the QPU through a **minor embedding**.
- Since real hardware has a limited connectivity, representing one logical qubit may require multiple physical qubits. These groups of qubits are called **chains**.
- Chains must be kept intact during the annealing procedure. This is achieved by tuning an hyperparameter called **chain strength**.

Variational Quantum Linear Solver (VQLS)

- Suppose we want to solve a **real-valued linear system** of the form

$$\mathbf{A}\mathbf{x} = \mathbf{b} \Leftrightarrow \mathbf{A}|x\rangle = |b\rangle$$

where we assume $\mathbf{A}$ can be written as a linear combination of unitary operators:

$$\mathbf{A} = \sum_{l=0}^{L} c_l A_l$$

- The goal of the algorithm is to find an optimal set of parameters $\boldsymbol{\theta}_{opt}$ such that

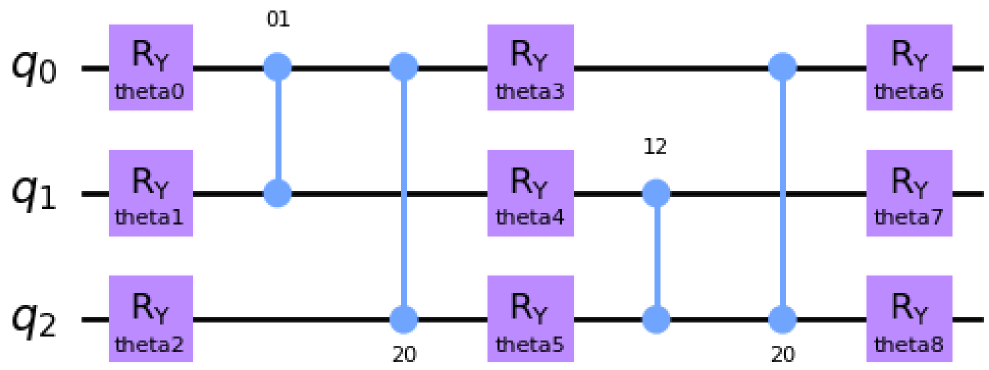
$$\mathbf{A}|x_{opt}\rangle \equiv \mathbf{A}|x(\boldsymbol{\theta}_{opt})\rangle \propto |b\rangle$$

- In our experiments, we used the system defined by

$$\mathbf{A} = 0.55\mathbb{I} + 0.225\boldsymbol{Z}_2 + 0.225\boldsymbol{Z}_1 \qquad\qquad |b\rangle = \mathbf{H}^{\otimes 3}|0\rangle$$

VQLS

- We use a **Hardware Efficient Ansatz** paired with a default reference state.



- As four our parameters $\boldsymbol{\theta}$, we test **four different initialization** techniques:

	Zero Vector	**Uniform Vector**	**Normal Distribution**	**Random Vector**
$\boldsymbol{\theta}^{(0)}$	$\begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$	$\begin{pmatrix} 1/N \\ \vdots \\ 1/N \end{pmatrix}$	Obtained through `np.random.normal(0,1)`	Obtained through `np.random.rand()`

VQLS

Global cost

- We can define a global cost by measuring the **orthogonality** of our estimate, i.e.

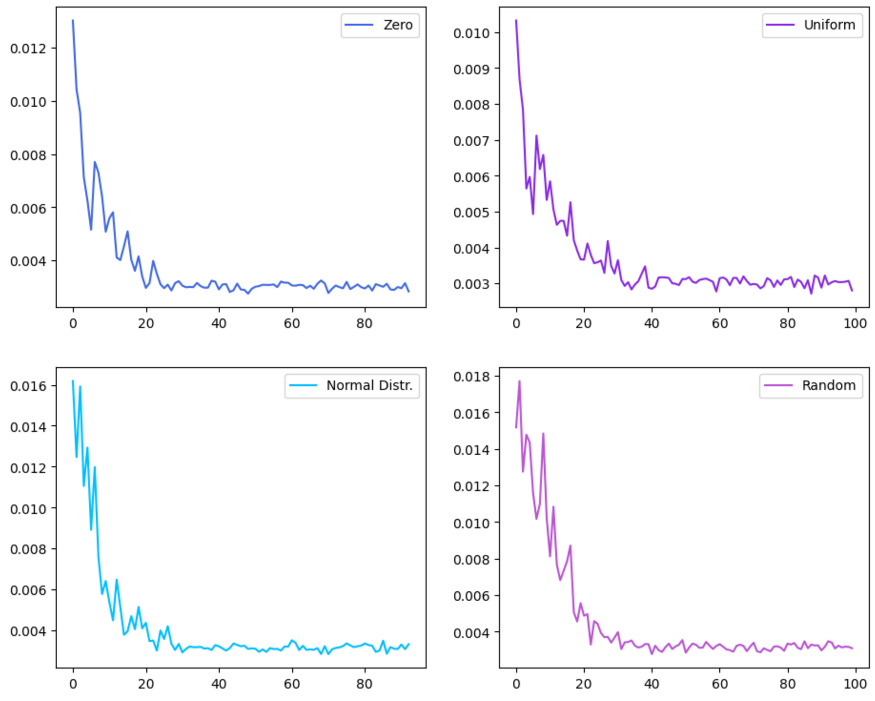
$$\hat{C}_G = \langle x|\mathcal{H}_G|x\rangle$$

where

$$\mathcal{H}_G = \mathbf{A}^\dagger(\mathbb{I} - |b\rangle\langle \mathrm{b}|)\mathbf{A}$$

- A **normalized** version is needed:

$$C_G = \frac{\langle x|\mathcal{H}_G|x\rangle}{\langle\psi|\psi\rangle} = 1 - \frac{|\langle b|\psi\rangle|^2}{\langle\psi|\psi\rangle}$$

