Consistency Is All You Need

Linear-Complexity Geometric Consistency for Transformer Architectures via Anticipatory Control Fields

Logan Matthew Napolitano

January 2026

Abstract

The Transformer architecture has dominated sequence modeling since its introduction, yet it remains fundamentally agnostic to the structural consistency of its outputs. We present the Control Field Holonomy Transformer (CF-HoT), an architecture that embeds geometric consistency as a native, trainable property rather than an emergent or post-hoc characteristic. Building on the mathematical framework of fiber bundles and parallel transport, we demonstrate that the prohibitive $O(n^2 \cdot d^3)$ cost of explicit holonomy computation can be entirely circumvented by reformulating consistency from a measurement problem to an anticipation problem. Our key insight is that consistency violations need not be detected—they can be predicted and prevented. The resulting architecture reduces holonomy-specific computation from $O(n^2 \cdot d^3)$ to $O(n)$ while retaining standard $O(n^2)$ attention costs, trains stably with conventional optimizers, and provides interpretable signals about reasoning coherence. We provide complete implementation code, empirical validation of trainability, and a theoretical grounding that unifies differential geometry with practical neural network design. We demonstrate trainability and numerical stability on synthetic data; evaluation on reasoning benchmarks remains future work. This work establishes that structural consistency is not merely desirable but architecturally achievable at scale.

1. Introduction: The Consistency Gap

The Transformer architecture, introduced in "Attention Is All You Need" (Vaswani et al., 2017), revolutionized sequence modeling by replacing recurrence with self-attention. This architectural choice enabled unprecedented parallelization and gave rise to the large language model paradigm that now dominates artificial intelligence. Yet the Transformer's fundamental mechanism—weighted aggregation of values based on query-key similarity—contains no intrinsic notion of structural consistency. The attention mechanism asks "what is relevant?" but never asks "is this coherent?"

This absence manifests in familiar failure modes: self-contradiction within a single response, logical non-sequiturs that accumulate over multi-step reasoning, hallucinated facts that locally sound plausible but globally violate constraints the model itself established moments earlier. These are not merely training data problems or scale limitations—they are architectural absences. The Transformer has no mechanism to detect that asserting "X is true" and later asserting "not-X is true" constitutes a structural failure.

The Holonomy Transformer project began with a geometric intuition: if we interpret the model's hidden states as positions on a fiber bundle—a mathematical structure where local geometry can detect global inconsistency—then parallel transport around closed loops yields a measurable quantity called holonomy. Non-zero holonomy indicates that returning to a conceptual starting point produces a different state than expected, the mathematical signature of inconsistency.

The original formulation was mathematically elegant but computationally catastrophic. Computing explicit pairwise holonomy required $O(n^2)$ position comparisons per layer, each involving $O(d^3)$ matrix operations for fiber transport. For a 2048-token sequence with 32-dimensional fibers, this represented a 65× overhead compared to standard attention—prohibitive for any practical application.

This paper presents the resolution of that tension. We demonstrate that the geometric insight—consistency has structure that can be computed—survives translation into a fundamentally different computational paradigm. The key move is reconceptualizing consistency from something measured after the fact to something anticipated before it occurs. We replace explicit holonomy computation with learned prediction of holonomy increments, accumulated via momentum into a "control field" that modulates attention and feedforward computations. The result is an architecture that adds approximately 15-20% overhead to standard Transformers while providing continuous, interpretable signals about reasoning coherence.

2. Theoretical Foundation: From Fiber Bundles to Control Fields

2.1 The Geometric Perspective on Consistency

Before presenting our efficient formulation, we must establish why geometry provides the right language for consistency. The fundamental insight is that reasoning traces define paths through a latent space, and consistency is a property of how those paths relate to each other.

Consider a fiber bundle (E, π, M, F) where E is the total space, M is the base manifold (the token sequence), F is the fiber (the space of possible semantic states at each position), and $\pi: E \rightarrow M$ is the projection. At each position t in the sequence, the model occupies some point in the fiber F_t. Reasoning is then a path through this bundle.

