Geometric Exactness and the Training–Exposure Distinction

Why Architectures Encoding Proven Identities
Are Degraded by Gradient Descent

Léon Fernando Vlegels
Independent Researcher

2025-12

Contents

Abstract

We formalize a distinction between two modes of computation: *training*, in which free parameters are adjusted by gradient descent to minimize a loss function, and *exposure*, in which a fixed-architecture system receives inputs that select trajectories through an already-correct geometric structure. We prove that when a computational graph exactly implements a mathematical identity (with zero approximation error at its design point), any gradient step generically increases error. The architecture sits at an isolated global minimum of the loss landscape, and gradient descent cannot improve it. We apply this framework to the identities established in the geometric Theory of Everything: the Hopf projection identity $v = \cos(2\beta)$, the Lorentz lapse $m = \sqrt{1 - v^2}$, the axis-resolved map, and the measure-reconciliation constant $\Omega_0 = \pi^3/4$. These identities define a class of *geometrically exact* computational architectures for which the correct operating mode is exposure, not training. We establish necessary and sufficient conditions for a computational graph to belong to this class and discuss implications for the relationship between mathematical structure and machine learning.

1 Introduction

There are two ways a computational system can come to compute a function correctly. It can search for the function, adjusting internal parameters until its outputs match observed data; or it can be built as the function, with every internal constant fixed by construction. The first way is the default culture of modern machine learning. The second is the natural mode for any system whose target is a proven mathematical identity. A trained system carries residual error that optimization can in principle reduce; an identity-implementing system carries none at its design point, so the same machinery, applied to it, can only move it away from correctness. This paper makes the distinction precise and draws the operational consequence: a system that implements an identity should be operated by *exposure*, by feeding it inputs and reading its outputs, with nothing inside it ever adjusted.

1.1 The Standard Paradigm

Modern machine learning operates by parameterizing a function class $\{f_\theta : \theta \in \Theta\}$ and optimizing

$$\theta^* = \arg\min_{\theta \in \Theta} \mathcal{L}(\theta) := \frac{1}{N} \sum_{i=1}^{N} \ell\big(f_\theta(x_i),\, y_i\big) \tag{1}$$

via gradient descent: $\theta_{t+1} = \theta_t - \eta \nabla_\theta \mathcal{L}(\theta_t)$. This assumes that the correct parameters are *unknown* and must be discovered from data.

The assumption is usually justified. Where the target function exists only as a statistical regularity in a corpus, search is the only available strategy, and the apparatus that has grown around equation (1), train–test splits, regularization, learning-rate schedules, early stopping, exists to manage the consequences of that ignorance. Every element of it presupposes a gap between f_θ and the target, to be closed by adjustment.

1.2 The Geometrically Exact Case

Consider instead a system where the target function f^* is known *exactly*: not as an approximation, not as an asymptotic limit, but as a proven mathematical identity. If the architecture can implement f^* with specific parameter values θ^* such that $f_{\theta^*} \equiv f^*$, then:

(i) The loss at θ^* is exactly zero: $\mathcal{L}(\theta^*) = 0$.

(ii) Any perturbation $\theta^* + \delta$ yields $\mathcal{L}(\theta^* + \delta) \geq 0$, with equality only if $f_{\theta^*+\delta} \equiv f^*$.

(iii) Gradient descent from θ^* cannot reduce loss further.

In this case the training apparatus loses its purpose. There is no gap to close: the parameters are known quantities to be written down, not unknown quantities to be estimated. And there is no generalization question in the statistical sense: an identity holds for every input by the meaning of the word identity, not with high probability over a distribution.

The question is: under what conditions is this minimum *isolated*, so that gradient descent from a nearby initialization cannot reliably reach θ^*, while direct construction can? Isolation is what turns the observation above into a design constraint: if the zero-loss point is isolated, every displacement costs accuracy, and any procedure that displaces parameters degrades the system.

1.3 Connection to the Geometric TOE

Paper 27 [3] established a collection of exact identities governing dual-observer dynamics on S^3:

$$v(q) = \cos(2\beta), \tag{2}$$

$$m(q) = \sqrt{1 - v(q)^2}, \tag{3}$$

$$v = n_{\parallel}^2 + (1 - n_{\parallel}^2)\cos\eta, \tag{4}$$

$$\Omega_0 = \pi^3/4. \tag{5}$$

These are not regression targets. They are theorems. A computational system that implements them does not benefit from optimization; it benefits from correct *construction*. This paper makes that claim precise.

