Quantum Natural Gradient

James Stokes, Josh Izaac, Nathan Killoran, and Giuseppe Carleo

August 31, 2025

What is a Bit? (比特)

- **Bit** = the smallest unit of classical information.
- Can only be **0** (off, 关) or **1** (on, 开).
- All data in a computer (text, images, videos) is stored as long strings of bits.
- Analogy: a light switch—either off (0) or on (1).

What is a Qubit? (量子比特)

- A **Qubit** can be in state $|0\rangle$, state $|1\rangle$, or a **superposition (叠加态)**:
$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$
- α, β are complex numbers with $|\alpha|^2 + |\beta|^2 = 1$.
- **Measurement**: outcome is 0 with probability $|\alpha|^2$, or 1 with probability $|\beta|^2$.
- Analogy: a spinning coin (在空中旋转的硬币) that is partly heads and partly tails until observed.

Entanglement (量子纠缠)

- **Definition**: A special correlation between qubits that cannot be explained classically.
- Example: two-qubit entangled state

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

- **Measurement (测量)**:
  - If the first qubit is measured as 0, the second must also be 0.
  - If the first is 1, the second is guaranteed to be 1.
- Intuition (直观类比): Like two perfectly synchronized coins (同步硬币) —flip one, the other shows the same, no matter how far apart.
- Importance (重要性): Entanglement is a key resource powering quantum algorithms and quantum communication.

What is Quantum Computing?

- **Quantum state (量子态)**: superposition

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

- **Quantum gates (量子门)**: unitary operations

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \quad X|0\rangle = |1\rangle$$

- **Quantum circuits (量子电路)**: sequence of gates

$$|\psi_{\text{out}}\rangle = U|0\rangle^{\otimes n}$$

- **In short**: Quantum Computing = Superposition + Entanglement + Circuits

Why Quantum Computing Matters?

- **Superposition (叠加)**: explore many possibilities in parallel.
- **Entanglement (纠缠)**: qubits work together with non-classical correlations.
- **Applications**:
  - **Optimization**
  - **Machine learning**
  - **Quantum chemistry**
  - ...
- **Reality**: not all problems are faster; current hardware is Noisy Intermediate-Scale Quantum (**NISQ**).

Variational Quantum Algorithms (VQAs)

- **Ansatz (参数化电路)**: prepare a state $|\psi_\theta\rangle = U(\theta)|0\rangle$ with tunable parameters θ.
- **Cost function (目标函数)**: e.g., energy expectation $L(\theta) = \langle\psi_\theta|H|\psi_\theta\rangle$.
- **Hybrid loop (混合优化循环)**:
  - Quantum device: evaluate $L(\theta)$ and gradients.
  - Classical optimizer: update parameters θ.
- **Why VQAs?** Suitable for today's **NISQ** (噪声中等规模) devices, more noise-resilient than deep circuits.

Optimization Challenges

- **Barren plateaus (贫瘠高原)**: gradients ≈ 0 in wide regions $\Rightarrow$ slow learning.
- **Ill-conditioning (尺度不均)**: some directions too steep/flat; vanilla GD zigzags, unstable steps.
- **Noisy/stochastic estimates (测量噪声)**: finite shots + hardware noise $\Rightarrow$ noisy $L(\theta)$, ∇L.
- **Costly gradients (梯度代价高)**: parameter-shift or finite-diff needs many circuit evaluations.
- **Reparametrization sensitivity (重参数化敏感)**: scaling of θ changes step quality.

Takeaway: We need a geometry-aware method $\Rightarrow$ **Natural Gradient** $\Rightarrow$ **Quantum Natural Gradient (QNG)**.

Natural Gradient (自然梯度)

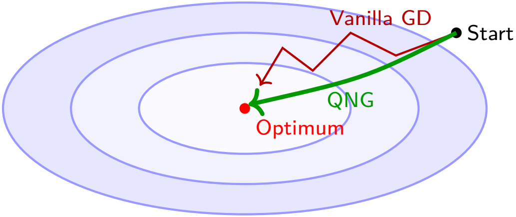
- **Vanilla Gradient Descent**: $\theta_{t+1} = \theta_t - \eta \nabla L(\theta_t)$
  Uses simple Euclidean geometry (ℓ_2 norm).
- **Natural Gradient**: scales the update by the **Fisher Information Matrix (FIM)** $F(\theta)$:

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

- **Key idea**: follow steepest descent direction under the **information geometry**.
- **Benefits**:
  - Invariant to reparametrization (重参数化不敏感).
  - More stable and efficient convergence.
- **Analogy (类比)**: Vanilla GD = straight line on a flat map; Natural Gradient = shortest path on Earth's surface (测地线).

Quantum Natural Gradient (量子自然梯度, QNG)

- **Idea**: Quantum states live on a curved space (Fubini–Study metric).
- **Vanilla GD**: ignores curvature ⇒ inefficient steps.
- **QNG**: geometry-aware steps ⇒ faster, stable.



Quantum Geometric Tensor (QGT)

- **Definition**: For variational state $|\psi(\theta)\rangle$, the Quantum Geometric Tensor is

$$G_{ij} = \langle \partial_i \psi | \partial_j \psi \rangle - \langle \partial_i \psi | \psi \rangle \langle \psi | \partial_j \psi \rangle.$$

- **Metric tensor (度量张量)**: The real part gives the **Fubini–Study metric**

$$g_{ij} = \mathrm{Re}\, G_{ij}.$$

- **Physical meaning (物理意义)**: g_{ij} measures the sensitivity of the quantum state to parameter changes —i.e. how "curved" the parameter space is.
- **Connection to QNG**: QNG update uses this metric tensor to rescale gradients, ensuring geometry-aware optimization.