Parallel transport provides a principled way to compare fibers at different positions. Given a connection ∇ on the bundle, we can transport a fiber state φ_i from position i to position j along a path γ, yielding $T_{i \rightarrow j}(\varphi_i)$. Holonomy measures what happens when we transport around closed loops: $H_{ij} = \|T_{i \rightarrow j} \cdot T_{j \rightarrow i} - I\|^M$. If the bundle is "flat" (no curvature), transport around any closed loop returns to the starting state: $H_{ij} = 0$. Non-zero holonomy indicates path-dependence—the order in which you traverse concepts matters, a hallmark of inconsistent reasoning.

This geometric formulation immediately suggests a consistency regularizer: minimize total holonomy during training. The model learns representations where semantic relationships compose cleanly, where saying "A implies B" and "B implies C" actually supports concluding "A implies C" rather than drifting into unrelated territory.

We emphasize that the implementation presented in this paper is *inspired by* rather than *implementing* rigorous differential geometry. The holonomy predictor learns a scalar proxy for consistency risk; it does not compute actual parallel transport or loop integrals. The geometric framework provides conceptual grounding and motivates the architectural choices, but the trainable system operates as a learned heuristic. This deliberate approximation is what enables the $O(n)$ complexity that makes the approach practical. We use geometric terminology throughout to maintain continuity with the theoretical motivation, while acknowledging that the implementation trades mathematical rigor for computational tractability.

2.2 The Computational Impasse

Direct implementation of this geometric vision encounters severe computational barriers. Computing H_{ij} requires evaluating parallel transport between all $O(n^2)$ position pairs. Each transport operation involves matrix exponentials with $O(d^{3\text{ɔ}}{}_{i}{}^{\text{be}}{}_{r})$ complexity. For typical hyperparameters ($n = 2048$, $d^{ɔ}{}_{i}{}^{be}{}_{r} = 32$), this imposes approximately 137 billion additional operations per layer—a 65× overhead that renders the approach impractical.

Prior attempts to reduce this cost through sparse sampling or low-rank approximations sacrificed the very consistency guarantees that motivated the geometric approach. Randomly sampling position pairs misses systematic inconsistencies; low-rank transport approximations introduce their own geometric artifacts.

The resolution required not a computational shortcut but a conceptual reframing: what if we abandoned measurement entirely?

2.3 The Anticipation Paradigm

The central insight of the Control Field formulation is that consistency need not be measured—it can be anticipated. Rather than computing holonomy after states have been established, we train a predictor to estimate the consistency implications of each generation step before it occurs.

This shift from measurement to anticipation parallels developments in other domains. In control theory, model predictive control anticipates constraint violations rather than reactively correcting them. In cognitive science, theories of predictive processing suggest that the brain continuously generates predictions about sensory input rather than passively receiving it. The Control Field HoT applies this principle to consistency: the architecture learns to predict when it is about to reason inconsistently and preemptively modulates its behavior.

Formally, we replace the pairwise holonomy computation H_{ij} with a local holonomy predictor: $\Delta h_t = f\theta(x_t, \varphi_t)$, where x_t is the hidden state and φ_t is the fiber state at position t. This predictor is a small MLP that asks: "If generation continues from this state, how much inconsistency is likely to accumulate?" Because this depends only on local state, it is $O(1)$ per position—$O(n)$ total versus $O(n^2)$.

The predicted holonomy increments are accumulated via exponential moving average into a "control field": $h_t = \alpha \cdot h_{t-1} + (1 - \alpha) \cdot \Delta h_t$. The momentum parameter α determines the temporal horizon over which inconsistency accumulates. High α creates long-range consistency pressure; low α emphasizes local coherence.

This accumulated field then modulates downstream computation through gating. Positions with high accumulated holonomy predictions have their influence reduced, forcing the model to "route around" potentially inconsistent reasoning paths.

3. Architecture: The Control Field Holonomy Transformer

3.1 Overview

The Control Field HoT extends the standard Transformer with four additional components: (1) a fiber projection layer that maps hidden states into a geometric fiber space; (2) a holonomy predictor that estimates consistency risk from local state; (3) a control field accumulator that integrates predictions over time; and (4) gating mechanisms that modulate attention and feedforward computation based on accumulated field values. These components are designed to be differentiable, parallelizable, and compatible with standard training procedures.