The stakes for the corpus are direct. Every constant in the chain of derivations is fixed by geometry: $\alpha^{-1} = 4\pi^3 + \pi^2 + \pi$ from the (B^4, S^3) volume structure [2], $\Omega_0 = \pi^3/4$ from measure reconciliation, the Hopf coefficients from the fibration itself. Operated as a trainable model, each derivation would be quietly converted back into a fitting problem, and the central claim of the corpus, that the constants are derived rather than fitted, would be undone at the implementation layer.

1.4 Outline

Section 2 fixes the formal vocabulary and proves the zero-loss theorem. Section 3 proves the isolation theorem, records two theorem-scope caveats explicitly, and works through a closed-form scalar example. Section 4 defines exposure as an operating mode and develops the resonance analogy as a worked case. Section 5 applies the framework to the TOE identity catalog, Section 6 states when each paradigm is appropriate, and Section 7 draws the implications.

2 Formal Framework

2.1 Computational Graphs and Parameter Spaces

The definitions below are deliberately minimal; nothing in the argument depends on the graph being a neural network.

Definition 2.1 (Parameterized Computational Graph)**.** A parameterized computational graph is a tuple $\mathcal{G} = (V, E, \theta)$ where V is a set of nodes (operations), E is a directed acyclic graph of data flow, and $\theta \in \Theta \subseteq \mathbb{R}^p$ is a vector of adjustable parameters. The graph computes a function $f_\theta : \mathcal{X} \to \mathcal{Y}$.

Definition 2.2 (Geometric Exactness)**.** A computational graph $\mathcal{G}$ is *geometrically exact* for a target function f^* if there exists $\theta^* \in \Theta$ such that

$$f_{\theta^*}(x) = f^*(x) \qquad \forall\, x \in \mathcal{X}. \tag{6}$$

We call θ^* the *geometric parameter*.

The qualifier *geometric* signals where such exactness comes from. Universal-approximation results guarantee approximation to any tolerance; they do not produce a θ^* at which the error is identically zero. Exact implementation is available only when the target can be written directly into the graph: a polynomial with its own coefficients, an algebraic operation as that operation, a constant as that constant. The identities of Section 5 are of exactly this kind.

Definition 2.3 (Loss Landscape)**.** Given a distribution μ on $\mathcal{X} \times \mathcal{Y}$ with $y = f^*(x)$, the population loss is

$$\mathcal{L}(\theta) = \mathbb{E}_{(x,y)\sim\mu}\big[\|f_\theta(x) - f^*(x)\|^2\big]. \tag{7}$$

Note that the labels are noiseless: $y = f^*(x)$ exactly, because the target is an identity, not a measurement. The loss floor is exactly zero, and it is attained.

2.2 The Zero-Loss Theorem

Theorem 2.4 (Zero Loss at Geometric Parameter)**.** *If $\mathcal{G}$ is geometrically exact for f^* with geometric parameter θ^*, then $\mathcal{L}(\theta^*) = 0$ and $\nabla_\theta \mathcal{L}(\theta^*) = 0$. That is, θ^* is a critical point of $\mathcal{L}$.*

Proof. By Definition 2.2, $f_{\theta^*}(x) = f^*(x)$ for all x, so $\|f_{\theta^*}(x) - f^*(x)\|^2 = 0$ for all x. Therefore $\mathcal{L}(\theta^*) = 0$. Since $\mathcal{L} \geq 0$ everywhere and $\mathcal{L}(\theta^*) = 0$, the point θ^* is a global minimum, hence $\nabla_\theta \mathcal{L}(\theta^*) = 0$. □

The proof is elementary by design, and it is worth noticing what it does not use: it does not differentiate through the graph, and it does not depend on μ at all, since the integrand vanishes pointwise. Everything distribution-sensitive enters later, in the isolation question, where one must show that points *near* θ^* have strictly positive loss; there the choice of μ matters, and the remarks in Section 3 record exactly how.

3 Isolation and Training Incompatibility

3.1 When Is the Minimum Isolated?

Zero loss at θ^* says training cannot improve the system. Isolation says training can damage it: any process that moves the parameters moves them uphill.