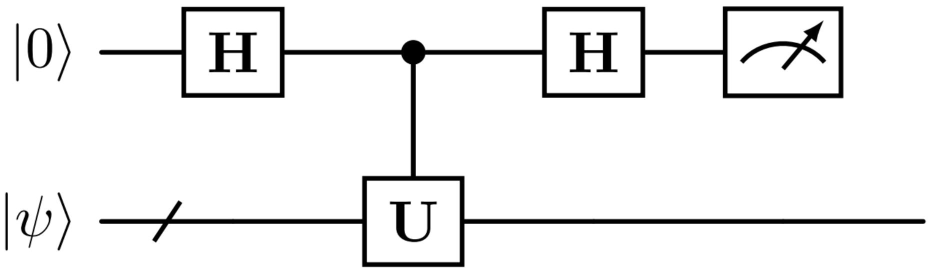


I
D
E
A
L

VQLS

- To estimate the cost function, we used a quantum subroutine called **Hadamard test**.
- Using this test, we can estimate any real operator $\mathbf{U}$ as

$$\mathrm{Re}(\langle\psi|\mathbf{U}|\psi\rangle) = 1 - 2p_1$$



- To use Hadamard tests, we decompose the terms we need to estimate:

$$\langle\psi|\psi\rangle = \sum_{ij} c_i c_j^* \langle 0|\mathbf{V}^\dagger \mathbf{A}_j^\dagger \mathbf{A}_i \mathbf{V}|0\rangle = \sum_{ij} c_i c_j^* \beta_{ij}$$

$$|\langle b|\psi\rangle|^2 = \sum_{ij} c_i c_j^* \langle 0|\mathbf{U}_b^\dagger \mathbf{A}_i \mathbf{V}|0\rangle\langle 0|\mathbf{V}^\dagger \mathbf{A}_j^\dagger \mathbf{U}_b|0\rangle = \sum_{ij} c_i c_j^* \gamma_{ij}$$

VQLS

Local cost

- To avoid **trainability issues**, we can define a local version of $\hat{C}_G$:

$$\hat{C}_L = \langle x|\mathcal{H}_L|x\rangle$$

where

$$\mathcal{H}_L = \mathbf{A}^\dagger \mathbf{U}_b \left(1 - \frac{1}{n}\sum_{k=1}^{n} |0_k\rangle\langle 0_k| \otimes \mathbb{I}_{\bar{k}}\right)\mathbf{U}_b^\dagger \mathbf{A}$$

- Once again, we **normalize** the cost function:

$$C_L = \frac{\langle x|\mathcal{H}_L|x\rangle}{\langle\psi|\psi\rangle}$$

Using the linearity of the expected value and the fact that

$$|0_k\rangle\langle 0_k| = \frac{\mathbb{I}_k + \mathbf{Z}_k}{2}$$

we can expand $\langle x|\mathcal{H}_L|x\rangle$ and rewrite C_L as

$$C_L = \frac{\langle x|\mathcal{H}_L|x\rangle}{\langle\psi|\psi\rangle} =$$

$$= \frac{1}{2} - \frac{1}{2n\langle\psi|\psi\rangle}\sum_{k=1}^{n}\sum_{ij} c_i c_j^* \, \delta_{ij}^{(k)}$$

where

$$\boldsymbol{\delta_{ij}^{(k)}} = \langle \mathbf{0}|\boldsymbol{V^\dagger A_j^\dagger U_b (Z_k} \otimes \mathbb{I}_{\bar{k}}\boldsymbol{) U_b^\dagger A_i V}|\mathbf{0}\rangle$$

VQLS

Local cost

- To avoid **trainability issues**, we can define a local version of $\hat{C}_G$:

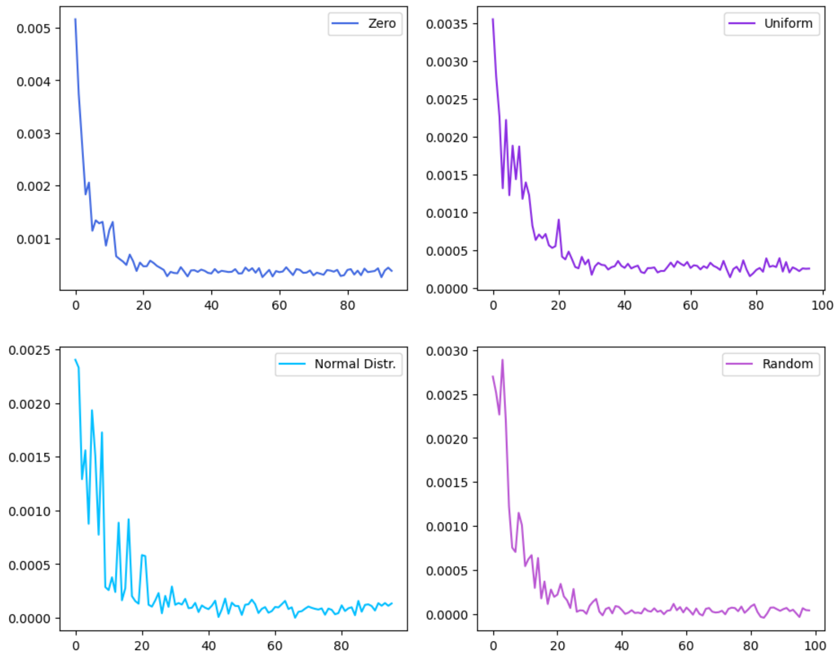
$$\hat{C}_L = \langle x|\mathcal{H}_L|x\rangle$$

where

$$\mathcal{H}_L = \mathbf{A}^\dagger \mathbf{U}_b \left(1 - \frac{1}{n}\sum_{k=1}^{n} |0_k\rangle\langle 0_k| \otimes \mathbb{I}_{\bar{k}} \right) \mathbf{U}_b^\dagger \mathbf{A}$$