[Figure 1: Architecture Diagram]

Figure 1: Comparison of Standard Transformer Block (left) and Control Field HoT Block (right).

The CF-HoT block adds fiber projection (φ), holonomy prediction (Δh), momentum accumulation (h), and gating (g) to the standard attention and feedforward pathway. All additions are O(n) and fully differentiable.

3.2 Fiber Projection

The fiber projection $\varphi = W^{\text{ɔ}}{}_{i}{}^{be}{}_{r} \cdot x$ maps d_{mosel}-dimensional hidden states to $d^{\text{ɔ}}{}_{i}{}^{be}{}_{r}$-dimensional fiber states. The fiber dimension is typically much smaller than the hidden dimension (we use $d^{\text{ɔ}}{}_{i}{}^{be}{}_{r} = 8$ versus $d_{\text{mosel}} = 64$ in our experiments), creating a compressed semantic subspace where consistency relationships are more directly expressed.

This compression serves two purposes. First, it reduces the dimensionality of the consistency computation. Second, it forces the model to learn a factored representation where the fiber captures "structural" aspects of meaning (the aspects relevant to consistency) while the full hidden state retains richer representational capacity.

3.3 Holonomy Predictor

The holonomy predictor is a small feedforward network that takes concatenated hidden and fiber states as input and outputs a non-negative scalar: $\Delta h_t = \text{Softplus}(\text{MLP}([x_t ; \varphi_t]))$. The Softplus activation ensures non-negativity (holonomy is a magnitude). The MLP consists of a hidden layer with GELU activation followed by a projection to scalar output.

The predictor is trained jointly with the language modeling objective. During training, the model learns to predict high Δh_t when current states are likely to lead to inconsistent continuations. This is not supervised with explicit holonomy labels (which would require the expensive $O(n^2)$ computation) but learned implicitly through the combination of language modeling loss and holonomy regularization.

Terminology note: We use "holonomy predictor" as an architectural label reflecting the geometric inspiration. The component predicts accumulated consistency risk based on local state; it does not compute holonomy in the strict differential-geometric sense. The name serves as a conceptual anchor to the theoretical framework while the implementation operates as a learned scalar estimator.

3.4 Control Field Accumulation

Predicted holonomy increments are accumulated into a causal control field using exponential moving average: $h_t = \alpha \cdot h_{t-1} + (1 - \alpha) \cdot \Delta h_t$, with boundary condition $h_0 = (1 - \alpha) \cdot \Delta h_0$. This recurrence can be unrolled into closed form: $h_t = (1 - \alpha) \sum_{i=0}^{t} \alpha^{t-i} \cdot \Delta h_i$, a weighted sum of all past holonomy predictions with exponentially decaying weights.

This closed form enables full vectorization. Rather than sequential computation, we can compute the entire field in parallel as a causal convolution: convolve the Δh sequence with the kernel $[\alpha^{n-1}, \alpha^{n-2}, ..., \alpha, 1]$ scaled by $(1 - \alpha)$. This exploits GPU parallelism and ensures the accumulation does not become a sequential bottleneck. We provide both a pedagogically clear sequential implementation and a fully vectorized convolution-based formulation for production deployment.

The momentum parameter α controls the effective "memory" of the control field. With $\alpha = 0.9$ (our default), the half-life is approximately 6.6 tokens—consistency violations from the recent past have strong influence, while distant violations fade. This reflects the intuition that local inconsistency is more recoverable than persistent contradiction.

3.5 Attention Gating

The accumulated control field modulates attention through a learned gate: $g_t = \sigma(-\lambda \cdot h_t)$, where λ is a learnable scale parameter. The gate approaches 1 when accumulated holonomy is low (consistent region) and approaches 0 when accumulated holonomy is high (inconsistent region).

