Optimal Transport and Optimization

Variational Quantum Algorithms · Module 6 — Optimization Dynamics

Node 6.6

Position in Knowledge Graph

System: Variational Quantum Algorithms **Structural Domain:** Optimization Dynamics **Node:** 6.6

This is the most **abstract** node in Module 6, and one of the most thesis-aligned.

The standard view of optimization is: pick a starting point in parameter space, follow a descent direction, end up at a local minimum. The **optimal transport** view is different: the optimizer is moving a *probability distribution* over parameter space, not a single point. The "trajectory" is a path of distributions, evolving according to a flow that decreases the expected cost.

This view is more general than the point-based view. It makes population-based optimizers (Section 6.7) look like a natural special case rather than an exotic alternative. It connects to the **gradient flow** view of differential equations, the **Wasserstein metric** on distributions, and the thesis's regularization framework.

This node introduces the optimal transport view of optimization, the technical machinery (Wasserstein gradient flow, JKO scheme), and explains why this perspective matters for VQA.

From Points To Distributions

In standard optimization, the optimizer's state is a single point $\boldsymbol{\theta}_k \in \mathbb{R}^P$.

In the distributional view, the optimizer's state is a probability distribution μ_k over $\mathbb{R}^P$. The state evolves over time according to a rule that depends on the cost function C.

A point-based optimizer is a special case: $\mu_k = \delta_{\boldsymbol{\theta}_k}$, a delta function at the current parameter. The distributional view collapses to the point view when the distribution has no spread.

A population-based optimizer is also a special case: $\mu_k = \frac{1}{N}\sum_{i=1}^{N} \delta_{\boldsymbol{\theta}_k^{(i)}}$, an empirical distribution from N particles. The distributional view captures the population's collective behavior.

A Bayesian optimizer is yet another case: μ_k is the posterior over parameters given the observed costs.

The distributional view unifies these.

Wasserstein Gradient Flow

The principal mathematical tool for distributional optimization is **Wasserstein gradient flow**.

The Wasserstein-2 distance between two distributions μ and ν is

$$W_2(\mu,\nu) = \left(\inf_{\pi \in \Pi(\mu,\nu)} \int \|x-y\|^2 \, d\pi(x,y) \right)^{1/2},$$

where $\Pi(\mu, \nu)$ is the set of joint distributions with marginals μ and ν. Intuitively, W_2 measures the "cost of transporting mass" from μ to ν, where the cost is squared Euclidean distance.

The space of distributions equipped with W_2 is a *metric space* (in fact, a Riemannian-like manifold called Wasserstein space).

A **Wasserstein gradient flow** is a curve μ_t in Wasserstein space that decreases a functional $\mathcal{F}[\mu]$ at each instant. The defining equation is

$$\partial_t \mu_t = -\nabla_{W_2} \mathcal{F}[\mu_t].$$

For $\mathcal{F}[\mu] = \int C(\boldsymbol{\theta})\, d\mu(\boldsymbol{\theta})$ (the expected cost under μ), the Wasserstein gradient flow is

$$\partial_t \mu_t = \nabla \cdot (\mu_t \nabla C),$$

which is a transport equation: each "particle" of mass moves with velocity $-\nabla C$, and the distribution as a whole flows downhill.

This is the continuous-time, distributional analogue of gradient descent.

Discretization: The JKO Scheme

To turn the continuous flow into an algorithm, you discretize in time. The canonical discretization is the **Jordan-Kinderlehrer-Otto (JKO) scheme**:

$$\mu_{k+1} = \arg\min_{\nu} \left[\mathcal{F}[\nu] + \frac{1}{2h} W_2^2(\nu, \mu_k) \right].$$

In words: move from μ_k to a new distribution ν that decreases $\mathcal{F}$ but doesn't move too far in Wasserstein distance.

This is a variational formulation: each step solves a small optimization problem (minimize cost plus transport penalty). The transport penalty acts as a regularizer that limits how much the distribution can change in one step — analogous to the trust-region constraint in classical optimization.

In the limit $h \to 0$, the JKO sequence converges to the continuous Wasserstein gradient flow.

For VQA: the JKO scheme can be implemented by an optimizer that maintains an empirical distribution (a population of points), updates it via cost-decreasing moves, and ensures the distribution doesn't change too much per step. Population methods like CMA-ES and particle swarm can be viewed as approximate JKO schemes.

