Quantum Gradients in Variational Qauntum Algorithms

Tong Shen, Zeru Zhu, Huifang Xiang, Guangying Wang

December 4, 2025

Stony Brook University

Outline

1 Introduction of Variational Quantum Algorithms

2 Simple Parameter-Shift Rule

3 General Parameter-Shift Rules

4 The Application of Generalized Parameter-Shift Rules
- Brief Introduction of QAOA
- Application

5 The Natural Gradient

Stony Brook University

Outline

1 Introduction of Variational Quantum Algorithms

2 Simple Parameter-Shift Rule

3 General Parameter-Shift Rules

4 The Application of Generalized Parameter-Shift Rules
- Brief Introduction of QAOA
- Application

5 The Natural Gradient

The NISQ Era and Noise

Noisy Intermediate-Scale Quantum (NISQ):

1. Limited number of qubits (≤ 1000).
2. Limited coherence time and gate fidelity.
3. Lack of full error correction.

Canonical quantum algorithms developed over the last three decades remain unreachable in practice.

Table 5. Depths and gate counts of HHL circuits for power flow problems at Iteration 1.

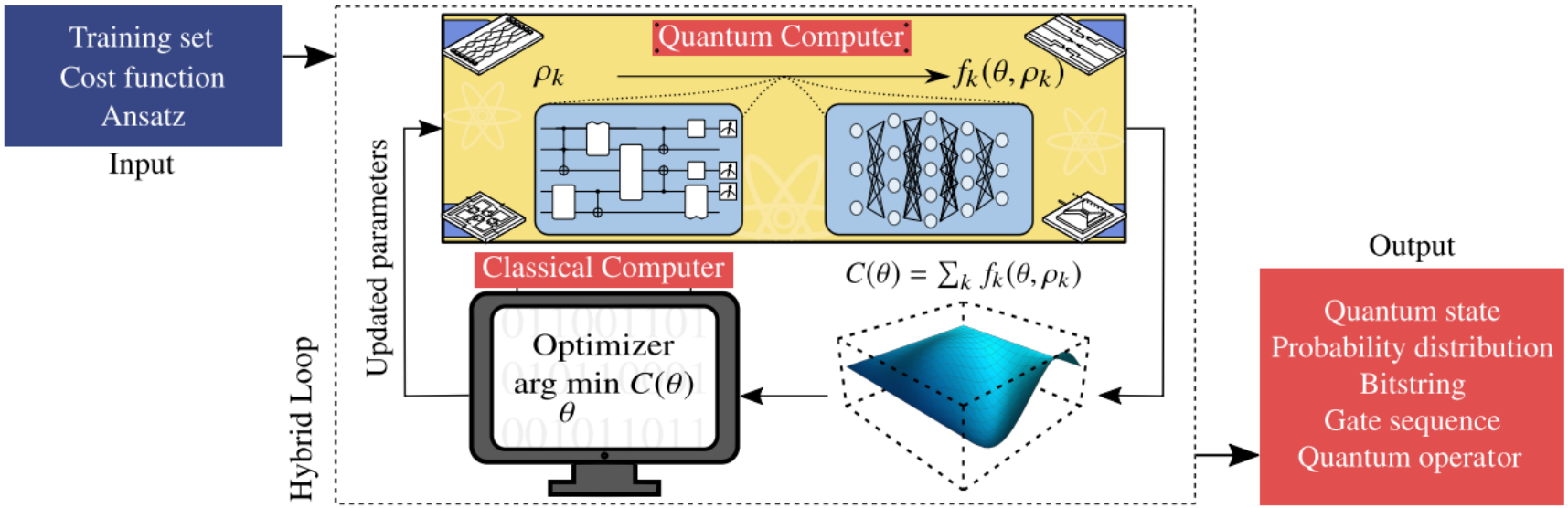
n_d	n_c	Depth	Gates	2-Qubit Gates	Gates After Fusion	Reduction from Fusion
4	4	65,824	86,262	30,651	18,060	79.06%
4	5	135,986	178,180	63,315	37,283	79.08%
4	6	276,308	361,980	128,631	75,717	79.08%
4	7	556,950	729,534	259,247	152,570	79.09%

n_d: qubit for b-register; n_c: qubit for c-register for QPE

The VQA Paradigm

Solution: Variational Quantum Algorithms, which encode the problem into a cost function

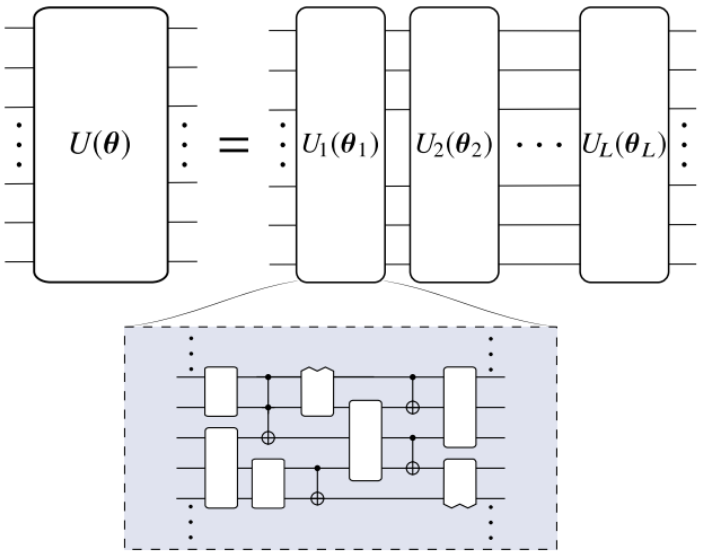
$$C(\boldsymbol{\theta}) = f(\{\rho_k\}, \{O_k\}, U(\boldsymbol{\theta}))$$

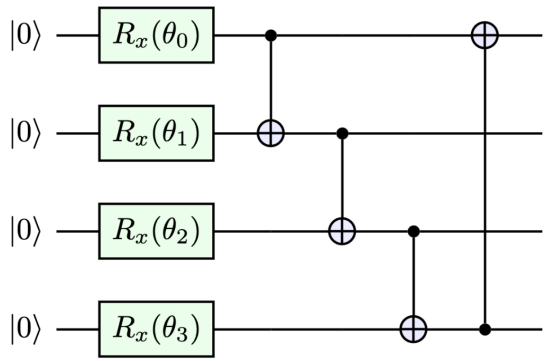


The hybrid of quantum and classical information reduces quantum circuit depth at the expense of additional classical optimization.

Variational Ansatz

The structure of ansatz dictates what parameters $\boldsymbol{\theta}$ are and how they are trained.