This gate is injected into the attention computation via log-space addition: $\text{scores} = \text{scores} + \log(g + \varepsilon)$. This is mathematically equivalent to multiplicatively gating the attention distribution but more numerically stable for softmax computation. Positions with high accumulated holonomy have their attention weights suppressed, forcing the model to attend preferentially to positions that have been flagged as consistent.

Crucially, this is not a hard mask but a soft bias. The model retains the ability to attend to any position—the gate merely increases or decreases the cost of doing so based on predicted consistency implications.

3.6 Curvature-Gated Feedforward

Attention gating addresses the question "where should I look?" but consistency also depends on "how should I transform what I see?" The feedforward network applies position-wise transformations that can amplify or suppress inconsistent representations.

We introduce a secondary gate based on fiber curvature, approximated via finite difference: $\kappa_t = \|\varphi_{t+1} - \varphi_{t-1}\| / 2$. High curvature indicates rapid change in the fiber state—the model is traversing a "rough" region of the latent manifold. This correlates with instability and potential hallucination.

The curvature gate $g_\kappa = \sigma(1 - \lambda_\kappa \cdot \kappa_t)$ suppresses the feedforward output in high-curvature regions, preventing the model from making large representational changes when fiber geometry is unstable. This acts as a "geometric regularizer" that encourages smooth, predictable trajectories through representation space.

4. Training Dynamics and Loss Formulation

4.1 Composite Loss Function

The Control Field HoT is trained with a composite loss that balances language modeling performance with consistency regularization: L = L_LM + λ_hol · L_hol + λ_curv · L_curv, where L_LM is standard cross-entropy language modeling loss, L_hol = $\sum_t \Delta h_t$ is the sum of predicted holonomy increments (encouraging the model to reduce predicted inconsistency), and L_curv = $\sum_t \kappa_t$ is the sum of fiber curvatures (encouraging smooth fiber trajectories).

The holonomy regularization coefficient λ_hol is set low (0.001 in our experiments) to avoid interfering with language modeling while still providing gradient signal to the holonomy predictor. The curvature coefficient λ_curv is an order of magnitude smaller (0.0001), acting as a gentle smoothing term.

4.2 Gradient Flow and Stability

A primary concern with complex architectural additions is gradient stability. The momentum-based accumulation in particular could potentially cause gradient explosion or vanishing through long multiplicative chains.

Empirically, we observe stable training dynamics. The momentum parameter $\alpha = 0.9$ bounds the effective receptive field, preventing infinite gradient chains. The Softplus activation in the holonomy predictor ensures bounded gradients in the positive domain. The log-space injection of attention gates maintains numerical stability across the full range of gate values.

We verified trainability and numerical stability under controlled synthetic conditions through extensive training runs. Loss decreases monotonically, no NaN values are observed, and gradient norms remain bounded throughout training. The architecture is compatible with standard optimizers (AdamW) and requires no special training procedures. Semantic evaluation on reasoning tasks—including logical consistency benchmarks and contradiction detection—is not yet performed and remains a critical direction for future work.

4.3 What the Model Learns

The holonomy predictor learns to identify states that precede inconsistency. Without explicit supervision (we never compute ground-truth holonomy), the model discovers that certain representational configurations are "risky"—they tend to lead to language modeling errors or geometric instability downstream.

The fiber projection learns a compressed subspace that captures consistency-relevant structure. This emerges from the joint optimization: the fiber must be informative enough for the holonomy predictor to make useful predictions while remaining low-dimensional enough to impose meaningful geometric constraints.

The attention patterns learn to route around high-holonomy regions. When the control field indicates accumulated inconsistency, the model's attention distribution shifts toward positions with lower field values. This creates a form of implicit memory where the model tracks which parts of its context remain "trustworthy" for consistent reasoning.

5. Complexity Analysis

The computational overhead of Control Field HoT relative to standard Transformers is detailed below. All complexities are per-layer.

