Trainability Bounds

Variational Quantum Algorithms · Module 8 — Expressivity Vs Trainability

Node 8.4

Position in Knowledge Graph

System: Variational Quantum Algorithms **Structural Domain:** Expressivity vs Trainability **Node:** 8.4

This is the **trainability counterpart** of nodes 8.1-8.3. Where those nodes asked "how big is the family of states an ansatz can reach?", this one asks "how hard is it to optimize over that family?"

The two questions are not independent. Module 4 already showed that a fully expressive (Haar 2-design) ansatz has barren plateaus. This node makes the relationship **quantitative**: it states the trainability bounds in their precise forms, walks through the most important variants, and connects each one to the corresponding expressivity property.

The result of this node is the inputs to node 8.5, which combines them into the **expressivity-trainability tradeoff curve** that is the central object of Module 8.

What "Trainability" Means

Several closely-related but distinct quantities can serve as "trainability":

1. Gradient variance. $\mathrm{Var}_{\boldsymbol{\theta}}[\partial_j C(\boldsymbol{\theta})]$ for randomly initialized parameters. Small variance = barren plateau = low trainability.

2. Gradient signal-to-noise ratio. $\mathbb{E}[\partial_j C]^2/\mathrm{Var}[\partial_j C]$ — how big is the expected gradient compared to its fluctuation? Low SNR = the gradient is dominated by noise.

3. Expected number of iterations to convergence. Operationally — how many optimizer steps does it take to reach a target accuracy? More iterations = lower trainability.

4. Probability of successful initialization. Out of a random sample of initial parameters, what fraction lead to successful training? Lower fraction = lower trainability.

5. Worst-case landscape geometry. Are there many local minima? Saddle points? Flat regions? More such structures = lower trainability.

The first one — **gradient variance** — is the standard quantitative measure in the VQA literature, and the one most barren plateau theorems are stated for. The others are operationally important but less amenable to formal analysis.

The Master Trainability Inequality

The most general statement, due to Cerezo et al. (2021) and refined by McClean et al. and Holmes et al.:

Theorem (general gradient variance bound). For an ansatz that forms an ϵ-approximate state-2-design, and a global cost $C = \langle\psi|H|\psi\rangle$:

$$\mathrm{Var}_{\boldsymbol{\theta}}[\partial_j C] \leq \frac{2\|H\|_2^2}{2^n - 1} + O(\epsilon).$$

For a true 2-design ($\epsilon = 0$), the bound is exactly $2\|H\|_2^2/(2^n - 1)$ — exponentially small in n.

This is the master inequality. All other trainability bounds in the VQA literature can be viewed as refinements that:

1. **Tighten the bound** for non-2-design ansätze.
2. **Loosen the assumption** to cover specific structured cost functions or ansatz families.
3. **Replace** the 2-design assumption with a Lie-algebraic condition (Ragone et al. 2024).
4. **Add corrections** for noise (Wang et al. 2021).

Bound 1: Original McClean Bound (Global Cost, Deep HEA)

The original barren plateau theorem of McClean et al. 2018:

$$\mathrm{Var}[\partial_j C_{\mathrm{global}}] \in O(2^{-n}).$$

Assumptions: the ansatz at depth $O(n)$ forms an approximate 2-design, the cost is global (sum of single-qubit operators on all qubits), parameters are uniformly initialized.

Implication: the gradient is exponentially small. To resolve it above shot noise requires $O(4^n)$ shots per evaluation. Infeasible.

This is the canonical statement and the one most often cited. Module 4 covered it in detail.

Bound 2: Cerezo Local-Cost Refinement

For a *local* cost function (sum of single-qubit observables), the bound improves dramatically.

