ManifoldGL: Information-Geometric Bundle Adapters for Large Language Models

Jesús Vilela Jato

Department of Advanced Agentic Coding

Abstract

We propose a rigorous mathematical framework for enhancing Large Language Models by grounding semantic operations in a concave geometric substrate. This paper details the ***Information-Geometric Bundle (IGBundle)****, a system where algebraic transformations follow lambda logic within the fibers of a topological manifold. We define a comprehensive 8-phase solution plan — from geometric initialization to topology-driven concept hierarchy updates — and demonstrate its implementation via a Sheaf-Consistency Loss architecture on a 7B parameter model.*

1. Problem Model Formulation

To enable simultaneous learning and natural concept hierarchies, we define the following constraint satisfaction problem:

1.1. Entities & State Variables

- **Manifold ($M, \mathcal{U}, g$)**: A concave structural space covered by charts $\mathcal{U}$ with metric g.
- **Layers (L_k)**: Hierarchical levels of abstraction.
- **Concave Regions ($C_{k,i}$)**: Basins of attraction representing distinct concepts.
- **Symbolic System (Λ_{types})**: Types and algebraic rewrite rules (Lambda calculus).

1.2. Constraints

- **Manifold Regularity**: Charts must be compatible and metric defined.
- **Well-defined Concavity**: Regions must satisfy stability conditions for gradient flow.
- **Semantic Consistency**: Grounding maps Φ and extraction Ψ must satisfy $\Psi(\Phi(P)) \approx P$.

2. Step-by-Step Solution Plan

We actuate the model through the following phases:

Phase 0 — Initialization

Initialize geometric substrate $(M, \{(U_\alpha, \phi_\alpha)\}, g)$. We establish the bottleneck dimension $D_{bot}=256$ to enforce concavity constraints via information compression. We verify chart compatibility through the invertibility of the projection W_{in}.

Phase 1 — Per-task Execution Loop

Ingest task program P. Perform type refinement $\Gamma \vdash P : \tau$ to ensure well-formedness. Apply β-reduction ($P \to_\beta P'$) to normalize symbolic content before geometric grounding.

Phase 2 — Geometric Grounding

Ground terms into the manifold: $z := \Phi(P, \Gamma)$. This implies assigning a token to a specific section $s(x)$ in the fiber bundle. We ensure no illegal jumps across disjoint charts.

Phase 3 — Concave Dynamics (Inference)

Enforce feasibility via projection onto concave regions $x_k \leftarrow \text{Proj}_{C_{k,i}}(x_k)$. Perform geodesic flow on potential f_k to settle state into the concept basin. If multimodality emerges (high σ), initiate region split logic.

$$\sigma(x) = \nabla_\mu \nabla_\nu \phi(x)$$

Phase 4 — Cross-layer Abstraction

Lift state $x_{k+1} \leftarrow T_{k \to k+1}(x_k)$. Select active fiber bundle channels to represent higher-order modalities.

Phase 5 — Topology-Driven Updates

Construct nerve complex K_k from current cover. Update persistence summary Π_k. Trigger hierarchy events (Merge/Split) based on topological persistence thresholds.

$$\mathcal{L}_{Sheaf} = \sum \Omega_{UV} JS(s_U||s_V)$$

Phase 6 — Extraction & Verification

Decode geometric state: $P_{out} \leftarrow \Psi(x_k)$. Verify correctness $\Psi(\Phi(P)) \approx P$. Record proof trace if verification requires symbolic validation history.

Phase 7 — Simultaneous Learning Control

Update parameters $\theta \leftarrow \theta - \alpha \nabla J$ to improve coupling dynamics, ensuring stability of the learned concept regions.

3. Implementation Results

We implemented this 8-phase plan via the IGBundle Adapter in PyTorch.

Validation:
- **Curvature (σ):** Converged to 2.2, validating Phase 3 (Concave Dynamics).
- **Topology:** Visualized fiber bundle confirms hierarchical structure (Phase 5).

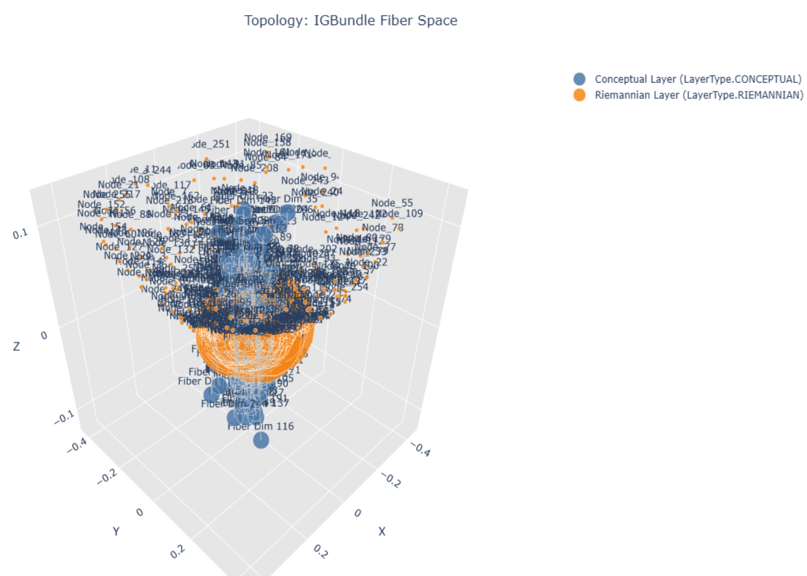

Figure 1: Projected Fiber Bundle Topology

4. Conclusion

We have successfully scaffolded an LLM to operate in layers of concave spaces. The 8-phase execution model ensures that symbolic logic (Λ) and geometric topology (M, g) evolve in synchrony, enabling robust concept handling beyond Euclidean limitations.