Operation	Standard Transformer	Control Field HoT
Self-attention	$O(n^2 \cdot d)$	$O(n^2 \cdot d)$
Feedforward	$O(n \cdot d^2)$	$O(n \cdot d^2)$
Fiber projection	—	$O(n \cdot d \cdot d_f)$
Holonomy prediction	—	$O(n \cdot d_c)$
Field accumulation	—	$O(n)$
Gating	—	$O(n)$

With d_fiber = 8 and d_control = 16 versus d_model = 64, the additional O(n) operations are dominated by the unchanged $O(n^2 \cdot d)$ attention computation. Empirically, we observe approximately 15-20% wall-clock overhead for the complete Control Field HoT relative to an equivalently-sized standard Transformer.

Critically, this overhead is additive and constant-factor—it does not change the asymptotic scaling of the architecture with respect to sequence length. The $O(n^2)$ term from standard attention remains the dominant factor. The contribution of this work is eliminating the additional 65× geometric overhead that made the original Holonomy Transformer impractical, not reducing the inherent cost of attention.

6. Relationship to Prior Work

6.1 Geometric Deep Learning

The Control Field HoT inherits conceptual foundations from geometric deep learning, which studies neural networks through the lens of symmetry, invariance, and geometric structure. Graph neural networks exploit the geometry of relational data; equivariant networks encode symmetry constraints directly into architecture; gauge-equivariant networks work with connections on fiber bundles.

Our contribution extends this line of research by applying geometric principles not to data structure but to reasoning structure. The fiber bundle formalism is not used to encode symmetries of the input domain but to constrain the consistency of sequential inference. This represents a shift from geometric inductive biases about data to geometric inductive biases about computation.

6.2 Consistency in Language Models

Prior work on consistency in language models has primarily taken post-hoc approaches: verification systems that check outputs against themselves or external knowledge bases, self-consistency decoding that samples multiple reasoning paths and aggregates results, or fine-tuning procedures that penalize detected contradictions.

The Control Field HoT differs fundamentally by building consistency into the forward pass. Rather than detecting and filtering inconsistency after generation, we provide continuous architectural pressure toward consistent reasoning during generation. This is both more efficient (no additional inference passes) and more fundamental (the model learns representations conducive to consistency rather than learning to avoid detectably inconsistent outputs).

6.3 Attention Modifications

Many works have proposed modifications to the attention mechanism: sparse attention patterns, linear attention approximations, relative position encodings, and various gating schemes. The Control Field HoT's attention gating is distinguished by being semantically grounded rather than structurally motivated. The gate is not based on position or sparsity patterns but on predicted consistency implications—a content-dependent modulation that reflects the model's learned understanding of reasoning coherence.

7. Broader Implications and Future Directions

7.1 Consistency as Architectural Primitive

The success of the Control Field formulation suggests that consistency should be treated as an architectural primitive rather than an emergent property to be hoped for or enforced post-hoc. Just as attention provides a mechanism for relevance and feedforward networks provide a mechanism for transformation, the control field provides a mechanism for coherence.

This architectural perspective has implications beyond the specific implementation presented here. If consistency can be predicted and accumulated, other structural properties of reasoning might be similarly amenable to learned, low-overhead computation. Causality, counterfactual dependence, or logical necessity might all have "field" formulations that provide architectural inductive biases without prohibitive computational cost.

7.2 Integration with Symbolic Systems

The control field provides an interface between neural and symbolic computation. The accumulated field values are interpretable signals that external systems can monitor and respond to. A symbolic reasoning system could observe rising control field values as an early warning of impending inconsistency, triggering intervention before the neural model generates contradictory output.

This opens possibilities for hybrid neuro-symbolic architectures where the Transformer handles fluent generation while symbolic systems maintain hard constraints. The control field serves as the communication channel, providing continuous feedback about the neural model's confidence in its own consistency.

7.3 Risk-Shaped Control

A limitation of the current formulation is that the holonomy predictor is trained on self-referential signals—it learns to predict inconsistency as measured by its own internal geometry. This creates potential for the model to "game" the metric, learning representations that appear consistent by the geometric measure while failing on downstream tasks.

Future work will explore Risk-Shaped Control Fields, where the holonomy predictor is trained on externally grounded outcomes rather than internal geometry. By training the predictor to anticipate task failure—whether the model will make factual errors, logical mistakes, or other externally verifiable failures—we ground the consistency signal in real-world consequences rather than geometric proxies.