- Once again, we **normalize** the cost function:

$$C_L = \frac{\langle x|\mathcal{H}_L|x\rangle}{\langle \psi|\psi\rangle}$$



I
D
E
A
L

VQLS

The results for both cost functions in the **ideal simulations** are reported in the following table:

Initialization	ζ_{global}	ζ_{local}
Zero vector	0.7486162	0.78634465
Uniform vector	0.7765784	0.7876895
Normal Distribution	0.7145117	0.8834687
Random vector	0.7466224	0.9157983

As for the **noisy simulation**:

Initialization	ζ_{global}	ζ_{local}
Random vector	0.6392252	0.8224284

MQ Boolean Problems

- Another interesting class of systems of equations is the one of **Multivariate Quadratic (MQ) Boolean** equations.
- The input to a MQ problem consists of m quadratic polynomials $p_1(\mathbf{x}), \dots, p_m(\mathbf{x})$ in n variables $\mathbf{x} = (x_1, \dots, x_n)$ and coefficients in the binary field $\mathbb{F}_2$.
- We want to find a vector $\mathbf{s}$ such that $p_i(\mathbf{s}) = 0$ for all $i = 1, \dots, m$.
- A **direct approach** to encode the problem in a cost function is to penalize with positive energy each equation that is not satisfied:

$$\mathcal{H}_P = \sum_{i=1}^{m} p_i(\mathbf{x})$$

Direct Encoding

- A **direct approach** to encode the problem in a cost function is to penalize with positive energy each equation that is not satisfied:

$$\mathcal{H}_P = \sum_{i=1}^{m} p_i(\mathbf{x})$$

- The polynomials $p_i(\mathbf{x})$ must be **converted** from the given ANF to the relative NNF by applying the substitution

$$(x_i + x_j) \mapsto x_i + x_j - 2x_i x_j$$

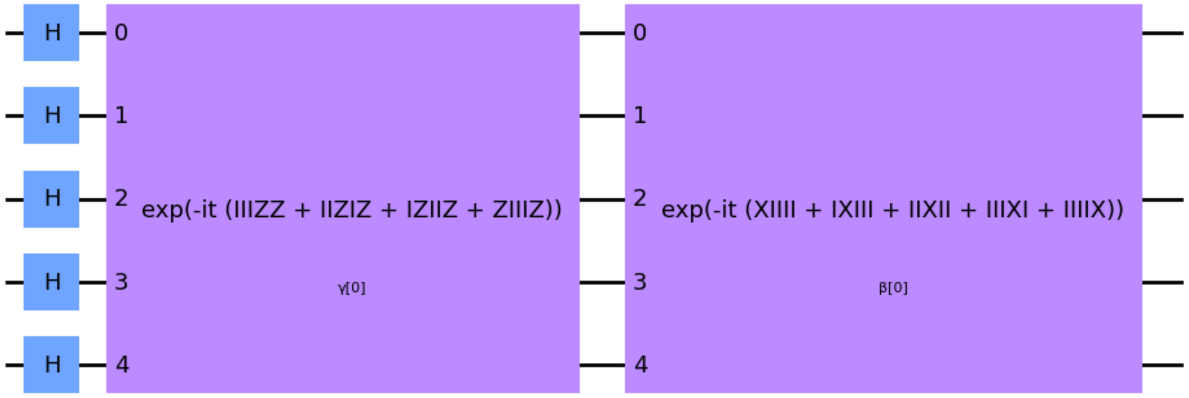
- This conversion, though, generates **multi-qubit interactions** of degree > 2, which cannot be executed on quantum hardware.
- The cost is then reduced to a two-body Hamiltonian by using **ancillary variables** and a penalization term that includes those ancillae.

QAOA

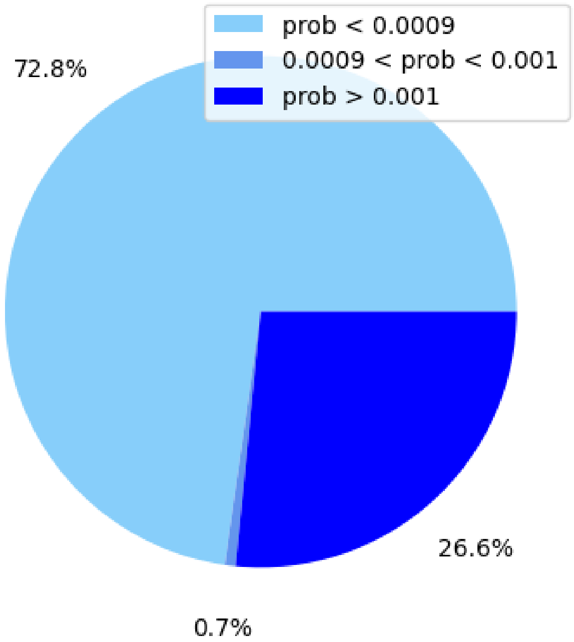
- The Quantum Approximate Optimization Algorithm (QAOA) is a variational algorithm that uses a **specific ansatz** (also named QAOA).
- The QAOAnsatz is inspired by an **approximated adiabatic transformation**, where the order p of the approximation determines the precision of the solution.
- The ansatz is defined as

