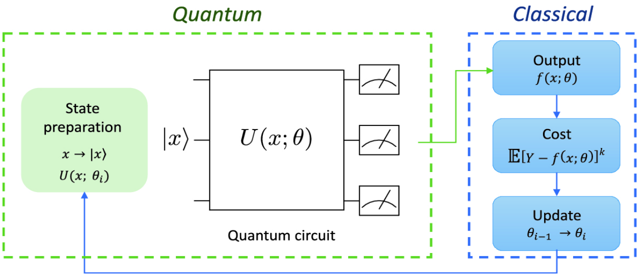
Introduction to Variational Quantum Algorithms: QAOA

Ruben Pariente Bassa

07 October 2025

Variational Algorithms for Noisy Quantum Devices (NISQ)

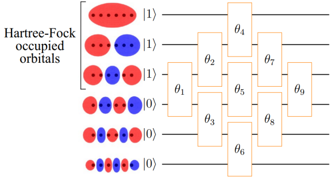
- Quantum advantage is still limited by noise and number of qubit counts: The idea is to combine quantum circuits with classical optimization.
- Variational Quantum Algorithms (VQAs) use a parameterized quantum circuit to prepare a trial wavefunction/quantum state $|\psi(\theta)\rangle = U(\theta)|0\rangle$ and a classical optimizer to minimize a cost function $C(\theta)$



SINTEF

Applications of Variational Quantum Algorithms (VQAs)

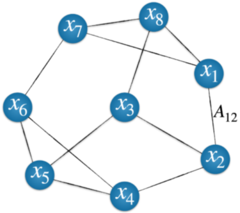
VQE:Quantum Chemistry



$$H = \sum_{ij} h_{ij} c_i^\dagger c_j + \sum_{i<j,k<l} h_{ijkl} c_i^\dagger c_j^\dagger c_k c_l$$

$$N = \sum_i c_i^\dagger c_i, \qquad [N, H] = 0$$

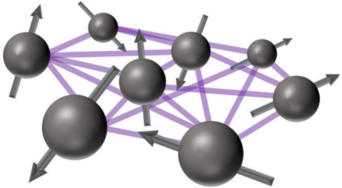
QAOA:Combinatorial Optimization



$$\max_{x \in \{0,+1\}^{|V|}} \quad x^t A x$$

s.t Bx=c

VQS:Quantum Spin Simulation



$$H = -J \sum_{(i,j)\in E(G)} \sigma_i^z \sigma_j^z + h \sum_i \sigma_i^x$$

Observables

- An observable H is a **self-adjoint/Hermitian** operator on the Hilbert space $(\mathbb{C}^2)^{\otimes n}$. This means $H^\dagger = H$
- Spectral theorem: $\exists$ orthonormal **basis** $\{|\psi_i\rangle\}_i$ of $(\mathbb{C}^2)^{\otimes n}$ consisting of eigenvectors of H, and all eigenvalues λ_i are **real**.
- We can write: $H = \sum_i \lambda_i |\psi_i\rangle \langle\psi_i|$
- To each energy λ_j corresponds to an **energy eigenstate**.
  - **ground state**: energy eigenstate $|v_1\rangle$ corresponding to the lowest energy
  - **first excited state**, **second excited state**, ...: $|v_2\rangle, |v_3\rangle, \ldots$

Expectation values

Given

- a state $|\phi\rangle$ prepared on a quantum computer using the unitary U such that $U|0\rangle = |\phi\rangle$
- an observable H we are interested to measure

Then the expectation value of H respect to the state $|\phi\rangle$ is given by

$$\langle H\rangle_{|\phi\rangle} := \langle\phi| H |\phi\rangle = \langle 0| UHU^{\dagger} |0\rangle \tag{1}$$

From the spectral theorem it follows:

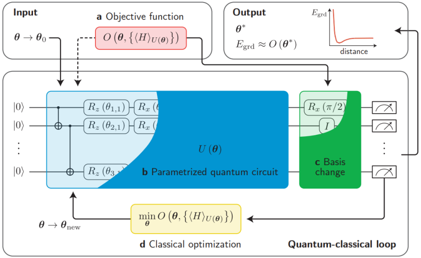
$$\langle H\rangle_{|\phi\rangle} = \langle\phi| \sum_i \lambda_i |\psi_i\rangle \langle\psi_i|\phi\rangle = \sum_i \lambda_i |\langle\phi|\psi_i\rangle|^2 \tag{2}$$