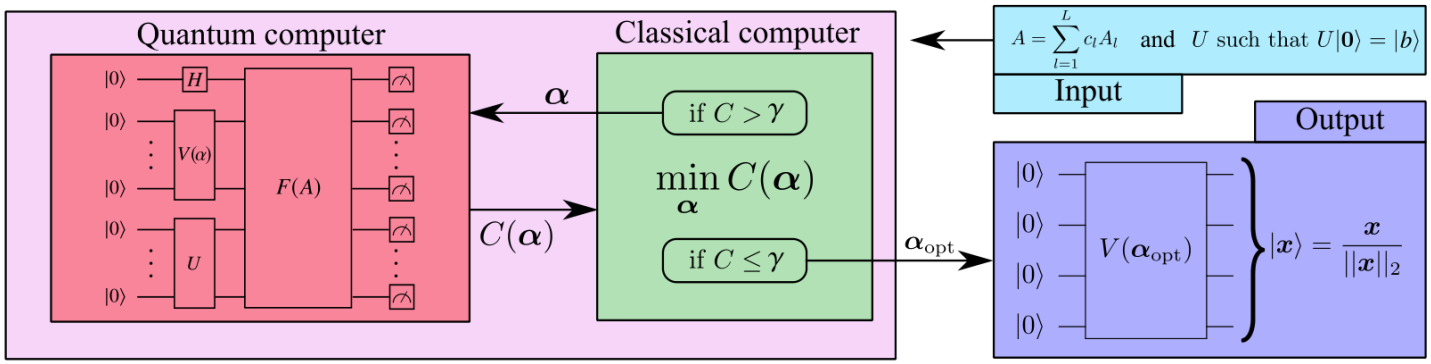


$$U_j(\theta_j) = e^{-i\theta_j G_j}$$

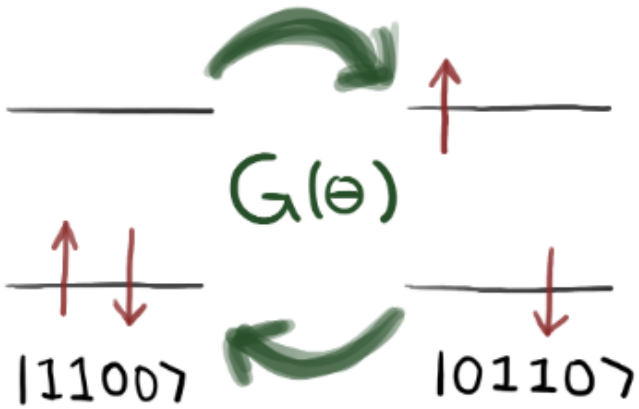
Hardware-efficient Ansatz: designed to reduce circuit depth; versatility

Examples

VQLS: Variational Quantum Linear Solver



VQE: Variational Quantum Eigensolver for molecule ground state.



Choice of optimizer

Why not use Gradient-Free methods (e.g., Nelder-Mead, COBYLA)?

Gradient-Free Methods:

- Pros: Fewer cost function evaluations for each optimization step.
- Cons: Poor scaling with number of parameters (N); Slow convergence.

Gradient-Based Methods:

- Pros: Convergence guarantees; Availability of workhorse algorithms (e.g., SGD) and software tooling developed for machine learning
- Cons: Parameter optimization landscape becomes exponentially flat and featureless as the problem size increases (Barren plateau).

Choice of calculating gradient

1. **Unsuitable for NISQ Hardware**

   Automatic Differentiation (Backpropagation)
   - **Method:** Compute through accumulation of intermediate derivatives, following the chain rule.
   - **Constraint:** The intermediate quantum states cannot be measured without affecting the overall computation.

   Finite Difference
   - **Method:** $\left(\frac{f(\theta+\epsilon)-f(\theta)}{\epsilon}\right)$
   - **Constraint:** Large noise makes it unfeasible to approximate the gradient.

2. **Reliable Methods for NISQ hardware**

   Parameter-Shift Rule
   - **Method:** Unbiased estimation (in some case exact) of the derivative by evaluation the almost same architecture twice.
   - **Best Use Case:** Small-to-medium circuits where **exact precision** is required.

   SPSA (Simultaneous Perturbation Stochastic Approxiamtion.)
   - **Method:** Approximates gradient by perturbing *all* components of the parameter vector simultaneously.
   - **Best Use Case:** Large-scale circuits (100+ params). Noise-resilient.

Outline

1 Introduction of Variational Quantum Algorithms

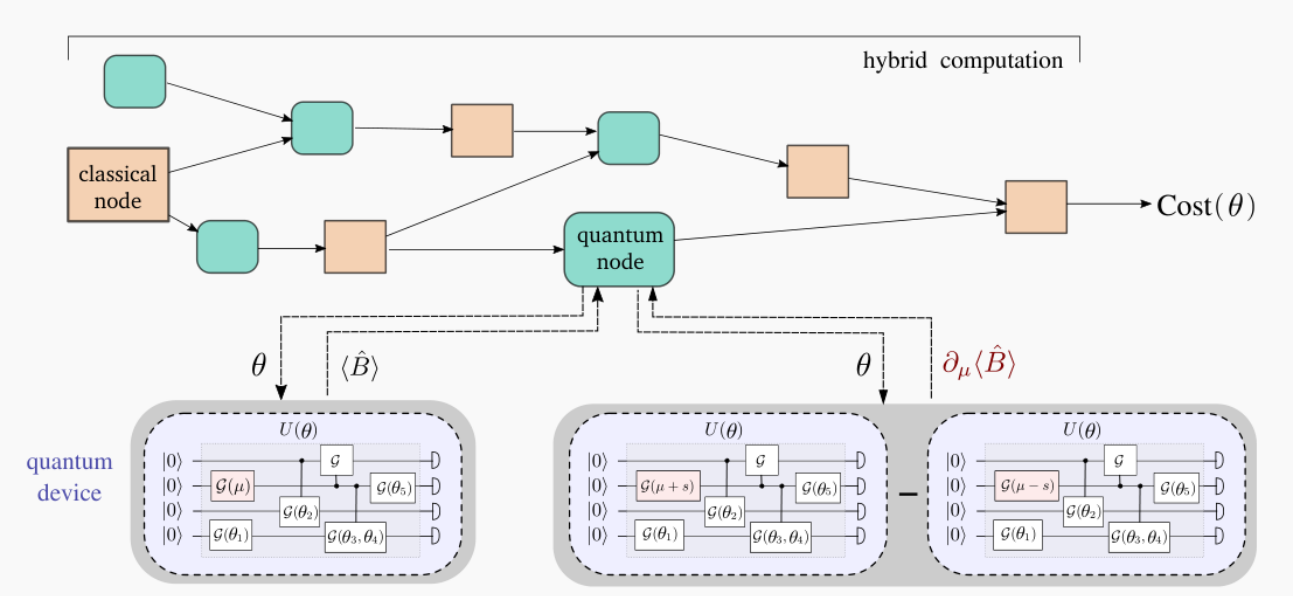
2 Simple Parameter-Shift Rule

3 General Parameter-Shift Rules

4 The Application of Generalized Parameter-Shift Rules
- Brief Introduction of QAOA
- Application

5 The Natural Gradient

Parameter-Shift Rule



$$E(x) := \langle \hat{B} \rangle = \langle \psi | U^\dagger(x) \hat{B} U(x) | \psi \rangle.$$