Why This Matters For VQA

The optimal transport view brings several benefits to VQA optimization:

1. Natural framework for population methods. Population optimizers (HOPSO, CMA-ES, genetic algorithms) are point-based at the algorithm level but distributional at the conceptual level. The Wasserstein gradient flow gives them a clean theoretical foundation.

2. Better handling of multi-modal landscapes. A distributional optimizer can have mass at multiple local minima simultaneously, and the relative weights can shift over time as the optimizer learns which minimum is best. A point-based optimizer is committed to one location.

3. Natural connection to regularization. Many regularization terms (entropy regularization, trust regions, KL penalties) are naturally distributional. Adding them to the JKO objective gives a clean way to design **regularized population optimizers**, which is one of the threads of the thesis.

4. Plateau-aware exploration. A distributional optimizer with high entropy (broad distribution) explores plateau regions more efficiently than a point optimizer, because it samples multiple parameter values simultaneously and can discover gorges or non-plateau regions without making sequential decisions.

5. Connection to physics. Wasserstein gradient flow is the natural continuous-time limit of Brownian motion in a potential, which is a model from statistical physics. This gives a physical intuition for the optimizer dynamics — the optimizer is a "particle" moving in a "potential landscape" with "thermal noise."

Entropy Regularization And Exploration

A particularly important variant: **entropy-regularized JKO**.

Modify the JKO objective to include an entropy term:

$$\mu_{k+1} = \arg\min_{\nu} \left[\int C \, d\nu - \tau H[\nu] + \frac{1}{2h} W_2^2(\nu, \mu_k) \right],$$

where $H[\nu] = -\int \nu \log \nu \, d\nu$ is the differential entropy and $\tau > 0$ is a temperature parameter.

The continuous-time limit is the **Fokker-Planck equation**:

$$\partial_t \mu_t = \nabla \cdot (\mu_t \nabla C) + \tau \Delta \mu_t.$$

This is the equation of a Brownian particle in the potential C at temperature τ. The optimizer **explores the landscape** with thermal-like fluctuations.

Properties:

- High τ: broad distribution, lots of exploration, slow descent.
- Low τ: narrow distribution, fast descent, prone to local minima.
- $\tau \to 0$: recovers the deterministic Wasserstein gradient flow.

Annealing τ over time (start high, decrease) is analogous to **simulated annealing**, and gives a principled exploration-exploitation schedule.

For VQAs, entropy-regularized population optimizers can escape barren plateaus by maintaining enough diversity (high entropy) to keep some particles outside the plateau region, where they continue to make progress and pull the rest of the population along.

Sinkhorn And Practical Wasserstein

In practice, computing Wasserstein distances exactly is expensive. The standard practical tool is **Sinkhorn divergence** (Cuturi 2013), which is an entropy-regularized approximation that can be computed in $O(n^2)$ for n particles via fixed-point iteration.

Sinkhorn-based Wasserstein gradient flows have become a workhorse in machine learning (generative models, domain adaptation, etc.) and are starting to appear in VQA as well.

For population-based VQA optimizers, Sinkhorn-based updates allow you to maintain the distributional view computationally.

The Thesis Connection

The thesis argues that **regularization should be viewed as dynamical modification rather than objective modification**. The Wasserstein gradient flow framework makes this explicit:

- **Objective-side regularization:** modify $\mathcal{F}[\mu] = \int C\, d\mu$ by adding a penalty term, e.g., $\mathcal{F}[\mu] = \int C\, d\mu + \lambda R[\mu]$.
- **Dynamics-side regularization:** modify the *flow equation* directly, e.g., add diffusion: $\partial_t \mu = \nabla \cdot (\mu \nabla C) + \tau \Delta \mu$.

These are *not* the same. Adding entropy penalty to the objective is not the same as adding diffusion to the flow — they have different stationary distributions and different transient behavior. The thesis dwells on this distinction at length, and Module 6 (this node specifically) is where it gets the cleanest statement.

The optimal transport view is **the natural language** in which to make this distinction precise.

Limitations And Practical Considerations

The Wasserstein view is conceptually beautiful but has practical limitations:

1. Cost. Maintaining a distribution requires more state (a population of particles, or a density estimate) than maintaining a single point. Each iteration is more expensive.

2. Convergence rates. Wasserstein gradient flow has its own convergence theory, but the rates are typically slower than point-based optimizers in the deterministic regime. The benefit shows up in non-convex / multi-modal / noisy settings, not in clean convex problems.

3. Discretization artifacts. JKO discretization introduces its own errors, and the Sinkhorn approximation introduces more. The distributional view is mathematically clean but algorithmically messier than point-based methods.