Definition 3.1 (Isolated Geometric Minimum)**.** The geometric parameter θ^* is *isolated* if there exists $\epsilon > 0$ such that for all $\delta \in \mathbb{R}^p$ with $0 < \|\delta\| < \epsilon$,

$$\mathcal{L}(\theta^* + \delta) > 0. \tag{8}$$

The obstruction to isolation is parameter redundancy: if distinct parameter vectors near θ^* compute the same function, the loss is zero along a continuum. The hypothesis that rules this out is local injectivity of the parameter-to-function map.

Theorem 3.2 (Isolation from Rigidity)**.** *Let $\mathcal{G}$ be geometrically exact for f^* with geometric parameter θ^*. Suppose f^* is not identically zero and the map $\theta \mapsto f_\theta$ is locally injective at θ^* (i.e., no redundant parameters near θ^*). Then θ^* is isolated.*

Proof. Local injectivity means: for all δ with $\|\delta\|$ sufficiently small and $\delta \neq 0$, there exists $x \in \mathcal{X}$ such that $f_{\theta^*+\delta}(x) \neq f_{\theta^*}(x) = f^*(x)$. Therefore $\|f_{\theta^*+\delta}(x) - f^*(x)\|^2 > 0$ for that x, giving $\mathcal{L}(\theta^* + \delta) > 0$. □

The argument is a two-step transfer: local injectivity supplies one input x at which the perturbed function disagrees with the target, and the loss is then claimed positive because the integrand is positive at that x. The second step is where an unstated hypothesis enters: one point of disagreement forces positive *population* loss only if μ assigns weight near that point. The following remarks record this and a second scope limit; the honest statement of the theorem is the theorem together with them.

Remark 3.3 (TBS). The proof of Theorem 3.2 does not establish that $\mathcal{L}(\theta^* + \delta) > 0$ for an *arbitrary* data distribution μ. Local injectivity guarantees a single x where the perturbation creates a nonzero residual, but if μ does not charge that point (e.g. if μ is supported on a lower-dimensional set), the population loss may remain zero despite the local non-injectivity of $\theta^* + \delta$. A full proof requires μ to have full topological support or an equivalent regularity condition not stated here.

Status. Recorded in the registry as P029_2_c (scope), confirmed by A295. The theorem requires a full-support hypothesis on the data distribution that the proof does not state; this is a theorem-hypothesis caveat, not a defect in the architecture. Corpus use of the training/exposure distinction is unaffected. Ledger: Paper 40.

A concrete instance of the gap: for $f_\theta(x) = \theta x$ on $[0, 1]$ with target $f^*(x) = x$, the map $\theta \mapsto f_\theta$ is injective, yet if μ is a point mass at $x = 0$ then $\mathcal{L}(\theta) = 0$ for every θ. Full support restores the conclusion, as the worked example below shows.

Remark 3.4 (TBS). The isolation of θ^* (Definition 3.1) establishes only that no *nearby* points achieve zero loss, not that θ^* is the *unique* global zero-loss point. Other isolated minima elsewhere in Θ are not precluded. Uniqueness of the global minimum would require a global convexity argument or a proof that $\mathcal{L}(\theta) = 0 \Rightarrow \theta = \theta^*$, neither of which is provided.

Status. Recorded in the registry as P029_3_c (scope), confirmed by A295. The isolation result is local; global uniqueness of the zero-loss point is a stronger statement that the paper does not claim to need. Corpus use is unaffected. Ledger: Paper 40.

This second limit is likewise illustrated by a one-line model: for $f_\theta(x) = (\theta - 2)^2 x$ with target $f^*(x) = x$, the loss vanishes at $\theta = 1$ and at $\theta = 3$; each zero is locally isolated, neither globally unique. Operational use does not require global uniqueness: the system is constructed at a specific θ^*, and the relevant question, whether motion away from that point costs accuracy, is the local statement.

Corollary 3.5 (Training Incompatibility)**.** *At an isolated geometric minimum, any gradient step from θ^* either:*

(a) remains at θ^* (zero gradient), or

(b) moves to $\theta^* + \delta$ with $\mathcal{L}(\theta^* + \delta) > 0$.

Gradient descent from θ^* cannot improve performance. If stochastic perturbation or regularization displaces θ from θ^*, subsequent gradient steps may or may not return to θ^* depending on the basin geometry, but they cannot find a better *point.*

The corollary is the operational content of the paper. Exact gradient descent at θ^* is a fixed point, but practical training is never exact gradient descent: minibatch sampling, finite precision, and regularizers all inject displacements, and at an isolated minimum every displacement has strictly positive cost. A training loop wrapped around a geometrically exact graph is a mechanism whose only possible effects are nothing or harm.