Parameter-shift rules

$$s = \frac{\pi}{4\omega}, \quad \partial_x E = \omega(E(x+s) - E(x-s)).$$

Derivation

Setup. Consider one parameter gate $U(x)$, the partial derivative becomes

$$\partial_x E = \partial_x \langle\psi|U^\dagger B U|\psi\rangle = \langle\psi|U^\dagger B(\partial_x U)|\psi\rangle + \text{h.c.} \tag{1}$$

$$= \frac{1}{2}\left(\langle\psi|(U+\partial_x U)^\dagger B(U+\partial_x U)|\psi\rangle - \langle\psi|(U-\partial_x U)^\dagger B(U-\partial_x U)|\psi\rangle\right). \tag{1}$$

The gate $U(x) = e^{-ixG}$ generated by Hermitian G with two distinct eigenvalue $\pm\omega$ The derivative is:

$$\partial_x U = -iGe^{-ixG} = -iGU(x)$$

Substitute into (1), we get

$$\partial_x f = \langle\psi'|B(-iG)|\psi'\rangle + \text{h.c.} \tag{2}$$

$$= \frac{\omega}{2}\left(\langle\psi'|(1-i\omega^{-1}G)^\dagger B(1-i\omega^{-1}G)|\psi'\rangle - \langle\psi'|(1+i\omega^{-1}G)^\dagger B(1+i\omega^{-1}G)|\psi'\rangle\right), \tag{2}$$

with $\psi' = U\psi$.

Derivation

Assume G has two eigenvalues $\pm\omega$. Then $G^2 = \omega^2 I$ and the exact Euler expansion holds:

$$U(x) = \exp(-ixG) = \sum_{k=0}^{\infty} \frac{(-ix)^k G^k}{k!} \tag{3}$$

$$= \sum_{k=0}^{\infty} \frac{(-ix)^{2k} G^{2k}}{(2k)!} + \sum_{k=0}^{\infty} \frac{(-ix)^{2k+1} G^{2k+1}}{(2k+1)!} \tag{4}$$

$$= I \sum_{k=0}^{\infty} \frac{(-1)^k (\omega x)^{2k}}{(2k)!} - i\omega^{-1} G \sum_{k=0}^{\infty} \frac{(-1)^k (\omega x)^{2k+1}}{(2k+1)!} \tag{5}$$

$$= I\cos(\omega x) - i\omega^{-1} G \sin(\omega x). \tag{6}$$

Then we have $U(\frac{\pi}{4\omega}) = \frac{1}{\sqrt{2}}(I - i\omega^{-1}G)$.

Substitute into (2), finally we get:

$$\partial_x E = \omega \left(\langle\psi| U^\dagger(x+s) B U(x+s) |\psi\rangle - \langle\psi| U^\dagger(x-s) B U(x-s) |\psi\rangle \right) \tag{7}$$

$$= \omega(E(x+s) - E(x-s)). \tag{8}$$

For Pauli operators ($G = \frac{1}{2}P$, eigenvalues $\pm\frac{1}{2}$), the shift reduces to $s = \frac{\pi}{2}$.

Outline

1 Introduction of Variational Quantum Algorithms

2 Simple Parameter-Shift Rule

3 General Parameter-Shift Rules

4 The Application of Generalized Parameter-Shift Rules
- Brief Introduction of QAOA
- Application

5 The Natural Gradient

General Parameter-Shift Rules

- **From Generators to Fourier Series**
  - Setup & notation
  - Eigen-basis decomposition
  - Finite Fourier series structure
- **Equidistant Frequency**
  - Reconstructing the Fourier coefficients
  - Equidistant parameter-shift rules
- **Arbitrary Spectra**
  - Non-uniform DFT
  - General parameter-shift rules
- **Generalized Gates**
  - Stochastic Parameter-Shift Rule
- **Second-Order Derivatives**
  - Diagonal shift rule for the Hessian
  - Metric tensor

Setup & Notation

We consider a variational circuit with a **single tunable parameter** x:

$$U(x) = e^{ixG}, \qquad G = G^\dagger.$$

The cost function is the expectation value

$$E(x) = \langle\psi|U^\dagger(x)\, B\, U(x)|\psi\rangle.$$

Diagonalize the generator:

$$G = \sum_j \omega_j jj, \qquad \Omega_\ell \in \{\omega_k - \omega_j\}.$$

All dependence on x arises through these **eigenvalue gaps** Ω_ℓ.

Relation to the Standard $f(\theta)$ Notation

It is common to express the cost function as $f(\theta)$ and evaluate the gradient at an arbitrary point θ. Define the shifted function

$$\tilde{E}(x) := E(\theta + x).$$

Then

$$\tilde{E}'(0) = \frac{d}{dx} E(\theta + x)\Big|_{x=0} = \frac{d}{d\theta} f(\theta).$$

The evaluations map as

$$E(x_1) = f(\theta + x_1), \qquad E(-x_1) = f(\theta - x_1).$$

Thus the parameter-shift rule

$$\partial_\theta f(\theta) = r\Big[f(\theta + x_1) - f(\theta - x_1)\Big]$$

corresponds exactly to

$$E'(0) = r\Big[E(x_1) - E(-x_1)\Big].$$

Eigen-basis Decomposition

Expand the state and observable in the eigenbasis of G:

$$|\psi\rangle = \sum_j \psi_j |j\rangle, \qquad B = \sum_{j,k} b_{jk} |j\rangle\langle k|.$$

Using

$$U(x) = \sum_j e^{i\omega_j x} |j\rangle\langle j|,$$

the cost function becomes

$$E(x) = \sum_{j,k} \overline{\psi_j}\, b_{jk}\, \psi_k\, e^{i(\omega_k - \omega_j)x}.$$

Thus $E(x)$ is a finite sum of complex exponentials with frequencies

$$\Omega_\ell = \omega_k - \omega_j.$$

Finite Fourier Series Structure

From the eigen-basis expansion,

$$E(x) = \sum_{j,k} \overline{\psi_j}\, b_{jk}\, \psi_k\, e^{i(\omega_k - \omega_j)x}.$$

Let Ω_ℓ denote the **distinct nonzero eigenvalue gaps**

$$\Omega_\ell \in \{\omega_k - \omega_j\}, \qquad \ell = 1, \ldots, R,$$

where R is the number of distinct frequencies.
Then $E(x)$ can be written as the finite Fourier series

$$E(x) = a_0 + \sum_{\ell=1}^{R} \Big(a_\ell \cos(\Omega_\ell x) + b_\ell \sin(\Omega_\ell x) \Big).$$

Thus $E(x)$ is a **finite** trigonometric polynomial whose structure is fully determined by the spectrum of G.