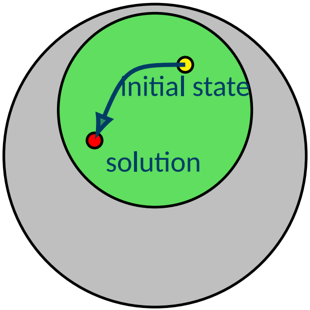
Particularly: $\langle H\rangle_{|\psi_i\rangle} = \lambda_i$

The Variational Principle

$$\langle H \rangle_{|\phi\rangle} = \sum_i \lambda_i \, |\langle \phi | \psi_i \rangle|^2 \geq \sum_i \lambda_{\text{min}} \, |\langle \phi | \psi_i \rangle|^2 = \lambda_{\text{min}} \tag{3}$$

- Find θ^* s.t. $\langle H \rangle_{|\phi(\theta^*)\rangle}$ minimal
- $H = \sum_\alpha w_\alpha \vec{\sigma_\alpha}, \quad \vec{\sigma_\alpha} \in \{I, X, Y, Z\}^{\otimes N}$
- $E_{VQE} = min_{\vec{\theta}} \sum_\alpha w_a \left\langle \psi(\vec{\theta}) \middle| \vec{\sigma_\alpha} \middle| \psi(\vec{\theta}) \right\rangle$





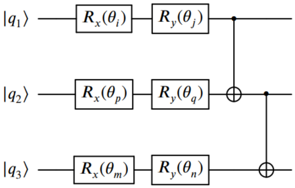
The Ansatz

The right choice of ansatz is critical to obtain a solution that is close to the ground state.

- **Expressability**: Refers the range of feasible states that the ansatz can achieve.
- **Trainability**: Refers to the ability to find the best set of parameters of the ansatz respect to expectation values of the Hamiltonian in a finite amount of time.
- **Depth**: Refers to the number of sequential operations required for the implementation, which impacts the overall runtime of the method and its resilience to noise

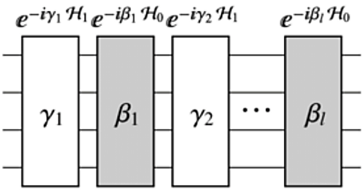
Hardware Efficient Ansatz

$$|\psi(\theta)\rangle_{HEA} = \prod_{i=1}^{p} U_{ent} U_{rot}(\theta_i) |0\rangle$$



Hamiltonian Variational Ansatz

$$|\psi(\theta)\rangle = \prod_{l=1}^{p} \left(\prod_j e^{i\theta_{lj} H_j}\right) |\psi_0\rangle, H = \sum_j H_j$$



SINTEF

The Classical Optimizer choice

Gradient Descent Based

Use the analytical property of the ansatz, the gradient of observables can be directly computed on a quantum computer.

Gradient Descent,
Quantum Natural Gradient

Stochastic Gradient Based

Approximated the true gradient using random sampled data at each iteration.

SPSA,QNSPSA,Adam

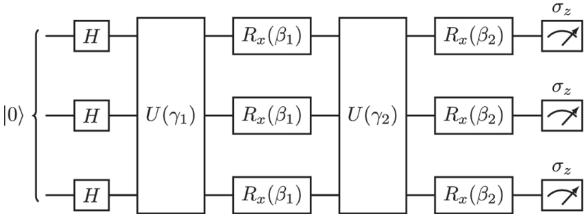
Gradient-free searching

Do not rely on gradient information and instead explore the parameter space using alternative techniques as random search, evolutionary algorithms or Bayesian optimization.

COBYLA, Nelder-Mead

The Quantum Alternating Operator Ansatz

- Objective function $f : \{0, 1\}^n \rightarrow \mathbb{R}$
- Where are looking for the optimal vector $x^* = argmin_{x\in\{0,1\}}f(x)$
- Encode each binary string into a quantum state: $z = \{0, 1\}^n \rightarrow |z\rangle$
- Encode the objective function into a problem Hamiltonian
  $H_P |z\rangle = f(z) |z\rangle \,, \quad \langle H_P \rangle_{|z\rangle} = f(z)$
- The Ground state of H_P correspond to the minima of the the objective function.
- $\left|\vec{\gamma}, \vec{\beta}\right\rangle = U_M(\beta_p)U_P(\gamma_p) \cdots U_M(\beta_1)U_P(\gamma_1) |\phi_0\rangle \,, U_P(\gamma) = e^{i\gamma H_P}, U_M = \prod_{i=1}^n RX_i(\beta)$
- Find $\vec{\gamma}, \vec{\beta} \in \mathbb{R}^p$, such that $\langle \gamma, \beta| H_P |\gamma, \beta\rangle$ is minimized.