3.2 A Scalar Worked Example

Every quantity in the framework can be computed in closed form for the simplest nontrivial model. Take $\mathcal{X} = [-1, 1]$ with μ uniform, target $f^*(x) = x$, and the one-parameter graph $f_\theta(x) = \theta x$. The population loss is

$$\mathcal{L}(\theta) = \mathbb{E}\big[(\theta x - x)^2\big] = (\theta - 1)^2\, \mathbb{E}[x^2] = \frac{(\theta - 1)^2}{3},$$

with geometric parameter $\theta^* = 1$. Then $\mathcal{L}(1) = 0$ and $\mathcal{L}'(\theta) = 2(\theta - 1)/3$ vanishes at θ^* (Theorem 2.4); the second derivative $\mathcal{L}'' = 2/3 > 0$ makes the minimum strict (Theorem 3.2); a nearby perturbation gives positive loss, $\mathcal{L}(1.1) = 0.01/3 \approx 0.0033$; and an exact gradient step from θ^* at any learning rate returns θ^*, the fixed-point behavior of Corollary 3.5. The same model with μ a point mass exhibits the failure recorded in Remark 3.3; the contrast between the two choices of μ is the content of the full-support hypothesis.

3.3 Hessian Structure

Proposition 3.6 (Positive-Definite Hessian)**.** *Under the conditions of Theorem 3.2, if f_θ is twice differentiable in θ and μ has full support, then the Hessian*

$$H_{ij} = \frac{\partial^2 \mathcal{L}}{\partial \theta_i \partial \theta_j}\bigg|_{\theta^*} = 2\,\mathbb{E}\left[\sum_k \frac{\partial f_k}{\partial \theta_i}\frac{\partial f_k}{\partial \theta_j}\right]_{\theta^*} \tag{9}$$

is positive semi-definite, and is strictly positive-definite if the Jacobian $\partial f/\partial\theta|_{\theta^*}$ has full column rank.

Proof. At the exact minimum, the second-order expansion of $\mathcal{L}$ reduces to the Gauss–Newton approximation (the residual term vanishes since $f_{\theta^*} = f^*$). The result follows from the rank condition on the Jacobian. □

Remark 3.7. A strictly positive-definite Hessian means θ^* is a *strict local minimum with quadratic basin*. Gradient descent from nearby points will converge toward θ^*, but from distant initializations there is no guarantee: the loss landscape may have other critical points. Direct construction bypasses this landscape entirely.

The Gauss–Newton reduction in the proof is special to the exact case: the residual-weighted curvature term, the usual source of saddle points in trained networks, is identically zero, and only the positive semi-definite Jacobian term survives. The landscape around θ^* is a clean quadratic bowl. In the scalar example above the formula gives $H = 2\,\mathbb{E}[x^2] = 2/3$, matching the direct computation.

4 The Exposure Paradigm

4.1 Definition

Definition 4.1 (Exposure)**.** Let $\mathcal{G}$ be a geometrically exact computational graph with fixed parameter θ^*. *Exposure* is the process of providing inputs $x \in \mathcal{X}$ and reading outputs $f_{\theta^*}(x) \in \mathcal{Y}$, with no modification of θ^*.

In the exposure paradigm:

(i) Parameters are *constructed*, not learned.

(ii) Inputs set *initial conditions*, not training signals.

(iii) The forward pass *computes* the target identity, not an approximation to it.

(iv) No loss function, no backward pass, no weight update.

Operationally, an exposure run has a different shape from a training run. There is no dataset to assemble: a single input is a complete unit of work, and its output is the exact value of the identity at that input. There is no held-out set, because the function computed on unseen inputs is the same function, by construction. What replaces all of this is verifying that the construction is correct. A wrong output from an exposure-mode system is not a fitting error to be reduced by more data; it is a construction error, repaired by reading the derivation, not by running an optimizer. Wrongness in exposure mode is a bug, not a residual.

4.2 Formal Comparison

Property	**Training**	**Exposure**
Parameters	Unknown, discovered	Known, constructed
Input role	Training signal	Initial condition
Output role	Prediction (approximate)	Computation (exact)
Error source	Parameter mismatch	None (at design point)
Gradient	Informative	Zero (at θ^*)
Data requirement	Large corpus	Single instance suffices
Generalization	Statistical	Guaranteed (identity holds $\forall\, x$)

Each row of the table is the same underlying fact viewed through a different operational lens; the rows are not independent advantages but stand or fall together with geometric exactness itself.