Reconstructing the Fourier Coefficients

In the equidistant case we can rescale x so that

$$E(x) = a_0 + \sum_{\ell=1}^{R} \big(a_\ell \cos(\ell x) + b_\ell \sin(\ell x)\big),$$

with $2R+1$ real coefficients

$$a_0,\ a_1, \ldots, a_R,\ b_1, \ldots, b_R.$$

Evaluate $E(x)$ at $2R+1$ distinct points x_μ:

$$E(x_\mu) = a_0 + \sum_{\ell=1}^{R} \big(a_\ell \cos(\ell x_\mu) + b_\ell \sin(\ell x_\mu)\big), \qquad \mu = 0, \ldots, 2R.$$

This gives a linear system for the unknown coefficients. Choosing equally spaced points

$$x_\mu = \frac{2\pi\mu}{2R+1}$$

turns this into a (uniform) discrete Fourier transform (DFT), with an explicit inverse.

Explicit Parameter-Shift Rules (Equidistant)

Equivalently, one can write

$$E(x) = \sum_{\mu=0}^{2R} E(x_\mu)\, K_\mu(x),$$

where $K_\mu(x)$ are Dirichlet kernel functions implementing a trigonometric interpolation.

Choose $2R$ shift points

$$x_\mu = \frac{(2\mu-1)\pi}{2R}, \qquad \mu = 1, \ldots, 2R.$$

Then one can have the follows:

First Derivative of Parameter-Shift (Equidistant)

$$E'(0) = \sum_{\mu=1}^{2R} c_\mu\, E(x_\mu), \qquad c_\mu = \frac{(-1)^{\mu-1}}{4R\, \sin^2\left(\frac{(2\mu-1)\pi}{4R}\right)}.$$

Explicit Parameter-Shift Rules (Equidistant)

Similarly, with points $x_\mu = \frac{\mu\pi}{R}$, one can also have:

Second Derivative of Parameter-Shift (Equidistant)

$$E''(0) = -\frac{2R^2+1}{6}E(0) + \sum_{\mu=1}^{2R-1} d_\mu \, E\Big(\frac{\mu\pi}{R}\Big), \qquad d_\mu = \frac{(-1)^{\mu-1}}{2\,\sin^2\big(\frac{\mu\pi}{2R}\big)}.$$

For $R = 1$ these formulas reduce to the familiar two-point parameter-shift rules.

Arbitrary Spectra

When the eigenvalue gaps

$$\Omega_\ell \in \{\omega_k - \omega_j\}$$

are not equally spaced, the Fourier series

$$E(x) = a_0 + \sum_{\ell=1}^{R} \left(a_\ell \cos(\Omega_\ell x) + b_\ell \sin(\Omega_\ell x)\right)$$

cannot be reconstructed using a uniform DFT.

Sampling at points x_μ gives

$$E(x_\mu) = a_0 + \sum_{\ell=1}^{R} \left(a_\ell \cos(\Omega_\ell x_\mu) + b_\ell \sin(\Omega_\ell x_\mu)\right),$$

but the corresponding linear system has **no closed-form inverse** for general Ω_ℓ.

Arbitrary Spectra

Nevertheless, the derivative still admits a parameter-shift form:

General Parameter-Shift (Arbitrary Spectra)

$$E'(0) = \sum_{\mu=1}^{R} y_\mu^{(1)} \big(E(x_\mu) - E(-x_\mu)\big),$$

$$E''(0) = y_0^{(2)} E(0) + \sum_{\mu=1}^{R} y_\mu^{(2)} \big(E(x_\mu) + E(-x_\mu)\big),$$

with coefficients $y_\mu^{(1)}, y_\mu^{(2)}$ obtained by solving a finite linear system.

Stochastic Parameter-Shift Rule

For a generalized gate

$$U_F(x) = e^{i(xG+F)},$$

the dependence on x is no longer of the form e^{ixG}, so closed-form shift rules do not apply directly. F can be any generator, even parameterized by other variables.

Using a splitting identity for $U_F(x_0)$, one obtains

$$E'(x_0) = \int_0^1 \sum_{\mu=1}^{R} y_\mu \Big[E_{+\mu}(x_0, t) - E_{-\mu}(x_0, t) \Big] \, dt, \tag{9}$$

where y_μ are the same coefficients as in the general shift rule for e^{ixG}.

Here $E_{\pm\mu}(x_0, t)$ are expectation values of circuits where

$$U_F(x_0) = U_F(tx_0)\ U(\pm x_\mu)\ U_F((1-t)x_0),$$

and $U(\pm x_\mu) = e^{\pm i x_\mu G}$.

In practice, the integral is estimated by sampling $t \in [0, 1]$, yielding a **stochastic** parameter-shift rule that applies to arbitrary gates $U_F(x)$.

Comparison of Parameter-Shift Settings

	Simple Gate $U(x) = e^{ixG}$	**Generalized Gate** $U_F(x) = e^{i(xG+F)}$
Equidistant	**Closed-form** Uniform DFT (Dirichlet kernel)	**Equidistant + Stochastic** Coefficients in simple gate Estimate integral stochastically
Arbitrary	**No Closed-form** Non-uniform DFT (Numerically computed coefficients y_μ)	**General + Stochastic** Coefficients in simple gate Estimate integral stochastically

Multivariate case: All results extend directly—differentiate with respect to one parameter while holding all others fixed and apply the corresponding univariate

Diagonal shift rule for the Hessian

Setup. We have a multivariate cost $E(\mathbf{x}), \mathbf{x} = (x_1, \ldots, x_n)$, and want Hessian entries at a reference point $\mathbf{x}_0$:

$$H_{km} = \partial_k \partial_m E(\mathbf{x}_0).$$

Univariate restriction. For an off-diagonal entry,

$$E^{(km)}(x) := E(\mathbf{x}_0 + x\mathbf{v}_{k,m}), \qquad \mathbf{v}_{k,m} := \mathbf{v}_k + \mathbf{v}_m.$$

Key observation. If the single-parameter dependence of E has *equidistant Fourier frequencies*, then $E^{(kk)}$ and $E^{(km)}$ are finite Fourier series with integer frequencies. Hence we can apply the univariate second-order parameter-shift rule to obtain

$$\frac{\mathrm{d}^2}{\mathrm{d}x^2} E^{(km)}(x)\Big|_{x=0} = H_{kk} + H_{mm} + 2H_{km}.$$

Diagonal shift rule. Rearranging gives

$$H_{km} = \frac{1}{2}\Big(E^{(km)\prime\prime}(0) - H_{kk} - H_{mm}\Big),$$

so H_{km} can be computed from univariate second derivatives plus the diagonal entries.