QNG in Practice: one iteration

- **Objective**: $L(\theta) = \langle\psi_\theta|H|\psi_\theta\rangle$, $|\psi_\theta\rangle = U(\theta)|0\rangle$.
- **Gradient (参数位移)** for Pauli rotations:

$$\partial_i L \;=\; \frac{1}{2}\Big[L(\theta + \tfrac{\pi}{2}e_i) - L(\theta - \tfrac{\pi}{2}e_i)\Big].$$

- **Metric tensor (度量张量)**: $g(\theta) = \mathrm{Re}\, G(\theta)$, where

$$G_{ij} = \langle\partial_i\psi|\partial_j\psi\rangle - \langle\partial_i\psi|\psi\rangle\langle\psi|\partial_j\psi\rangle.$$

  Practical: use **block-diagonal metric tensor** per layer (或只用 **diagonal metric tensor** 近似).
- **Update**: solve a damped system

$$(g(\theta) + \lambda I)\,\Delta\theta = -\eta\,\nabla L(\theta), \qquad \theta \leftarrow \theta + \Delta\theta,$$

  with small $\lambda > 0$ (稳定); optional: clip $\|\Delta\theta\|$.

Estimating the metric tensor

- For layer l with commuting generators $\{K_i\}$ (e.g. Pauli rotations):

$$g_{ij}^{(l)} = \langle K_i K_j \rangle - \langle K_i \rangle \langle K_j \rangle \quad \text{on state } |\psi_l\rangle.$$

- Because $[K_i, K_j] = 0$ in a layer, **one measurement basis per layer** suffices.
- **Diagonal metric tensor**: $g_{ii} = \mathrm{Var}(K_i) = \langle K_i^2 \rangle - \langle K_i \rangle^2$ (最省 shots).
- **Cost tips**: share shots across i, j in the same layer; reuse cached estimates when θ 变化很小。
- **Numerics**: solve $(g + \lambda I)\Delta\theta = -\eta \nabla L$ by Cholesky (block-wise) or CG.

Quantum Natural Gradient (QNG): One Iteration

QNG Iteration (基于度量张量的一步更新)

Input: Ansatz $U(\theta)$, cost $L(\theta) = \langle\psi_\theta|H|\psi_\theta\rangle$, step size η, damping λ
Output: Updated parameters θ^+

Prepare: $|\psi_\theta\rangle = U(\theta)|0\rangle$

1) Metric tensor (Fubini–Study) per layer:

For each layer with commuting generators $\{K_i\}$:

$$g_{ij}^{(l)} = \langle K_i K_j\rangle - \langle K_i\rangle\langle K_j\rangle$$

(block-diag metric tensor) Diagonal option:
$g_{ii} = 1 - \langle K_i\rangle^2$

2) Gradient (parameter-shift):

$$\partial_{\theta_i} L(\theta) = \frac{1}{2}\left[L(\theta_i + \tfrac{\pi}{2}) - L(\theta_i - \tfrac{\pi}{2})\right]$$

3) QNG step (solve linear system):

$$(g + \lambda I)\Delta\theta = -\eta\nabla L(\theta), \quad \theta^+ \leftarrow \theta + \Delta\theta$$

Experiments: Setup

- **Tasks**:
  - Variational Quantum Eigensolver (VQE) for molecular Hamiltonians (e.g. H_2, LiH).
  - Quantum Approximate Optimization Algorithm (QAOA) for MaxCut problems.
- **Ansatz circuits**:
  - Hardware-efficient layered ansatz (含参数化旋转门 + CNOT entanglers).
  - Depth L varied to test scalability.
- **Optimizers compared**:
  - Vanilla Gradient Descent (GD)
  - Adam optimizer
  - Quantum Natural Gradient (QNG): block-diagonal & diagonal approximations
- **Metrics**:
  - Convergence speed (iterations until near-optimal energy).
  - Final energy error vs. ground truth.
  - Shot complexity for estimating gradients / metric tensor.

Experimental results: Figure 1

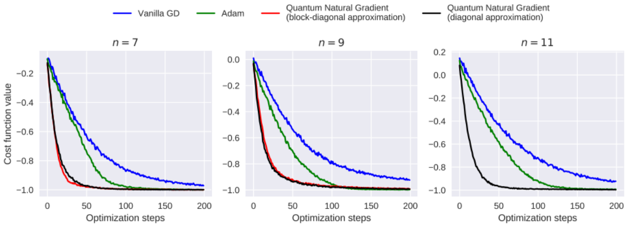


Figure 1: The cost function value for $n = 7, 9, 11$ qubits and $l = 5$ layers as a function of training iteration for four different optimization dynamics. 8192 shots (samples) are used per required expectation value during optimization.

- **Observation**: QNG (red = block-diag, black = diagonal) converges much faster than Vanilla GD (blue) and Adam (green).
- **Trend**: performance gap increases as qubit number grows.

Experimental results: Figure 2

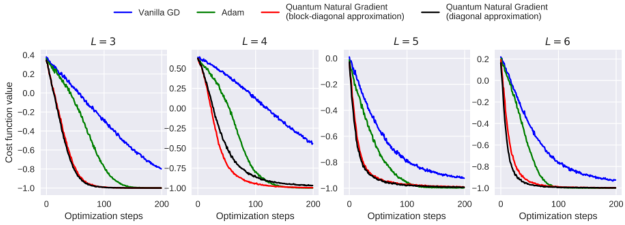


Figure 2: The cost function value for $n = 9$ qubits and $l = 3, 4, 5, 6$ layers as a function of training iteration for four different optimization dynamics. 8192 shots (samples) are used per required expectation value during optimization.

- **Observation**: QNG (red/black) consistently outperforms GD and Adam across circuit depths.
- **Comparison**: Block-diag (red) slightly better than diagonal (black), but both far superior to classical optimizers.