4.3 The Resonance Analogy (Formalized)

Consider a physical oscillator with resonant frequency ω_0 determined by its geometry (mass, stiffness). Exposure to a driving signal at frequency ω produces response amplitude

$$A(\omega) = \frac{F_0}{\sqrt{(\omega_0^2 - \omega^2)^2 + \gamma^2\omega^2}}. \tag{10}$$

The resonance peak at $\omega = \omega_0$ is *not learned*; it is a consequence of the oscillator's architecture. "Training" the oscillator (modifying its mass or stiffness) would shift the resonance away from the correct frequency. Exposure (driving it at ω_0) activates the pre-existing response.

As a worked case, set $\omega = \omega_0$ in equation (10): the first term of the denominator collapses, leaving $A(\omega_0) = F_0/(\gamma\,\omega_0)$. For $F_0 = 1$, $\omega_0 = 2$, $\gamma = 1$ this gives $A(\omega_0) = 1/2$ exactly, fixed by the architecture's constants alone, with nothing fitted. The oscillator did not discover ω_0; it *is* ω_0, in the same sense that a geometrically exact graph is the identity it computes.

Remark 4.2 (TBS). The stated formula for the damped resonance peak frequency is incorrect; the correct peak of the amplitude response occurs at $\omega_{\text{peak}} = \sqrt{\omega_0^2 - \gamma^2/2}$, not $\sqrt{\omega_0^2 - \gamma^2}$. The derivation must be corrected accordingly.

Status. Retired. The registry records P029_1 as closed by A278, which supplied the corrected peak formula. The remark above stands as the record of the original gap. Ledger: Paper 40.

The corrected peak location, $\omega_{\text{peak}} = \sqrt{\omega_0^2 - \gamma^2/2}$, sits slightly below ω_0 when $\gamma > 0$ (about 1.8708 at the sample values above). The qualitative point is unchanged: the response curve, peak included, is determined entirely by the oscillator's constants, and no part of it is learned.

A geometrically exact computational graph is the discrete analogue: its "resonant frequencies" are the mathematical identities it implements, and exposure activates them without modification.

5 Application to the TOE Identities

5.1 Catalog of Exact Identities

Paper 27 [3] established the following identities, each of which defines a geometrically exact computational subgraph:

(1) **Hopf observable.** $v(a,b,c,d) = (a^2+b^2) - (c^2+d^2)$ for $(a,b,c,d) \in S^3$. This is a polynomial, exactly implementable with fixed weights (coefficients $+1, +1, -1, -1$).

(2) **Standing-wave identity.** $v = \cos(2\beta)$ in Hopf coordinates. Implemented by the coordinate transformation layer.

(3) **Lorentz lapse.** $m = \sqrt{1-v^2}$. A fixed nonlinear activation; no parameters.

(4) **Quaternion evolution.** $\dot{q} = \frac{1}{2}\Omega q$. A bilinear operation with no free parameters (the $\frac{1}{2}$ is structural).

(5) **Relational quaternion.** $q_{\text{rel}} = q^{(-)}\overline{q^{(+)}}$. Quaternion multiplication and conjugation; no free parameters.

(6) **Axis-resolved identity.** $v = n_\parallel^2 + (1-n_\parallel^2)\cos\eta$. Follows from the log map; no free parameters.

(7) **Measure constant.** $\Omega_0 = \pi^3/4$. A single fixed scalar, derived from $V_6/V_3 = \pi^2/8$.

(8) **Monad closure.** $\Omega_{\text{monad}} = 4\pi^3 + \pi^2 + \pi = \alpha^{-1}$. A fixed scalar from the density integral.

Each entry satisfies Definition 2.2 for an elementary reason. The Hopf observable is a quadratic polynomial whose coefficients are written into the graph as themselves, evaluating to $\cos(2\beta)$ identically in Hopf coordinates. The lapse is a single fixed nonlinearity; at $v = 0.6$ it returns $m = 0.8$ exactly. The evolution and relational maps are quaternion algebra, whose only constant is the structural $\frac{1}{2}$. The two scalars are derived numbers: $\pi^3/4$ arises as $2\pi\cdot(\pi^2/8)$ from the volume-ratio normalization, and $4\pi^3 + \pi^2 + \pi$ evaluates to $137.0363\ldots$, the fine-structure value derived in [2]. In every case the geometric parameter is not the solution to an optimization problem; it is a line in a derivation.