Natural Gradient and the Fubini–Study Metric

Motivation. Parameter space corresponds to a manifold of quantum states (the projective Hilbert space). Ordinary Euclidean gradient descent does not respect this geometry. The **quantum natural gradient** rescales updates by the Riemannian metric of the state manifold:

$$\nabla_n E(x) := \mathcal{F}^{-1}(x)\,\nabla E(x).$$

Fubini–Study metric. The metric tensor for a parametrized state $|\psi(x)\rangle = V(x)|0\rangle$ is given by

$$\mathcal{F}_{km}(x_0) = \Re\{\langle\partial_k\psi|\partial_m\psi\rangle\} - \langle\partial_k\psi|\psi\rangle\langle\psi|\partial_m\psi\rangle. \tag{44}$$

Key insight: the metric is a Hessian. Define the overlap cost

$$f(x) := -\tfrac{1}{2}|\langle\psi(x)|\psi(x_0)\rangle|^2.$$

Then

$$\mathcal{F}_{km}(x_0) = \partial_k\partial_m f(x_0). \tag{45}$$

Consequence. Since $\mathcal{F}$ is a Hessian, *the same parameter-shift rules from Sec. 4 apply directly* to compute each $\mathcal{F}_{km}$.

Computing the Metric Tensor Using Parameter Shifts

Goal: natural gradient requires the metric

$$\nabla_n E(x) = \mathcal{F}^{-1}(x)\, \nabla E(x). \tag{48}$$

Thus we must compute all metric entries $\mathcal{F}_{km}$.

Observation. Because

$$\mathcal{F}_{km}(x_0) = \partial_k \partial_m f(x_0),$$

we can compute the metric by evaluating the Hessian of the scalar function

$$f(x) = -\tfrac{1}{2} |\langle \psi(x) | \psi(x_0) \rangle|^2.$$

Diagonal shift rule. For equidistant spectra, the univariate reconstruction of f along shift directions

$$E^{(k)}(x), \quad E^{(m)}(x), \quad E^{(km)}(x)$$

allows us to use the **same diagonal-shift formula** as for the Hessian of a cost function:

$$\mathcal{F}_{km} = \tfrac{1}{2}\Big(f^{(km)\prime\prime}(0) - f^{(k)\prime\prime}(0) - f^{(m)\prime\prime}(0) \Big).$$

Takeaway.

- Natural gradient requires the Fubini–Study metric.
- The metric is a Hessian, so its entries are obtained via the *same* parameter-shift rules as Sec. 4.
- This makes natural gradient computation compatible with standard VQA gradient/Hessian techniques.

Outline

1 Introduction of Variational Quantum Algorithms

2 Simple Parameter-Shift Rule

3 General Parameter-Shift Rules

4 The Application of Generalized Parameter-Shift Rules
- Brief Introduction of QAOA
- Application

5 The Natural Gradient

Generalized Parameter-Shift Rule for QAOA

The quantum approximate optimization algorithm (QAOA) is a general technique that can be used to find approximate solutions to combinatorial optimization problems, in particular problems that can be cast as searching for an optimal bitstring.
The generalized parameter-shift rule (Eq. 24) says that for a function E with R frequencies, we only need $2R$ evaluations of E to get its derivative.

$$E'(0) = \sum_{\mu=1}^{2R} E\left(\frac{2\mu-1}{2R}\pi\right) \frac{(-1)^{\mu-1}}{4R\sin^2\left(\frac{2\mu-1}{4R}\pi\right)}, \tag{24}$$

A single-parameter layer in QAOA (such as U_M or U_P) can be seen as a single-parameter gate with many frequencies. Therefore, QAOA is a typical application of the generalized parameter-shift formula.

Connection Between Quantum Circuits and Hamiltonians

In physics, quantum gates are made by letting a system change over time under a special Hamiltonian.
This change is described by the *time evolution operator*. It is a unitary operator and is written as:

$$U(H,t) = e^{-iHt/\hbar}$$

The time evolution operator depends on a Hamiltonian and a number that means time. Any unitary operator can be written in this form, where the Hamiltonian is a Hermitian operator.
So, time evolution gives us a way to describe quantum circuits using Hamiltonians.

Hamiltonians With Many Non-Commuting Terms

Sometimes, a quantum circuit comes from a Hamiltonian with many non-commuting parts:

$$H = H_1 + H_2 + \cdots + H_N$$

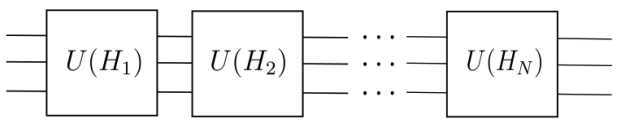
It is very hard to compute the exact exponential of this Hamiltonian.
Instead, we use the Trotter-Suzuki formula:

$$e^{A+B} \approx (e^{A/n} e^{B/n})^n$$

This gives an approximate time-evolution unitary:

$$U(H, t, n) = \prod_{j=1}^{n} \prod_{k} e^{-iH_k t/n}, \text{ where } H = \sum_k H_k.$$

$$H = H_1 + H_2 + \cdots + H_N$$



QAOA steps:

- Define a cost Hamiltonian H_C such that its ground state encodes the solution to the optimization problem.
- Define a mixer Hamiltonian H_M.
- Construct the circuits and We call these the cost and mixer layers $e^{-i\gamma H_C} e^{-i\alpha H_M}$, respectively.
- Choose a number of layers $n \geq 1$, repeat application of the cost and mixer layers and build the circuit:

$$U(\boldsymbol{\gamma}, \boldsymbol{\alpha}) = e^{-i\alpha_n H_M} e^{-i\gamma_n H_C} \ldots e^{-i\alpha_1 H_M} e^{-i\gamma_1 H_C}$$

- Prepare an initial state $U(\boldsymbol{\gamma}, \boldsymbol{\alpha})$, apply the circuit, and use classical optimization to find good parameters.
- After optimization, measurements of the output state reveal approximate solutions to the optimization problem.