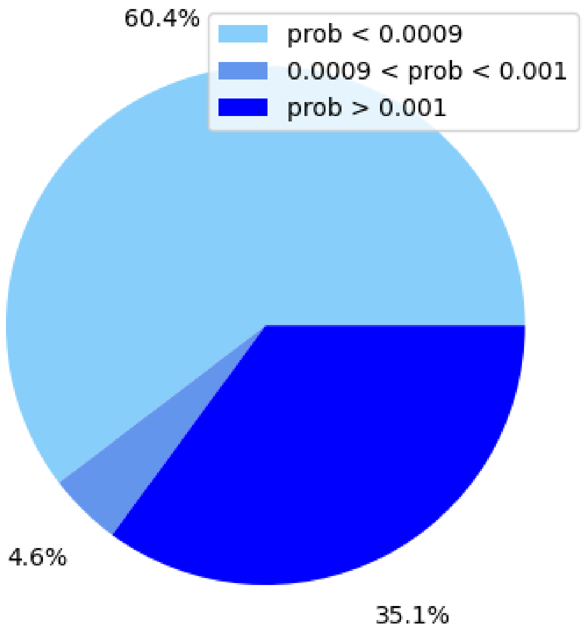
$$\mathbf{V}(\boldsymbol{\gamma}, \boldsymbol{\beta}) = \prod_{l=1}^{p} e^{-i\beta_l \mathcal{H}_M} e^{-i\gamma_l \mathcal{H}_P}$$

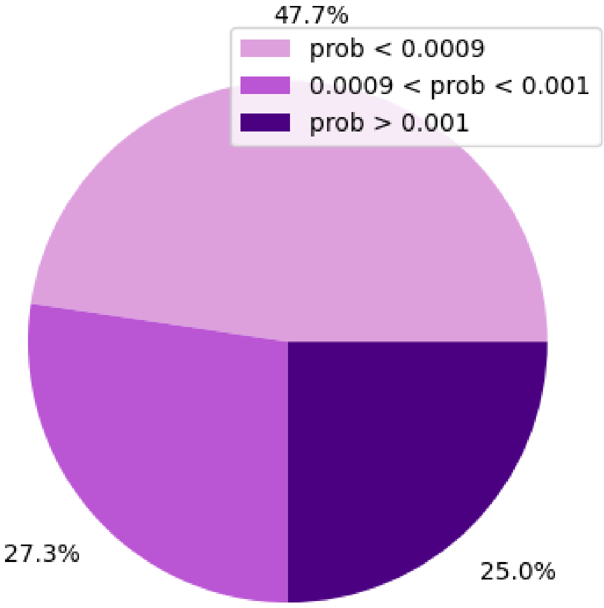
  where $\mathcal{H}_P$ is the problem Hamiltonian and $\mathcal{H}_M$ is the mixing Hamiltonian.



- We use QAOAlgorithm to solve an MQ instance with five variables (henceforth, **MQ5**).