Until externally-grounded training is implemented, the control field should be understood as a learned regularizer that correlates with representational stability, rather than a verified consistency detector. The architecture provides the mechanism for consistency pressure; validating that this pressure translates to improved reasoning outcomes requires the empirical evaluation outlined above.

7.4 Scaling Properties

The current work validates trainability at modest scale. A critical question for future research is how the architecture behaves under scaling—both in model size and dataset size. Does the consistency benefit compound with scale, or does it become a limiting factor? Do larger models require larger fiber dimensions, or does the compressed consistency representation generalize?

We hypothesize that consistency pressure becomes more valuable at scale, where models have sufficient capacity to memorize inconsistent patterns from training data. The control field may provide regularization that prevents such memorization, forcing the model toward generalizable consistent reasoning patterns. Empirical validation of this hypothesis awaits future scaling experiments.

7.5 Probing Fiber Representations

An open question is what structure emerges in the learned fiber space. Do fibers specialize by reasoning type—with distinct regions encoding logical inference, factual recall, and creative generation? Do fiber representations collapse to low-dimensional manifolds, or do they utilize the full capacity of the fiber dimension? Does the geometric inductive bias produce interpretable internal structure, or does the fiber function primarily as a black-box regularizer?

Probing studies examining fiber activations across task types will illuminate these questions. If fibers exhibit interpretable specialization, this would support the geometric intuition that consistency has domain-specific structure. If fibers remain distributed and task-agnostic, the control field may be capturing a more general notion of representational stability. Either outcome has implications for how the architecture should be extended and applied.

8. Conclusion

We have presented the Control Field Holonomy Transformer, an architecture that embeds structural consistency as a trainable property of sequence modeling. By reconceptualizing consistency from measurement to anticipation, we transformed a theoretically elegant but computationally prohibitive geometric framework into a practical architectural component with minimal overhead.

The key contributions are threefold. First, we establish that geometric consistency constraints from differential geometry can be effectively approximated through learned prediction rather than explicit computation. Second, we demonstrate that this approximation maintains the essential inductive bias—pressure toward consistent reasoning—while reducing the holonomy-specific computation from $O(n^2 \cdot d^3)$ to $O(n)$. Third, we provide a complete, trainable implementation that validates these claims empirically under controlled conditions.

The Transformer revolutionized sequence modeling by showing that attention is all you need for learning contextual dependencies. We propose a complementary thesis: consistency is all you need for reliable reasoning. Attention tells the model where to look; the control field tells it what to trust. Together, they form an architecture that can reason coherently at scale.

Geometry can inspire architecture without dominating computation. The insight that consistency has structure—that it is not merely an emergent property to hope for but a computable quantity to optimize—survives translation from pure mathematics to practical neural networks. The control field makes this insight operational, providing a mechanism for consistency that is differentiable, parallelizable, and compatible with the training procedures that have made modern language models possible.

The loop from theory to implementation is closed. Consistency is no longer just measured—it is anticipated and trained as an architectural bias. The path forward is clear: rigorous evaluation on reasoning benchmarks, scaling studies to validate behavior at production sizes, and integration with symbolic systems that can leverage the interpretable consistency signals. The foundation is laid. What remains is to build upon it.

References

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.

[2] Napolitano, L. M. (2025). The Holonomy Crusher: Geometric Consistency Enforcement in Neural Reasoning Systems. Zenodo. https://doi.org/10.5281/zenodo.14609863

[3] Napolitano, L. M. (2025). From Explicit Holonomy to Latent Control Fields: An O(n) Reformulation of Geometric Consistency. Zenodo. https://doi.org/10.5281/zenodo.14615992

[4] Napolitano, L. M. (2025). The Holonomy Transformer: Technical Report on Training Dynamics and Extended Applications. Zenodo. https://doi.org/10.5281/zenodo.14612551

[5] Napolitano, L. M. (2025). Risk-Shaped Control Fields: Externally-Grounded Anticipatory Consistency for Language Model Reasoning. Zenodo. https://doi.org/10.5281/zenodo.14619088

[6] Napolitano, L. M. (2025). The Symbolic Control Runtime: A Systems Architecture for Consistent Language Model Reasoning. Zenodo. https://doi.org/10.5281/zenodo.14627981

[7] Bronstein, M. M., Bruna, J., Cohen, T., & Veličković, P. (2021). Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. arXiv preprint arXiv:2104.13478.