Combinatorial Optimization Example: The Max-k-Cut Problem

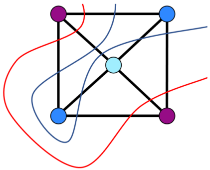
$$\max_{\mathbf{x}\in\{1,\dots,k\}^n} C(\mathbf{x}), \qquad C(\mathbf{x}) = \sum_{(i,j)\in E} w_{ij} \begin{cases} 1, & \text{if } x_i \neq x_j \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Solving NP hard optimization problems.

- **Heuristic algorithms**. No polynomial run time guarantee; appear to perform well on some instances.
- **Approximate algorithms**. Efficient and provide provable guarantees. With high probability we get a solution x^* such that

$$\frac{C(x^*) - \min_x C(x)}{\max_x C(x) - \min_x C(x)} \geq \alpha, \tag{5}$$

where $0 < \alpha \leq 1$ is the approximation ratio.

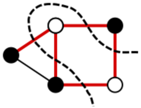


k	2	3	4	5
α	.878567	.836008	.857487	.876610

k	6	7	8	9
α	.891543	.903259	.912664	.920367

SINTEF

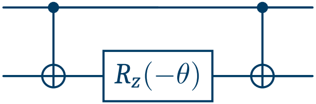
The MaxCut Implementation



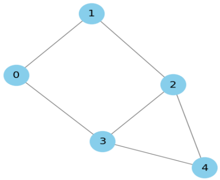
$$\widehat{H}_e = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} = |01\rangle\langle 01| + |10\rangle\langle 10| = \frac{\mathbb{I} - Z \otimes Z}{2} \tag{6}$$

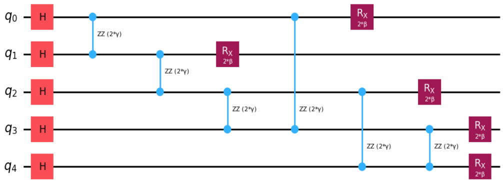
$$H_{Maxcut} = \sum_{(i,j)\in E} w_{ij} \frac{1 - Z_i Z_j}{2} \Rightarrow e^{i\theta H_{maxcut}} = \prod_{(i,j)\in E} e^{i\frac{\theta}{2} w_{ij} Z_i Z_j} \tag{7}$$

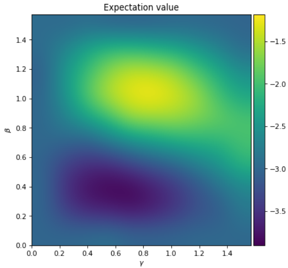
$$e^{-i\theta Z\otimes Z} = \begin{pmatrix} e^{-i\theta/2} & 0 & 0 & 0 \\ 0 & e^{i\theta/2} & 0 & 0 \\ 0 & 0 & e^{i\theta/2} & 0 \\ 0 & 0 & 0 & e^{-i\theta/2} \end{pmatrix} = \tag{8}$$

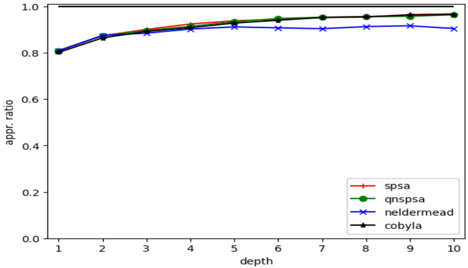


Example: Solving Max-Cut with QAOA









QAOA for Constrained Optimization Problems

The solutions constrained to a feasible subspace $\mathrm{span}(B) \subset \mathcal{H} = (\mathbb{C}^2)^{\otimes n}$:

$$B = \left\{ \left|z_j\right\rangle, \;\; 1 \leq j \leq J, \;\; z_j \in \{0,1\}^n \right\}. \tag{9}$$

Definition valid mixer

- Preserve the feasible subspace

$$U_M(\beta) \left|v\right\rangle \in \mathrm{span}(B), \quad \forall \left|v\right\rangle \in \mathrm{span}(B), \forall \beta \in \mathbb{R}, \tag{10}$$

- Provide transitions between all pairs of feasible states, i.e., for each pair of computational basis states $\left|x\right\rangle, \left|y\right\rangle \in B$ there exist $\beta^* \in \mathbb{R}$ and $r \in \mathbb{N} \cup \{0\}$, such that

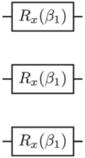
$$\left| \left\langle x \right| \underbrace{U_M(\beta^*) \cdots U_M(\beta^*)}_{r \text{ times}} \left|y\right\rangle \right| > 0. \tag{11}$$