IDEAL

probability : 0.013212

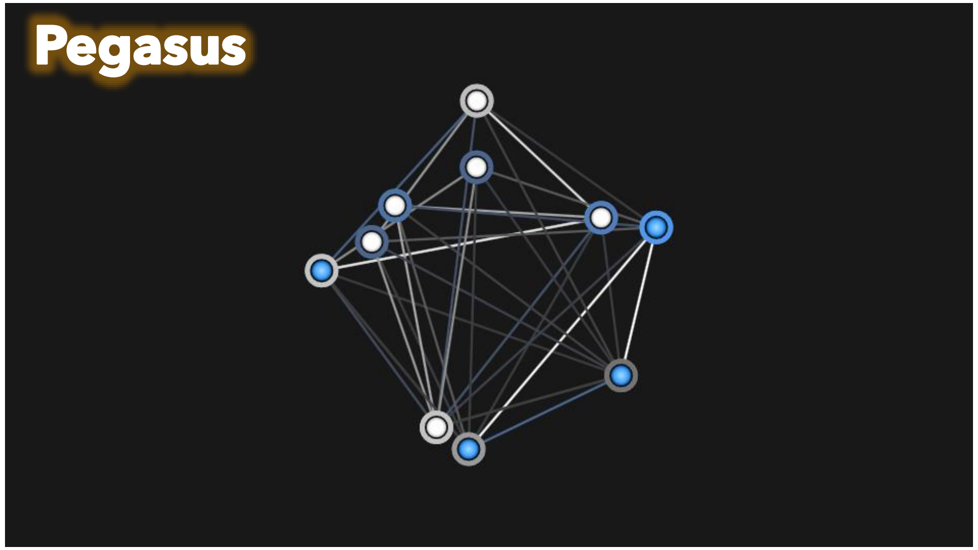
NOISY

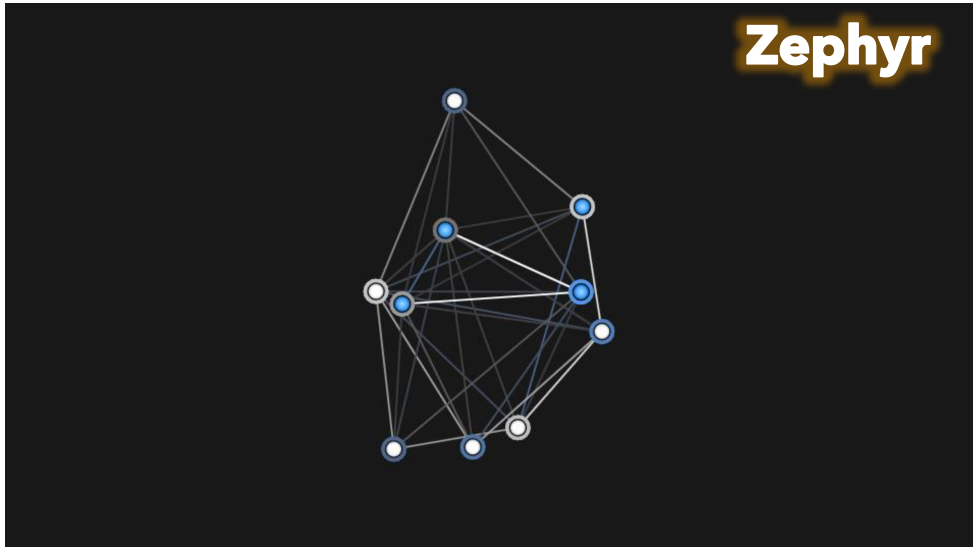
probability : 0.00293

QAOA

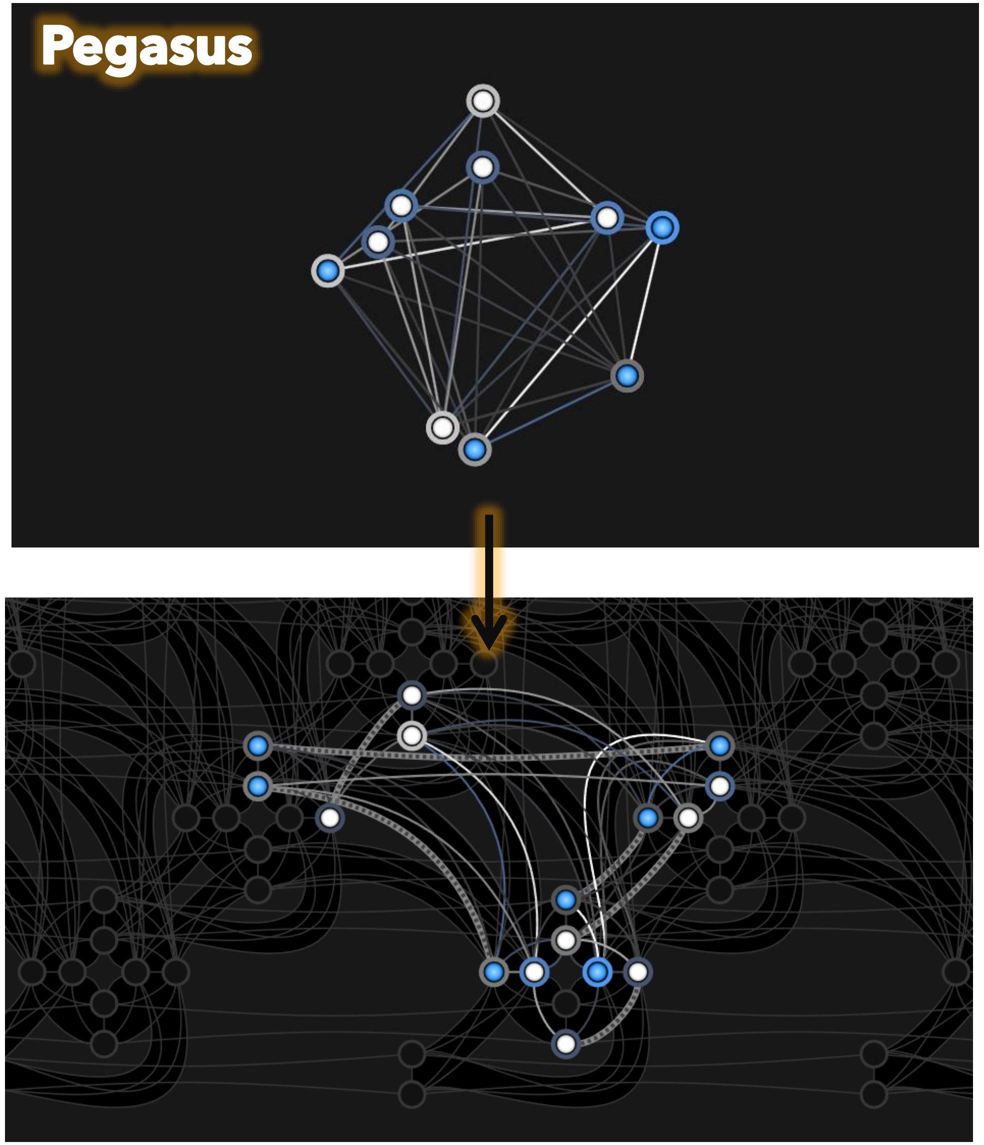
MQ5 through Quantum Annealing