Problem Setting

In general, given a problem Hamiltonian H_P that encodes the solution to the problem of interest onto N qubits, QAOA applies two types of layers aternatively to the initial state $|+\rangle^{\otimes N}$:

$$V_{\mathrm{QAOA}}(x) = \prod_{j=p}^{1} U_M(x_{2j})\, U_P(x_{2j-1})$$

Here, p is the block number, which decides the depth of the circuit.

Mixing layer:

$$U_M(x) = e^{-ixH_M}$$

Problem layer:

$$U_P(x) = e^{-ixH_P}$$

Objective Function in QAOA

The parameters x can then be optimized to try to minimize the objective function

$$E(x) = \langle +|^{\otimes N} V_{\mathrm{QAOA}}^{\dagger}(x) H_P V_{\mathrm{QAOA}}(x) |+\rangle^{\otimes N}.$$

Explanation:
We first define the parameterized state: $|\psi(x)\rangle = V_{\mathrm{QAOA}}(x)|+\rangle^{\otimes N}$,
Then we define the cost function as the expectation value of the problem Hamiltonian H_P:

$$E(x) = \langle \psi(x)|H_P|\psi(x)\rangle.$$

Substituting the expression of $|\psi(x)\rangle$, we obtain

$$E(x) = \langle +|^{\otimes N} V_{\mathrm{QAOA}}^{\dagger}(x) H_P V_{\mathrm{QAOA}}(x) |+\rangle^{\otimes N}.$$

MaxCut Example

Here, we focus on the problem layer U_P, using MaxCut as an example. For an unweighted graph $G = (V, E)$, where $|V|$ denotes the number of vertices and $|E|$ denotes the number of edges, the corresponding MaxCut Hamiltonian is given by

$$H_P = \sum_{(a,b)\in E} \frac{1}{2}(1 - Z_a Z_b).$$

- $Z_a Z_b = +1$: nodes on same side $\rightarrow$ no cut
- $Z_a Z_b = -1$: nodes on different sides $\rightarrow$ cut

The largest eigenvalue of this Hamiltonian is the best MaxCut value. So QAOA tries to get close to this maximum value.

Frequencies of Mixing Layer and Problem Layer

Eigenvalues of $H_M = \sum_k X_k$:

$$-N, -N+2, \ldots, N$$

Set of eigenvalue gaps:

$$\{2, 4, \ldots, 2N\}$$

Therefore, the number of frequencies in the mixing layer is

$$R_M = N$$

Frequencies of the problem layer H_P lie in the set

$$[M] = \{1, \ldots, M\}$$

The largest frequency is the largest eigenvalue: $\lambda = \mathrm{MaxCut}(G)$.
Therefore,

$$R_P \leq \lambda \leq M$$

Using an Upper Bound on the Frequencies

In general, we cannot get the exact set of frequencies $\{\Omega_\ell\}$ or the true number of frequencies R.
Instead, we use an upper bound $\varphi(G)$ such that

$$\varphi(G) \geq \mathrm{MaxCut}(G) = \lambda.$$

This upper bound tells us the largest possible frequency, and helps us choose a safe number of frequencies R for the parameter-shift rules.

Ways to get the upper bound:

- Analytic upper bound from the graph Laplacian.
- Use a semidefinite program (SDP) that relaxes the MaxCut problem.

Comparison

To illustrate the above strategy, **Table 3** summarizes the number of evaluations required to compute the gradient or both the gradient and the Hessian for QAOA circuits solving the MAXCUT problem, across several different classes of graphs.

Graph type	Decomposition-based		Gen. shift rule		
	∇E	$\nabla E \& \nabla^{\otimes 2} E$	Bound φ	∇E	$\nabla E \& \nabla^{\otimes 2} E$
General	$(M+N)n$	$\mathcal{O}(n^2(M+N)^2)$	φ	$n(\varphi+N)$	$\mathcal{O}(n^2(\varphi+N))$
Complete	$\frac{1}{2}n(N^2+N)$	$\mathcal{O}(n^2N^4)$	$\left\lfloor \frac{N^2}{4} \right\rfloor$	$n\left(\left\lfloor \frac{N^2}{4} \right\rfloor + N\right)$	$\mathcal{O}(n^2N^2)$
$2k$-regular	$(k+1)nN$	$\mathcal{O}(k^2n^2N^2)$	kN	$\frac{k+2}{2}nN$	$\mathcal{O}(kn^2N)$
$(2k+1)$-regular	$\frac{2k+3}{2}nN$	$\mathcal{O}(k^2n^2N^2)$	$\frac{2k+1}{2}N$	$\frac{2k+3}{2}nN$	$\mathcal{O}(kn^2N)$

Table 3: Evaluation numbers for the gradient, or both the gradient and the Hessian, for QAOA circuits for MaxCut on several types of graphs. Each graph has N vertices and a graph type-specific number M of edges, and the (even) number of parameters is denoted as n. For K-regular graphs, we know $M = \min\{(N^2-N)/2, KN/2\}$, and the latter value is used in the displayed evaluation costs; if the former value forms the minimum, the graph is in fact complete. The left column is based on decomposing the circuit, applying the conventional two-term parameter-shift rule per elementary gate and iterating it for the Hessian. The right column employs the generalized parameter-shift rule Eq. (24) combined with an upper bound φ for the largest eigenvalue λ of the problem Hamiltonian, as well as the reduced number of evaluations for Hessian off-diagonal terms from Sec. 4.1. The bound φ for complete graphs can be found in Ref. [67].

Overall, across all considered graph families, the generalized parameter-shift rule offers **significant efficiency improvements** for QAOA: gradient evaluations become less costly, Hessian evaluations are substantially reduced, and the overall computational efficiency is greatly enhanced.