Theorem (Cerezo et al. 2021). For $C_{\mathrm{local}} = \sum_i \langle\psi|O_i|\psi\rangle$ where each O_i acts on $O(\mathrm{poly}\log n)$ qubits:

$$\mathrm{Var}[\partial_j C_{\mathrm{local}}] \in \Omega(\mathrm{poly}(1/n)),$$

provided the ansatz is "shallow" (depth $O(\mathrm{poly}\log n)$) and the cost is built from local observables.

Implication: local costs evade the barren plateau in shallow ansätze. The polynomial lower bound on gradient variance is enough for gradient descent to make progress.

Caveat: as the ansatz becomes deeper than $O(\log n)$, even the local-cost bound deteriorates. The refinement only works in the *shallow* regime.

This is the most important practical refinement. It justifies using **depth-restricted, local-cost** VQAs as the default, with global-cost variants only when necessary.

Bound 3: Marrero Entanglement Bound

A more physical view: trainability is related to the **average entanglement** of the ansatz states.

Theorem (Marrero-Kieferová-Wiebe 2021). Let $S(\rho_A)$ be the average entanglement entropy of the reduced state on subsystem A (averaged over parameter initializations). Then

$$\mathrm{Var}[\partial_j C] \leq f(\dim A) \cdot 2^{-S(\rho_A)},$$

where f is a function of subsystem size only.

Implication: high entanglement → small gradient variance → barren plateau. The "volume law" entanglement of Haar-random states is exactly the case where S scales linearly in n, giving $2^{-S} = 2^{-O(n)}$ — barren plateau.

This bound provides the *physical mechanism* for the plateau (entanglement spreading) and a recipe for avoiding it: keep the entanglement low (area-law or log-law).

Bound 4: Ragone Lie-Algebraic Bound

The most recent and theoretically clean variant. Replaces the 2-design assumption with a Lie-algebraic condition.

Theorem (Ragone et al. 2024). Let $\mathfrak{g}$ be the dynamical Lie algebra of the ansatz (Section 8.1). Then

$$\mathrm{Var}[\partial_j C] \leq \frac{P_{\text{purity}}(\mathfrak{g}, H)}{\dim \mathfrak{g}},$$

where P_{purity} is a "purity" of the cost Hamiltonian relative to the algebra (a combinatorial quantity that can be computed from $\mathfrak{g}$ and H).

Implication: trainability is governed by the dimension of the dynamical Lie algebra. Small DLA → large variance → trainable. Large DLA → small variance → barren plateau.

This is the **cleanest form** of the bound because:

1. It does not assume the ansatz forms a 2-design (which is hard to verify).
2. It works for *any* polynomial-dimension DLA, not just specific families.
3. It connects directly to the expressivity theory (8.1) — the same DLA dimension that controls expressivity also controls trainability.

This unifies expressivity and trainability under a single Lie-algebraic framework, and is the basis for the explicit tradeoff curve in 8.5.

Bound 5: Wang Noise-Induced Plateau

A very different bound, capturing the noise-induced plateau (Module 4 node 4.5).

Theorem (Wang et al. 2021). For an ansatz of depth L under depolarizing noise with rate p per gate:

$$\mathrm{Var}[\partial_j C] \leq C_0 \cdot (1-p)^{2L},$$

for a constant C_0.

Implication: the gradient variance decays *exponentially in depth times noise rate*. Even on a non-2-design ansatz, sufficient depth and noise will create a barren plateau.

Distinction from the unitary plateau: this bound has nothing to do with expressivity or 2-designs. It is a *purely noise-driven* effect. Increasing the ansatz expressivity does not change this bound; only reducing depth or noise does.

This bound matters because it sets the **noise-corrected ceiling** on practical ansatz depth: even if you have a structured (low-DLA) ansatz, deep noise will plateau it.

Bound 6: Local Approximate 2-Designs

A partial-design refinement: ansätze that are approximate 2-designs only on a *subspace*.