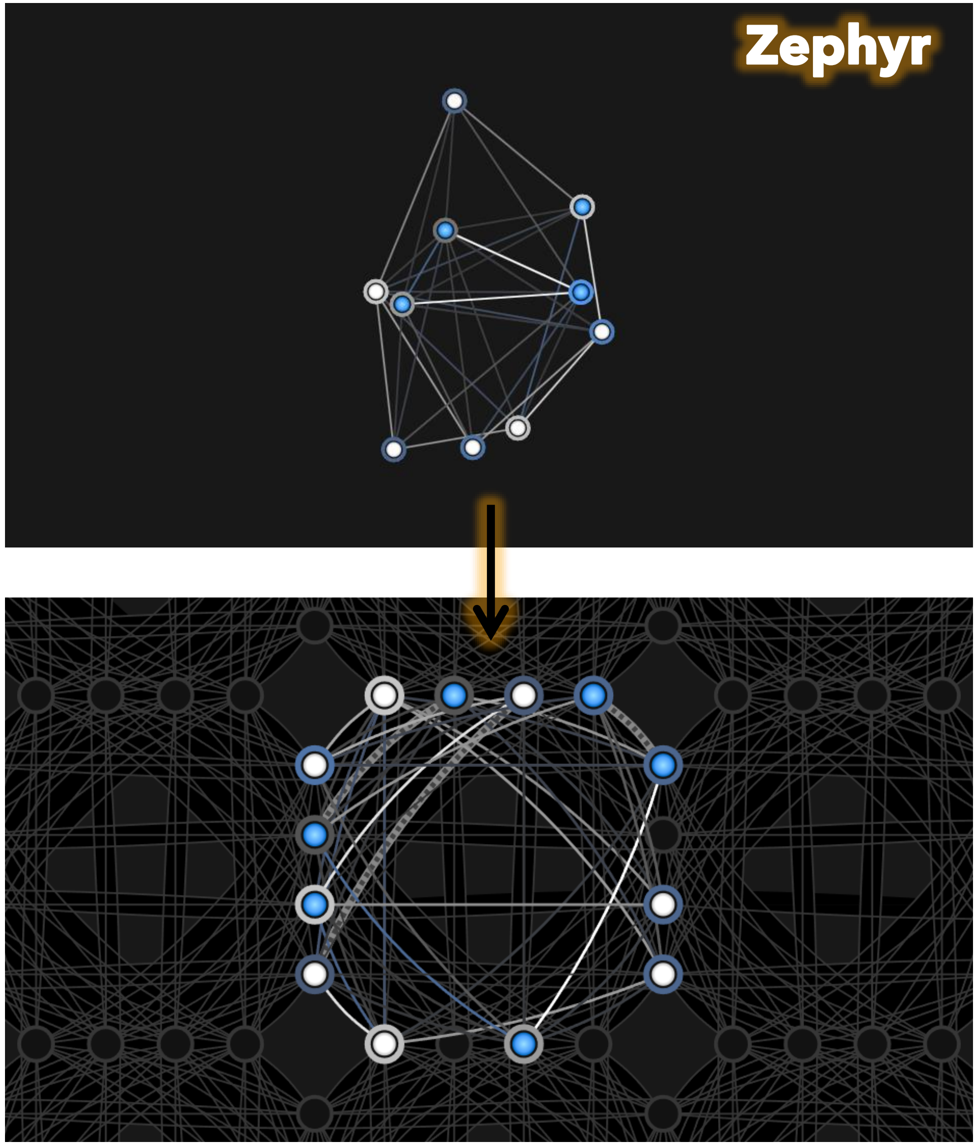
- We start the tests on quantum annealing solving the same **MQ5** instance used in the QAOA experiment.
- All the experiments run on two different topologies: **Pegasus** and **Zephyr**.