4. Limited adoption. Most VQA software packages don't have first-class support for distributional optimizers. You'd typically implement them by hand or use a population optimizer (Section 6.7-6.8) as the practical proxy.

For these reasons, optimal-transport-based VQA optimization is more of a theoretical viewpoint than a practical tool in 2026. But it's an important viewpoint, because it grounds population methods and dynamics-side regularization in a coherent framework.

Structural Role

This node is the **distributional viewpoint** of Module 6. It abstracts the point-based optimizers (6.1-6.5) into a distributional framework, motivates population methods (6.7-6.8) as natural special cases, and provides the cleanest setting for the thesis's distinction between objective-side and dynamics-side regularization.

Relations to Other Concepts

- **Wasserstein metric** — the distance on the space of distributions.
- **JKO scheme (Jordan-Kinderlehrer-Otto 1998)** — the variational discretization.
- **Fokker-Planck equation** — the entropy-regularized continuous flow.
- **Sinkhorn algorithm (Cuturi 2013)** — practical Wasserstein computation.
- **Simulated annealing** — the analogue of entropy-regularized flow with annealed temperature.

Worked Example — Wasserstein Gradient Flow For A 1D Cost

Consider $C(\theta) = \cos(\theta)$ on $[-\pi, \pi]$, with global minimum at $\theta = \pi$ (or equivalently $-\pi$).

A point optimizer starting at $\theta_0 = 0.5$ follows GD: $\theta_{k+1} = \theta_k - \eta \cdot (-\sin(\theta_k))$, reaching the minimum.

A distributional optimizer starting at $\mu_0 = \mathcal{N}(0.5, 0.5^2)$ (a Gaussian centered at 0.5 with std 0.5) follows the JKO scheme. Each iteration: - Compute the expected cost $\int C \, d\mu_k$ and its gradient. - Move the distribution in the descent direction, with a Wasserstein penalty limiting the step.

The distribution slowly drifts toward $\theta = \pi$, simultaneously narrowing as the entropy decreases (in unregularized JKO) or staying broad (in entropy-regularized JKO).

After many steps, the unregularized JKO converges to a delta at π, while entropy-regularized JKO converges to a Boltzmann distribution $\exp(-C/\tau)$ concentrated near π but with some spread.

For a multi-modal landscape (e.g., $C(\theta) = \cos(2\theta)$ with minima at $\pi/2$ and $-\pi/2$), the distributional optimizer can have mass at both minima simultaneously, while the point optimizer is committed to one.

This is the conceptual benefit: distributional optimizers handle multi-modal landscapes natively.

Visualization

Pending. A 1D plot showing the evolution of μ_t over time on a non-convex $C(\theta)$ landscape. Initial Gaussian, then progressively concentrated at the global minimum. Side-by-side with a point-based GD trajectory. Caption: "The distribution flows downhill. The point only moves once."

Exercises

1. For $C(\theta) = \theta^2$ and $\mu_0 = \mathcal{N}(2, 1)$, compute the JKO update with $h = 0.1$. What is μ_1?
2. The Fokker-Planck equation has stationary distribution $\mu^* \propto \exp(-C/\tau)$. Verify this for $C(\theta) = \theta^2$. What is the limit as $\tau \to 0$?
3. (VQA) Sketch how an entropy-regularized population optimizer would behave on a 2D barren plateau landscape with a small "gorge" containing the global minimum. Compare to a point-based optimizer with random init.

Research Extensions

- **Sinkhorn-based VQA optimizers** — practical population optimizers with Wasserstein structure.
- **Wasserstein natural gradient** — combining the Fubini-Study metric with the Wasserstein metric.
- **Entropy regularization for barren plateaus** — using temperature annealing to escape plateaus.
- **JKO schemes for VQA** — algorithmic implementations of the JKO discretization tailored to quantum costs.

Cross-links to Other Courses

- **Optimal Transport (Villani, Peyré-Cuturi)** — the underlying mathematical theory.
- **Statistical Mechanics** — Brownian motion, Fokker-Planck, thermal exploration.
- **Generative Models in ML** — Wasserstein GANs and related distributional methods.
- **Module 5 (Regularization)** — the dynamics-side framework.
- **Module 6 Section 6.7 (Population-Based Optimizers)** — the practical embodiment.

Variational Quantum Algorithms · Module 6 — Optimization Dynamics · Node 6.6
Concepts: stochastic-optimization, gradient-descent, natural-gradient
Prerequisites: 6.2, 6.5
Enables: 6.7

hbar.university — own.your.cognition