Theorem. If the ansatz is a 2-design on a k-dimensional subspace of the full 2^n-dimensional Hilbert space, then for cost functions whose support is contained in that subspace:

$$\mathrm{Var}[\partial_j C] \in \Theta(1/k).$$

Implication: if you can restrict the ansatz to a polynomially-large subspace (e.g., particle-number conservation gives $k = \binom{2n}{n}$, super-polynomial but much less than 2^{2n}), the trainability is only polynomially small, not exponentially.

This is the formal statement of "symmetry-restriction helps." Module 2's UCCSD ansatz uses particle-number conservation; Module 4 explained the heuristic; this bound provides the rigorous version.

The Bounds As A Hierarchy

The bounds form a hierarchy. From most pessimistic to most optimistic:

1. **Wang (noise-induced):** $(1-p)^{2L}$ — applies whenever there is gate noise.
2. **McClean (deep HEA, global cost):** $O(2^{-n})$ — barren plateau.
3. **Ragone (Lie algebraic):** $O(1/\dim \mathfrak{g})$ — depends on ansatz structure.
4. **Marrero (entanglement):** $O(2^{-S})$ — depends on average entanglement.
5. **Subspace 2-design:** $O(1/k)$ — depends on accessible subspace size.
6. **Cerezo (local cost, shallow):** $\Omega(\mathrm{poly}(1/n))$ — trainable.

The right bound depends on the situation:

- **Are you using a deep generic ansatz with global cost?** → McClean / Ragone.
- **Are you using a deep ansatz with local cost?** → Cerezo or Wang (depth/noise).
- **Are you using a structured ansatz with symmetries?** → Subspace 2-design.
- **Are you on noisy hardware?** → Wang dominates.

The practical question is: which of these bounds is *tightest* for your specific case?

How To Use These Bounds In Practice

A workflow for ansatz design using these bounds:

Step 1: Compute (or estimate) the DLA dimension of your candidate ansatz. If polynomial in n, Ragone gives a polynomial bound — likely trainable.

Step 2: Check the cost function locality. If global, you'll need either a shallow ansatz or a structured DLA. If local, Cerezo gives polynomial trainability for shallow circuits.

Step 3: Estimate ansatz entanglement. If average entanglement is volume-law, Marrero predicts a plateau. If area-law or log-law, you're likely fine.

Step 4: Account for noise. Multiply by Wang's $(1-p)^{2L}$ factor. If your depth times noise rate exceeds 1, the noise plateau dominates.

Step 5: Verify with empirical gradient variance. Sample 100 random initializations, compute the gradient norm at each, look at the distribution. If most are below the shot noise floor, you have a plateau in practice.

This workflow is what experienced VQA practitioners do, often informally. The bounds in this node make it formal.

Lower Bounds Vs Upper Bounds

A subtle but important point: **all these are upper bounds on gradient variance**.

An upper bound says "the variance is at most X." If X is small, you're guaranteed a plateau. A lower bound would say "the variance is at least X." If X is large, you're guaranteed *not* to have a plateau.

The literature has many upper bounds (showing plateaus are inevitable in various regimes) and very few lower bounds (showing trainability is guaranteed in various regimes). The Cerezo bound for shallow local-cost circuits is one of the few clean lower bounds.

This asymmetry reflects the difficulty: it's easier to prove "this thing is bad" than "this thing is good." For practical VQA design, the lack of clean lower bounds means you often have to verify trainability empirically rather than rely on theory.

Structural Role

This node is the **trainability theorem dump** of Module 8. It collects the formal trainability bounds, organizes them into a hierarchy, and provides the practitioner workflow for using them.

It directly enables node 8.5 (the explicit expressivity-trainability tradeoff curve), which combines the bounds in this node with the expressivity metrics in 8.1-8.2 to produce the central object of Module 8.

Relations to Other Concepts

- **McClean et al. 2018** — original barren plateau theorem.
- **Cerezo et al. 2021** — local cost refinement.
- **Marrero-Kieferová-Wiebe 2021** — entanglement view.
- **Ragone et al. 2024** — Lie-algebraic theory.
- **Wang et al. 2021** — noise-induced plateaus.
- **Module 4 (Barren Plateaus)** — the qualitative version of all this.