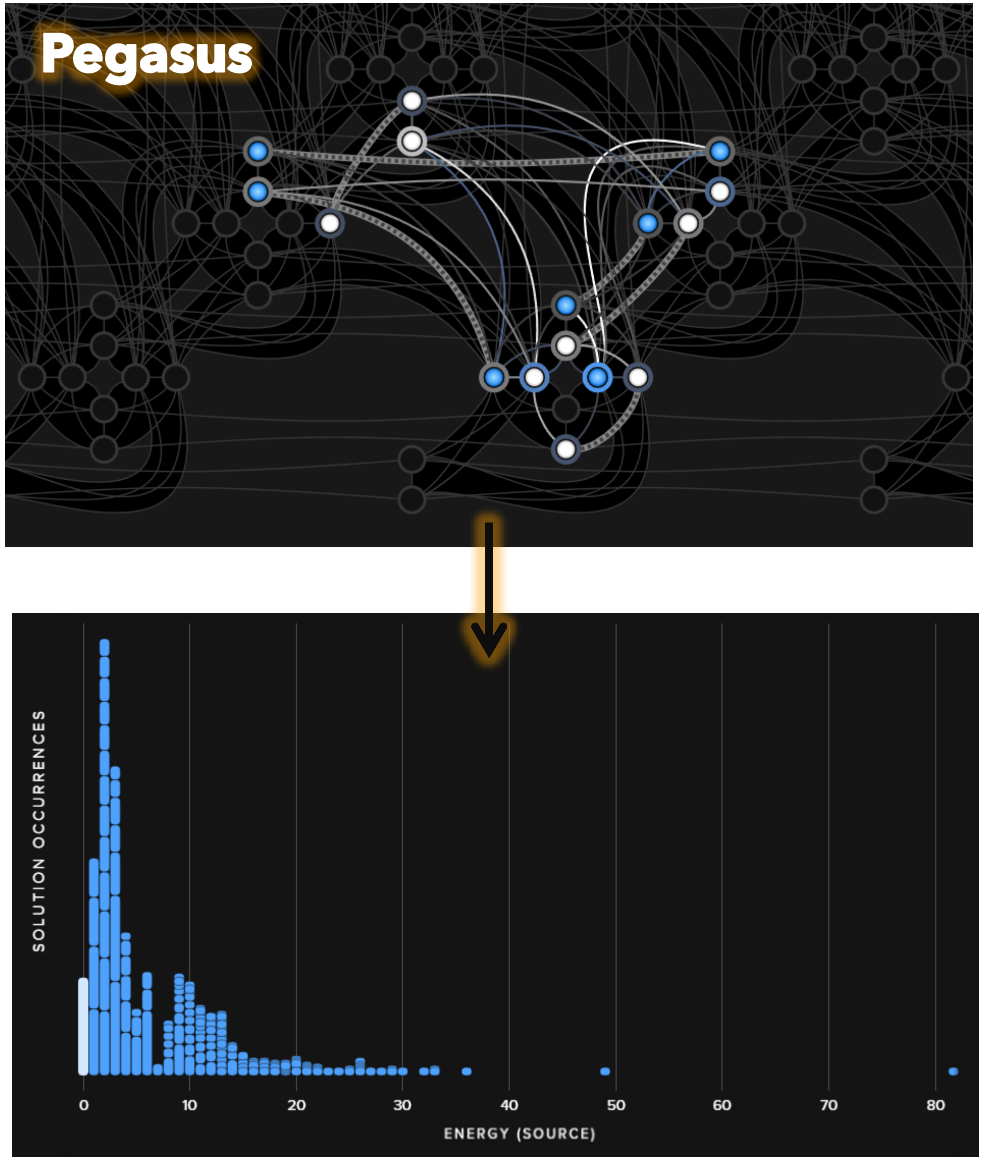


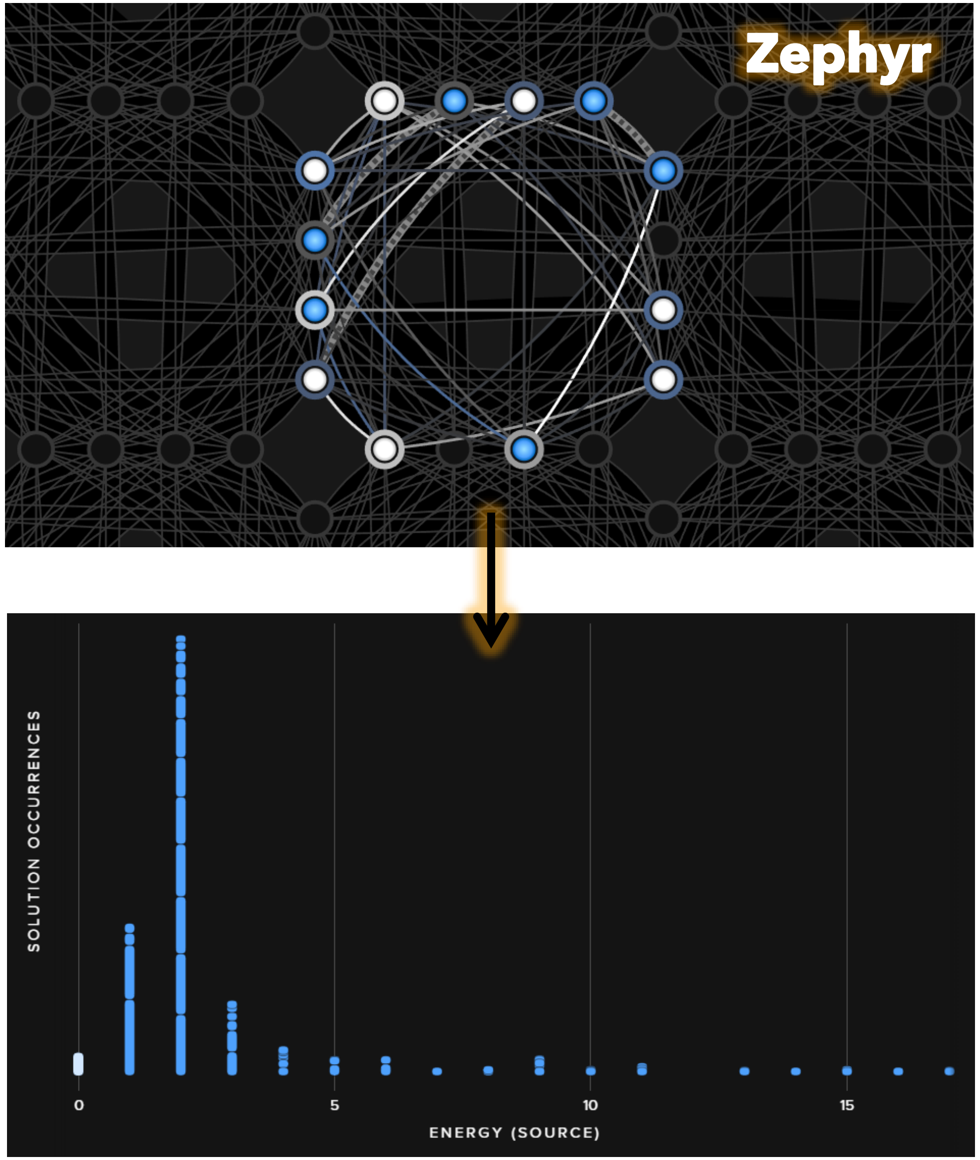
MQ5 through QA





MQ5 through QA

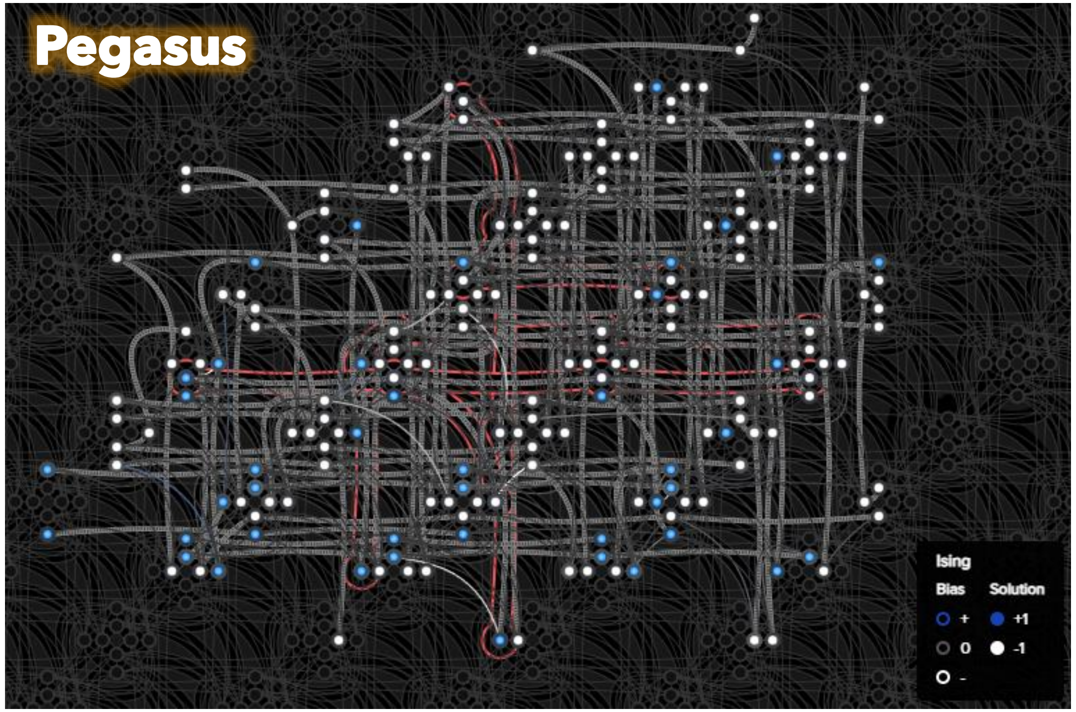


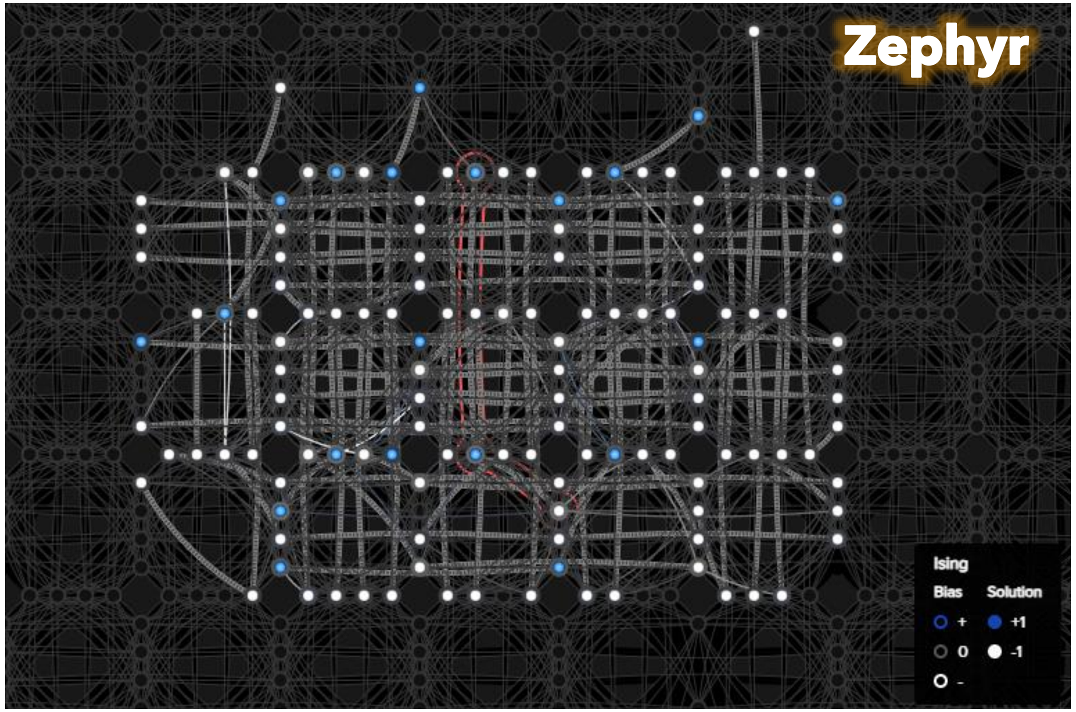


MQ9 and the Iterative Method

- Increasing the number of variables may negatively impact the search for the ground state.
- We can use an iterative routine that **fixes ancillae** based on the most common sampled value:
    1) At every iteration, the algorithm checks the first k samples, ordered in ascending order based on the **associated energy value**.
    2) If an ancilla has the same value in all these k samples, then said ancilla is substituted by the sampled value.
    3) Repeat until convergence or until the stopping criteria is met.
- We use this method to solve an **MQ** instance with **nine variables**, encoded using the **direct approach**.

MQ5 through QA





Conclusions and Future Work

VQLS → Fast-and-Slow Algorithm

- The algorithm requires multiple runs to achieve an acceptable accuracy level.
- It may be useful to test a different initialization technique to not get stuck in a local minima.

QAOA → Grover Adaptive Search + more noisy simulations

- The main limitation of this approach is the size of the problem and the depth of the circuit.
- Further analysis and tests on noisy runs to see if a pattern can be inferred.