[8] Cohen, T., & Welling, M. (2016). Group equivariant convolutional networks. International Conference on Machine Learning.

[9] Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., & Zhou, D. (2022). Self-consistency improves chain of thought reasoning in language models. arXiv preprint arXiv:2203.11171.

[10] Rae, J. W., Potapenko, A., Jayakumar, S. M., Hillier, C., & Lillicrap, T. P. (2020). Compressive Transformers for long-range sequence modelling. International Conference on Learning Representations.

Appendix A: Implementation Details

Complete implementation code is available in the supplementary materials. Here we highlight key implementation decisions.

A.1 Hyperparameters

Default configuration: d_model = 64, d_fiber = 8, d_control = 16, n_heads = 2, n_layers = 2, momentum $\alpha = 0.9$, attention gate scale $\lambda = 1.0$, curvature gate scale $\lambda\_\kappa = 0.1$, holonomy loss weight = 0.001, curvature loss weight = 0.0001.

A.2 Vectorized Accumulation

The control field recurrence $h_t = \alpha \cdot h_{t-1} + (1 - \alpha) \cdot \Delta h_t$ can be computed in closed form as a causal convolution with kernel $[(1-\alpha)\alpha^{n-1}, (1-\alpha)\alpha^{n-2}, ..., (1-\alpha)]$. This enables full GPU parallelization via torch.nn.functional.conv1d with appropriate padding. Both sequential and vectorized implementations are provided; the sequential version serves pedagogical purposes while the vectorized version is recommended for production deployment.

A.3 Numerical Stability

Attention gating uses log-space addition (scores + log(gate + ε)) rather than multiplicative gating to maintain numerical stability across the full range of gate values. The small $\varepsilon = 10^{-8}$ prevents log(0) while remaining negligible in the softmax computation.

A.4 Weight Initialization

Standard Xavier/Glorot initialization is used for all linear layers. The holonomy predictor's final layer uses a smaller initialization scale (0.01) to ensure the control field starts near zero, allowing the model to learn consistency patterns gradually rather than imposing strong gating from the start of training.

Appendix B: Mathematical Derivations

B.1 Closed-Form Control Field

Starting from the recurrence $h_t = \alpha \cdot h_{t-1} + (1 - \alpha) \cdot \Delta h_t$ with $h_0 = (1 - \alpha) \cdot \Delta h_0$:

$$h_1 = \alpha(1-\alpha)\Delta h_0 + (1-\alpha)\Delta h_1$$

$$h_2 = \alpha^2(1-\alpha)\Delta h_0 + \alpha(1-\alpha)\Delta h_1 + (1-\alpha)\Delta h_2$$

$$h_t = (1-\alpha) \sum_{i=0}^{t} \alpha^{t-i} \cdot \Delta h_i$$

This is a discrete convolution of the Δh sequence with an exponentially decaying kernel, computable in O(n log n) via FFT or O(n) via direct convolution for moderate sequence lengths.

B.2 Gradient of Accumulated Field

For backpropagation, we require $\partial h_t/\partial \Delta h_i$ for $i \leq t$. From the closed form: $\partial h_t/\partial \Delta h_i = (1-\alpha)\alpha^{t-}{}_{i}$. This decays exponentially into the past, automatically bounding gradient magnitudes and preventing explosion through long sequences.

B.3 Effective Memory Horizon

The half-life of the control field—the number of tokens after which a holonomy prediction's influence decays to 50%—is given by t½ = $-1 / \log_2(\alpha)$. For $\alpha = 0.9$, t½ ≈ 6.6 tokens. For $\alpha = 0.95$, t½ ≈ 13.5 tokens. This provides a principled way to tune the temporal horizon of consistency pressure based on the expected reasoning span of the target domain.