5.2 Parameter Count

Proposition 5.1 (Zero Free Parameters)**.** *The complete computational pipeline*

$$(\Omega^{(+)}, \Omega^{(-)}, q_0^{(+)}, q_0^{(-)}) \xrightarrow{\text{evolve}} q_{\text{rel}}(t) \xrightarrow{\text{Hopf}} v(t) \xrightarrow{\sqrt{1-(\cdot)^2}} m(t)$$

contains zero free parameters. *Every constant ($\frac{1}{2}$, $\pi^3/4$, α^{-1}, the Hopf coefficients ± 1) is either structural or derived. The only inputs are the initial conditions and generator schedules $\Omega^{(\pm)}(t)$.*

The final sentence of the proposition carries the operational division of labor. The pipeline's interior is parameter-free, but its boundary is not empty: the initial conditions and generator schedules are genuine inputs, supplied from outside on every run. This is not a loophole in the zero-parameter claim; it is what exposure means. The degrees of freedom of an exposure-mode system live entirely in its inputs, where they select trajectories, not in its architecture, where they would corrupt identities.

5.3 Training Would Corrupt

Corollary 5.2 (Training Degradation)**.** *If any of the fixed constants in the pipeline (the Hopf coefficients, the factor $\frac{1}{2}$ in quaternion evolution, the constant $\Omega_0 = \pi^3/4$) are promoted to trainable parameters and gradient descent is applied with any nonzero learning rate from the geometric values, the loss $\mathcal{L}$ can only increase or remain zero. In particular, stochastic gradient noise will generically displace parameters from their exact values, increasing error.*

This was verified empirically in Paper 27: multiplying the Θ-rate by α or α^{-1} (i.e., perturbing Ω_0 away from $\pi^3/4$) worsened the SR lapse agreement. The Ω_0 sweep confirmed that the geometric value sits at the minimum of the RMSE surface.

The sweep is the empirical face of Theorem 3.2: a one-dimensional slice through the loss landscape along the Ω_0 direction, with the derived value at the bottom of the basin and error rising on both sides. The catalog and its empirical confirmation belong to Paper 27 and its antecedents; what this paper adds is the landscape-level reading, that a basin minimum at the derived value is exactly what an isolated geometric minimum looks like from the outside.

6 General Conditions for the Distinction

The distinction is not a blanket argument against training; it is a classification of problems, and the conditions below make the boundary explicit.

6.1 When Training Is Appropriate

Training is the correct paradigm when:

(i) The target function f^* is *unknown* or only accessible through samples.

(ii) The architecture cannot implement f^* exactly (approximation is necessary).

(iii) The relationship between parameters and function is *many-to-one* (symmetries in parameter space), so gradient landscape exploration is needed to find good representatives.

These are the standard conditions of statistical learning, and most practical prediction problems satisfy all three; nothing in this paper disputes the fitting paradigm on its own ground.

6.2 When Exposure Is Appropriate

Exposure is the correct paradigm when:

(i) The target function f^* is known exactly as a mathematical identity.

(ii) The architecture can implement f^* with specific, constructible parameter values.

(iii) The parameter-to-function map is locally injective near θ^* (Theorem 3.2).

(iv) The identity holds universally ($\forall\, x \in \mathcal{X}$), not just on a training distribution.

The first condition is the demanding one, and it is mathematical, not engineering: it is met by proving a theorem, not by collecting data. One does not drift into the exposure class by accumulating accuracy; one enters it by derivation.

Theorem 6.1 (Dichotomy)**.** *For a geometrically exact architecture satisfying the conditions of Theorem 3.2:*

(a) Training from θ^* is wasteful (zero gradient, no improvement possible).

(b) Training from $\theta \neq \theta^*$ may converge to θ^* if θ is in the basin of attraction, but is less efficient than direct construction.

(c) Training with regularization or noise generically prevents *convergence to θ^* by penalizing the exact values.*

(d) Exposure at θ^* yields exact computation with zero error on every input.

The four clauses inherit the hypotheses of Theorem 3.2, including the support condition of Remark 3.3 and the local reading of Remark 3.4. Clause (b) concedes the strongest case for training: gradient descent initialized inside the quadratic basin will flow toward the geometric values. But converging toward a number one already possesses in closed form is not a method; it is a detour. For this class of architectures, construction strictly dominates search.