Outline

1 Introduction of Variational Quantum Algorithms

2 Simple Parameter-Shift Rule

3 General Parameter-Shift Rules

4 The Application of Generalized Parameter-Shift Rules
- Brief Introduction of QAOA
- Application

5 The Natural Gradient

Gradient Descent

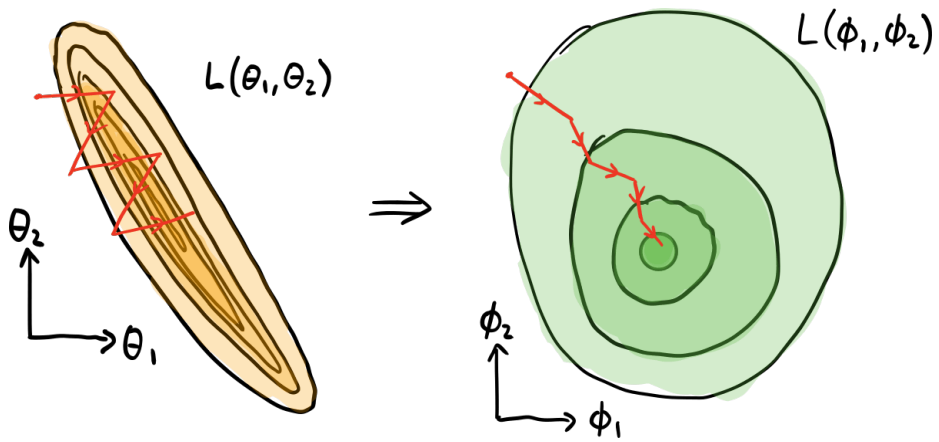
Iterative update rule of vanilla gradient descent:

$$\theta_{t+1} := \theta_t - \eta \nabla \mathcal{L}(\theta_t), = \arg\min_{\theta \in \mathbb{R}^d} \left[\langle \theta - \theta_t, \nabla \mathcal{L}(\theta_t) \rangle + \frac{1}{2\eta} \|\theta - \theta_t\|_2^2 \right] \tag{10}$$

Moves in the steepest descent direction with respect to the l_2 geometry.

- Small updates in parameters may cause large or negligible changes in the quantum state.
- Euclidean gradients assume a flat parameter space, but quantum states live on a curved manifold.
- Vanilla gradients may move in directions that minimally change the state, leading to slow convergence.
- Ignoring geometry often results in tiny gradients and stalled training.

Gradient Descent



Each optimization step is strongly connected to a Euclidean geometry on the parameter space. The parametrization is not unique, and different parametrizations can distort distances within the optimization landscape.

Natural Gradient

1. Ordinary gradient descent depends on the chosen parametrization.
2. Natural gradient[1] performs gradient descent in distribution space using an invariant metric.
3. Yields optimally scaled, reparametrization-invariant update directions.

$$\theta_{t+1} = \theta_t - \eta F^{-1} \nabla \mathcal{L}(\theta) \tag{11}$$

where F is the Fisher information matrix. The Fisher information matrix acts as a metric tensor, transforming the steepest descent in the Euclidean parameter space to the steepest descent in the distribution space.

[1] Shun-ichi Amari; Natural Gradient Works Efficiently in Learning. Neural Comput 1998; 10 (2): 251–276. doi: https://doi.org/10.1162/089976698300017746

Natural Gradient

1. **Respects quantum state geometry:** Natural gradient follows the Fubini–Study metric, the intrinsic geometry of the projective Hilbert space where variational states live.
2. **Provides the true steepest-descent direction:** Ordinary gradients assume a Euclidean geometry, while natural gradient yields the steepest descent direction with respect to the quantum metric.
3. **Leads to more efficient and stable optimization:** As demonstrated in the paper, natural gradient achieves faster convergence and smoother training trajectories compared to standard gradient-based methods.

Fisher information

Fisher–Rao metric. The Fisher–Rao metric defines a natural Riemannian geometry on the probability simplex. The distance between two probability distributions p and q is given by

$$d(p, q) = \arccos\big(\langle \sqrt{p}, \sqrt{q} \rangle\big). \tag{12}$$

Line element

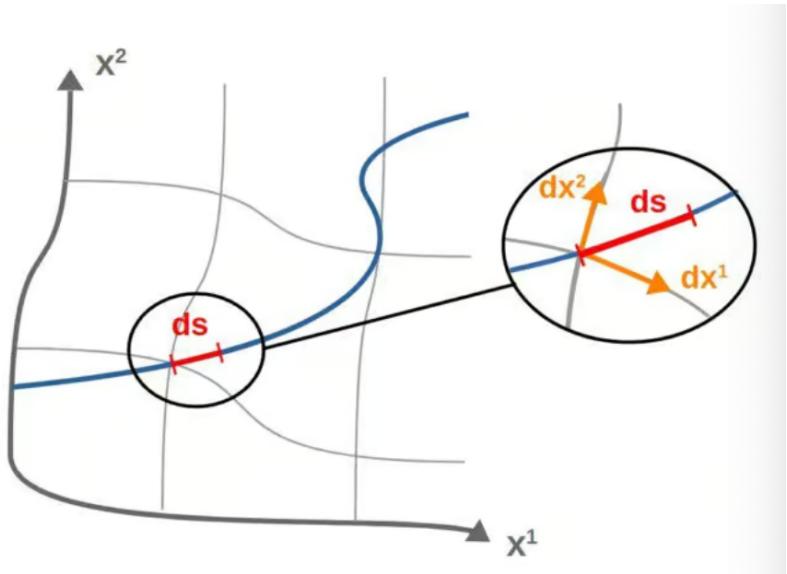


Figure: How to Measure Distance Between Two Quantum States

$$ds^2 = g_{ij}\, dx^i dx^j \tag{13}$$

ds is actual length, g_{ij} the metric components act as conversion factors" and $dx^i dx^j$ is the coordinate displacement.

Line element

For a parametric family p_θ, the induced Fisher–Rao metric generates the following infinitesimal line element:

$$d^2(p_\theta, p_{\theta+d\theta}) = \frac{1}{4}\sum_{i=1}^{d}\sum_{j=1}^{d} I_{ij}(\theta)\, d\theta_i\, d\theta_j. \tag{14}$$

The Fisher information matrix is given by:

$$I_{ij}(\theta) = \sum_{x\in\{1,\ldots,N\}} p_\theta(x)\, \frac{\partial \log p_\theta(x)}{\partial \theta_i}\, \frac{\partial \log p_\theta(x)}{\partial \theta_j}. \tag{15}$$

Fubini–Study Metric for Parameterized Quantum States

For a parametric family of states ψ_θ, the Fubini–Study distance between two states is

$$d(P_\psi, P_\phi) = \arccos(|\langle\psi, \phi\rangle|). \tag{16}$$

The corresponding infinitesimal line element on the state manifold is

$$d^2(P_{\psi_\theta}, P_{\psi_{\theta+d\theta}}) = \sum_{(i,j)\in[d]^2} g_{ij}(\theta)\, d\theta^i d\theta^j, \tag{17}$$

where $g_{ij}(\theta) = \mathrm{Re}\, G_{ij}(\theta)$ s the Fubini-Study metric tensor.

This metric determines the natural geometry of variational quantum states and forms the basis for the **quantum natural gradient**.