SINTEF

Example of Valid Mixers

Unconstrained case: X mixer

$$U_X(\beta) = \prod_i RX_i(\beta) = \prod_i (cos(\beta)\mathbb{I} + isin(\beta)X_i) \tag{12}$$

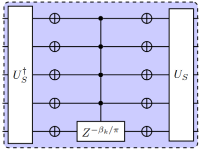


$$U_X(\tfrac{\pi}{2}) = \frac{1}{2\sqrt{2}}(\mathbb{I} + i(X_1 + X_2 + X_3) - (X_1X_2 + X_2X_3 + X_1X_3) - iX_1X_2X_3)$$

Constrained case: Grover mixer

$$|F\rangle = \frac{1}{\sqrt{|B|}} \sum_{i \in B} |i\rangle = U_S |0\rangle \Rightarrow \tag{13}$$

$$U_{Grover}(\beta) = e^{i\beta|F\rangle\langle F|} = U_S e^{i\beta|0\rangle\langle 0|} U_S^\dagger \tag{14}$$



$$(|F\rangle\langle F|)^2 = |F\rangle\langle F| \Rightarrow U_{Grover}(\beta) = \sum_i \frac{(i\beta)^n}{n!}(|F\rangle\langle F|)^n = \mathbb{I} + (e^{i\beta} - 1)|F\rangle\langle F|$$

Portfolio Optimization Problem

Motivation: The goal here is to decide which assets to include in a portfolio to balance risk and return. In the binary formulation, each asset is either included in the portfolio ($z_i = 1$) or excluded ($z_i = 0$).

Objective Function:

$$F(z_1, z_2, \ldots, z_n) = q \sum_{i,j=1}^{n} z_i z_j \sigma_{ij} \; - \; (1-q) \sum_{i=1}^{n} z_i \mu_i, \quad z_i \in \{0, 1\}.$$

- n: number of available assets
- σ_{ij}: covariance matrix of asset returns
- μ_i: expected return of asset i
- $q \in [0, 1]$: investor's risk preference
  - $q = 1$: fully risk-averse (minimize variance)
  - $q = 0$: fully risk-seeking (maximize return)

The Budget Constraint and XY Mixer

Budget Constraint:

$$\sum_{i=1}^{n} z_i = B,$$

where B is the number of assets selected in the portfolio.

Hamiltonian: $z_i = \frac{1-Z_i}{2} \Rightarrow H = \sum_{ij} w_{ij} Z_i Z_j + \sum_i h_i Z_i$

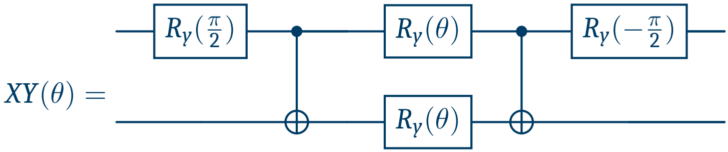
Initial State (Dicke State):

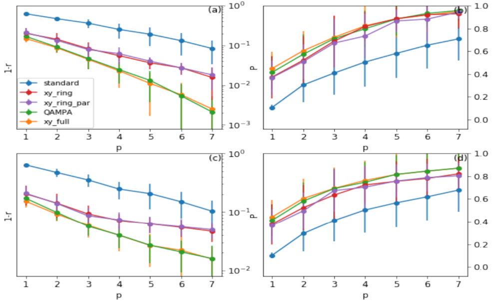
$$|\psi_0^{M_{XY}}\rangle = |D_n^B\rangle = \frac{1}{\sqrt{\binom{n}{B}}} \sum_{\substack{i_1,\dots,i_n=0,1 \\ i_1+\dots+i_n=B}} |i_1 i_2 \dots i_n\rangle$$

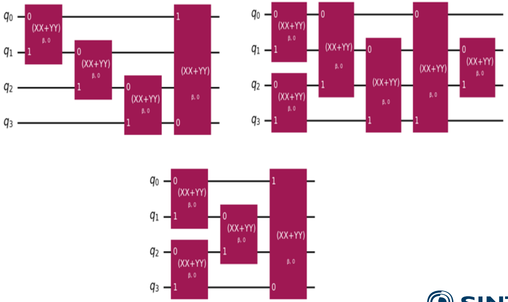
XY Mixer: $XY_{i,j}(\beta) = e^{\,i\beta(\hat{X}_i\hat{X}_j + \hat{Y}_i\hat{Y}_j)}, \quad XX + YY = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} = |10\rangle\langle 01| + |01\rangle\langle 10|$

SINTEF

Example: Quantum Portfolio Optimization with XY Mixers







SINTEF

Thank You!

Ruben Pariente Bassa
Sintef/University of Oslo
`ruben.bassa@sintef.no`