Worked Example — Comparing Bounds On A 6-Qubit Ansatz

Consider a 6-qubit Hardware-Efficient Ansatz with 4 layers and a global cost $C = \langle Z^{\otimes 6} \rangle$.

McClean bound: $\mathrm{Var}[\partial_j C] \leq 2 \cdot 1/(2^6 - 1) = 2/63 \approx 0.032$.

The shot noise floor at $S = 1000$ shots is $\sim 1/\sqrt{S} \approx 0.032$. The gradient is **right at the noise floor**. Training will be very hard.

Cerezo bound (if cost were local): if $C = \sum_i \langle Z_i \rangle$ instead, the bound would be $\Omega(1/n^c)$ for some small c, giving Var $\approx 0.1 - 0.5$ — well above the noise floor. Training would be feasible.

Ragone bound: the DLA of a 4-layer 6-qubit HEA is approximately $2^{12} = 4096$ (full unitary algebra). $P_{\text{purity}} \approx 1$. So Var $\leq 1/4096 \approx 0.0002$. **Tighter than McClean** — predicts a worse plateau than McClean's bound. Empirically, this is consistent with what's observed.

Wang bound at $p = 10^{-3}$ noise: $(1 - 10^{-3})^{2 \cdot 24} \approx 0.95$. Negligible noise contribution at this depth.

Marrero bound: average entanglement of a 4-layer HEA on 6 qubits is approximately the Page value, $S \approx 3$. So $2^{-3} = 0.125$. Looser than McClean but in the same regime.

The verdict: for this configuration, all bounds say "barren plateau." Switching to a local cost (Cerezo) or restricting to a structured ansatz (smaller DLA in Ragone) would make it trainable.

Visualization

Pending. A log-scale plot of gradient variance vs qubit count for: (a) McClean (deep HEA global), (b) Cerezo (shallow HEA local), (c) Ragone (depends on DLA dim), (d) Wang (depends on depth and noise). Each is a different curve, with the shot noise floor marked. Caption: "Different bounds; different regimes; different conclusions."

Exercises

1. For a 4-qubit HEA with 2 layers and global cost $\langle Z_0 Z_1 Z_2 Z_3 \rangle$ with $\|H\| = 1$, what does McClean's bound give? Compare to a shot noise floor at 10000 shots.
2. The Ragone bound is $O(1/\dim \mathfrak{g})$. For a UCCSD ansatz on H_2 (2 electrons in 4 spin orbitals), $\dim \mathfrak{g} = ?$ (Hint: it's the dimension of the particle-number-conserving subalgebra.) What does the bound predict for trainability?
3. (VQA) For a 10-qubit HEA at depth 30 with depolarizing noise $p = 10^{-3}$, is the gradient plateau dominated by McClean (unitary) or Wang (noise)? Show your reasoning.

Research Extensions

- **Lower bounds for structured ansätze** — proving trainability rather than disproving it.
- **Bounds for non-i.i.d. parameter initializations** — most theorems assume uniform; what about smarter init?
- **Bounds under realistic hardware noise** — beyond depolarizing.
- **Bounds for hybrid optimizers** — how does HOPSO change the effective trainability bound?

Cross-links to Other Courses

- **Random Matrix Theory** — Haar measures, t-design moments.
- **Lie Theory** — dynamical Lie algebras (Ragone).
- **Module 4 (Barren Plateaus)** — the qualitative discussion these bounds formalize.
- **Module 6 (Optimization Dynamics)** — how the bounds affect optimizer choice.

Variational Quantum Algorithms · Module 8 — Expressivity Vs Trainability · Node 8.4
Concepts: barren-plateaus, variance-and-covariance, concentration-of-measure
Prerequisites: 4.2, 8.1, 8.2
Enables: 8.5

hbar.university — own.your.cognition