Quantum Geometric Tensor and Natural Gradient

Quantum Geometric Tensor:

$$G_{ij}(\theta) = \left\langle \frac{\partial \psi_\theta}{\partial \theta^i}, \frac{\partial \psi_\theta}{\partial \theta^j} \right\rangle - \left\langle \frac{\partial \psi_\theta}{\partial \theta^i}, \psi_\theta \right\rangle \left\langle \psi_\theta, \frac{\partial \psi_\theta}{\partial \theta^j} \right\rangle \tag{18}$$

which rescales gradients by the intrinsic geometry of the quantum state manifold, improving the optimization efficiency in variational quantum algorithms.

Optimization Problem & Natural Gradient Update

Objective minimize the generic VQA loss $\mathcal{L}(\theta) = \frac{1}{2}\langle\psi_\theta|H|\psi_\theta\rangle$
Natural Gradient Formulation: Instead of Euclidean updates, we solve the local quadratic problem:

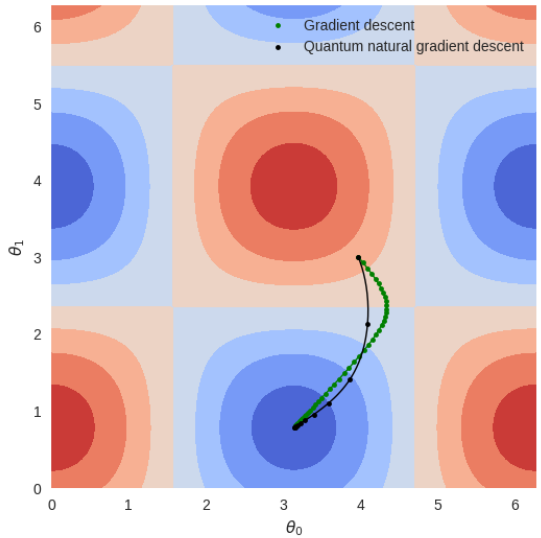
$$\theta_{t+1} = \arg\min_{\theta} \left[\langle \theta - \theta_t, \nabla\mathcal{L}(\theta_t)\rangle + \frac{1}{2\eta}\|\theta - \theta_t\|^2_{g(\theta_t)} \right],$$

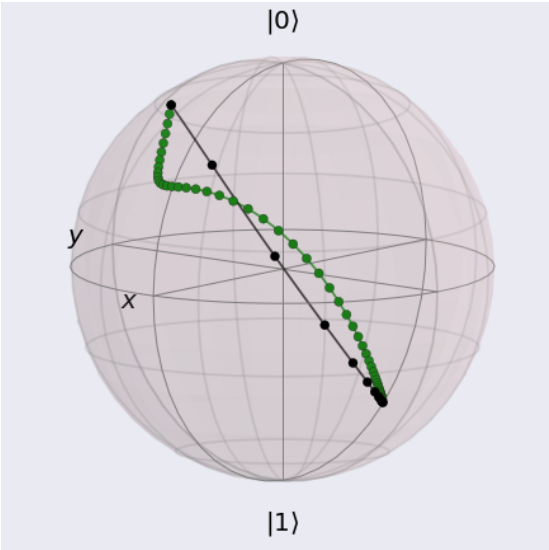
with the metric norm $\|\theta - \theta_t\|^2_{g(\theta_t)} = \langle \theta - \theta_t,\ g(\theta_t)(\theta - \theta_t)\rangle$.
the natural gradient update is

$$\theta_{t+1} = \theta_t - \eta\, g^{+}(\theta_t)\, \nabla\mathcal{L}(\theta_t),$$

where g^+ is the (pseudo-)inverse of the quantum metric.

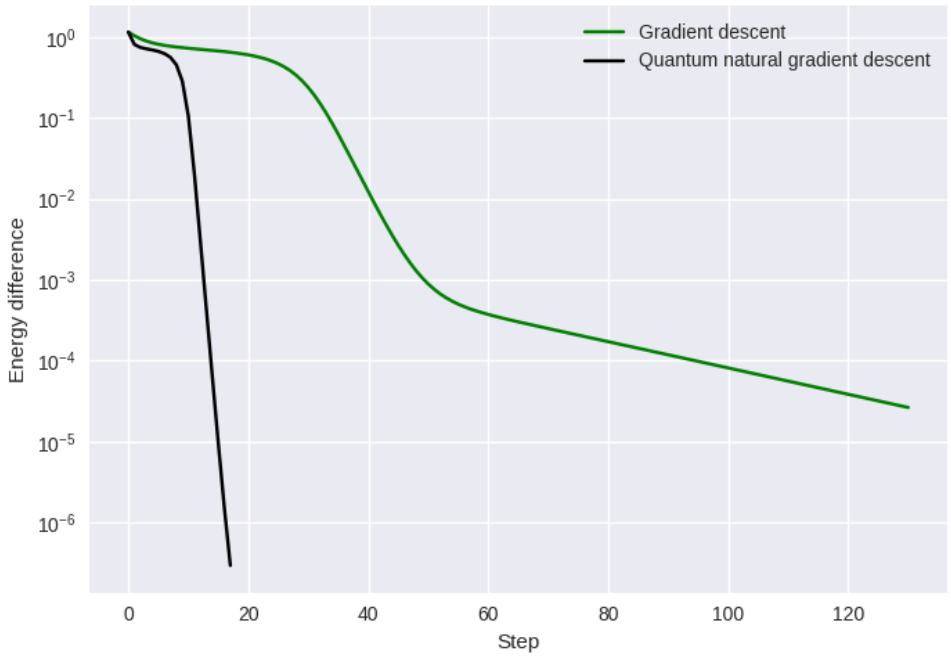
Visualization





The blue regions indicate states with lower energies, and the red regions indicate states with higher energies. We can see that the Natrual Gradient -based Optimizer takes a more direct route to the minimum in larger strides compared to the path taken by the Gradient Descent Optimizer.

Visualization



Quantum natural gradient requires fewer steps to reach an energy lower than that obtained using vanilla gradient descent.

Reference

1. Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. "Evaluating analytic gradients on quantum hardware". Phys. Rev. A 99, 032331 (2019).
2. Cerezo, M. and Arrasmith, Andrew and Babbush, Ryan and Benjamin, Simon C. and Endo, Suguru and Fujii, Keisuke and McClean, Jarrod R. and Mitarai, Kosuke and Yuan, Xiao and Cincio, Lukasz and Coles, Patrick J. "Variational quantum algorithms"(2021). arxiv.org/abs/2012.09265.
3. Wierichs, David and Izaac, Josh and Wang, Cody and Lin, Cedric Yen-Yu. "General parameter-shift rules for quantum gradients"(2022), arxiv.org/abs/2107.12390.
4. Leonardo Banchi and Gavin E. Crooks. Measur ing analytic gradients of general quantum evolu tion with the stochastic parameter shift rule . Quantum 5, 386 (2021).

Questions and Answers

Thank you!
Q & A