Quantum Annealing → MinRank Problem + more BQMs analysis

- Most performant quantum computing paradigm.
- Though with some preprocessing, it could be used to develop an algebraic attack.

General → Metalearning

- Run a statistical analysis on different BQMs associated to the same problem.

Main References

[NC10] Michael A Nielsen and Isaac L Chuang. Quantum computation and quantum information. Cambridge university press, 2010

[Sip96] Michael Sipser. “Introduction to the Theory of Computation”. In: ACM Sigact News 27.1 (1996), pp. 27–29.

[IBMd] IBM Quantum. IBM Quantum Learning - Variational algorithm design. Last accessed February 14, 2024. url: https://learning.quantum.ibm.com/course/variational-algorithm-design.

[Bra+23] Carlos Bravo-Prieto et al. “Variational quantum linear solver”. In: Quantum 7 (2023), p. 1188.

[FGG14] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. “A quantum approximate optimization algorithm”. In: arXiv preprint arXiv:1411.4028 (2014).

[KN98] Tadashi Kadowaki and Hidetoshi Nishimori. “Quantum annealing in the transverse Ising model”. In: Physical Review E 58.5 (1998), p. 5355.

[Ram+22] Sergi Ramos-Calderer et al. “Solving systems of Boolean multivariate equations with quantum annealing”. In: Physical Review Research 4.1 (2022), p. 013096.

Thank you for your attention!