7 Implications

7.1 For Computational Architecture

The training–exposure distinction implies that hybrid systems are natural: a system may contain both *exact subgraphs* (geometric layers with fixed parameters) and *learned subgraphs* (trainable layers that handle unknown functions). The key design principle is: *never train what you can construct.*

For the TOE pipeline, the geometric layers (quaternion algebra, Hopf projection, measure constants) should be hardcoded. Only the *input interface*, the mapping from raw physical data to initial conditions $q_0^{(\pm)}$ and generator schedules $\Omega^{(\pm)}(t)$, is a candidate for learning, since that mapping depends on the specific measurement apparatus and is not determined by the geometry alone.

The boundary between the two regimes must sit at the input interface and nowhere deeper: gradients from a trained interface must never propagate into the exact layers, because by Corollary 5.2 any update they induce there is degradation. The exact subgraph is frozen not as an engineering convenience but as a correctness condition.

7.2 For the Corpus

Within the volume, this paper carries a methodological load. The corpus derives its constants; an implementation of the corpus must therefore never re-fit them, or the derivations become decoration on what is, at the implementation layer, just another fitted model. The training–exposure distinction is the warrant for the way the corpus is operated: every operator hardcoded from a proven identity, zero trainable parameters in the geometric core, and each run consisting of a single input selecting a trajectory through the fixed structure, with nothing adjusted on the way through. When a computation disagrees with expectation, the response is to audit the construction against the derivation, never to relax a constant and re-optimize.

7.3 For the Philosophy of Knowledge

The distinction maps onto a classical epistemological divide:

(i) **Training** corresponds to *empiricism*: knowledge is extracted from experience by adjusting internal states.

(ii) **Exposure** corresponds to *rationalism* (or more precisely, geometric nativism): the structure is already present and experience merely activates it.

The geometric TOE suggests that the fundamental laws of physics belong to the exposure category: they are not learned from data but are the architecture of reality itself. Any system that correctly instantiates that architecture computes physics exactly, requiring only initial conditions (exposure) to produce specific predictions.

7.4 For the Relationship Between Mathematics and Physics

If physical law consists of exact geometric identities (as the TOE proposes), then the relationship between mathematics and physics is not one of *description* (mathematics models physics) but of *identity* (physics is geometry, computed). The training–exposure distinction formalizes this: you do not approximate an identity; you implement it.

On this reading, measurement itself is exposure: an experiment supplies an initial condition and reads the trajectory the fixed structure assigns to it. What physicists update is their description, which lives on the training side of the divide until a derivation moves a relation across, after which it is implemented rather than estimated.

8 Conclusion

We have established a formal distinction between training and exposure as computational paradigms. The key results are:

(i) A geometrically exact computational graph sits at an isolated global minimum of the loss landscape (Theorems 2.4–3.2).

(ii) Gradient descent from the geometric parameter is either trivial (zero gradient) or destructive (any displacement increases error).

(iii) The TOE identities from Paper 27 define a zero-free-parameter pipeline that is degraded by training (Proposition 5.1, Corollary 5.2).

(iv) The correct operating mode for such architectures is exposure: providing inputs and reading outputs, with no parameter modification.

(v) The distinction generalizes: any computational system containing both proven-identity subgraphs and unknown-function subgraphs should freeze the former and train only the latter.

The scope of these results is kept deliberately visible: the isolation theorem requires a full-support condition its proof does not state (Remark 3.3), and isolation is local, not a global uniqueness claim (Remark 3.4). Neither caveat touches the operational use of the distinction, which concerns motion away from a constructed θ^* under full-support evaluation.

A mathematical identity is not a hypothesis to be tested. It is a structure to be instantiated.

References

[1] L. F. Vlegels, *The Monad Rosetta Map: From Void to Structure*, This volume (2025).

[2] L. F. Vlegels, *The Perfect Stable Sphere: Deriving α^{-1} from (B^4, S^3) Geometry*, This volume (2025).

[3] L. F. Vlegels, *Universal Wave Geometry: Emergent Lorentz Structure from Dual-Observer Phase Dynamics on S^3*, This volume (2025).

[4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press (2016).

[5] M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković, "Geometric deep learning: grids, groups, graphs, geodesics, and gauges," *arXiv:2104.13478* (2021).

[6] T. S. Cohen and M. Welling, "Group equivariant convolutional networks," *Proc. ICML* (2016), 2990–2999.