OCTA: A Structured Projection Architecture for T-Selection-Based Artificial General Intelligence

OCTA Research
(Dated: December 2025)

We present OCTA, a structured projection architecture for artificial general intelligence explicitly designed as a T-closure engine. Building on the T-Selection Principle introduced in Ref. [1], which characterizes reality and intelligence as fixed points of minimal contradiction residue across projections, scales, and descriptions, OCTA instantiates a family of learned projection operators over world and model states. The architecture decomposes its objective into (i) world-model accuracy, (ii) internal self-consistency, and (iii) multi-agent consensus, each expressed as Kullback–Leibler divergence terms over predictive distributions. We describe the lobe-based design (perceptual, model, introspective, and social), the multi-term loss, the training procedure, and evaluation metrics oriented around Δ-reduction efficiency rather than task scores alone. OCTA thus operationalizes T-selection as a concrete optimization principle for AGI.

I. INTRODUCTION

The T-Selection Principle, introduced in Ref. [1], proposes that persistent dynamical structures—including physical laws, biological organisms, and intelligent agents—are characterized by *tautological closure*: they maintain self-consistency across their own projections, scales, and recursive transformations. This is formalized via a nonnegative contradiction-residue functional $\Delta(X)$ on a configuration space $\mathcal{C}$ and a closure functional

$$T(X) = \exp\bigl(-\Delta(X)\bigr), \tag{1}$$

with T-selection dynamics given by gradient flow on Δ.

Within this framework, intelligence is interpreted as the capacity to reduce Δ with respect to an external world. Modern machine learning architectures, however, are typically optimized for task-specific loss functions without explicit regard for internal consistency or multi-view closure. This leads to systems that perform well on benchmarks but lack principled guarantees of self-consistency or robustness under distribution shift.

In this work, we define OCTA (Orthogonal Consistency and T-Attractor) as a concrete AGI architecture explicitly designed to minimize contradiction residue across a structured family of projections. OCTA treats world and model as elements of a joint configuration space, instantiates distinct projection families (perceptual, model, introspective, social), and constructs a multi-term loss that mirrors the decomposition of Δ in Ref. [1]. The result is an architecture whose training objective is not merely task accuracy but explicit T-closure.

Our contributions are:

- a formalization of OCTA as a structured projection family over world and model states;
- a lobe-based architecture with distinct but coupled modules for perception, modeling, introspection, and social coordination;
- a multi-term loss function decomposed into world-model, self-consistency, and consensus residues, expressed via Kullback–Leibler divergences;
- training procedures and evaluation metrics framed in terms of Δ-reduction efficiency;
- a conceptual bridge from T-selection theory to implementable AGI systems.

II. BACKGROUND: T-SELECTION AND CONTRADICTION RESIDUE

We briefly summarize the elements of the T-Selection Principle relevant to OCTA; see Ref. [1] for full details.

Let $\mathcal{C}$ be a configuration space of dynamical structures. For each $X \in \mathcal{C}$, a family of projections $\{\Pi_i\}_{i\in I}$ maps X to observational or reduced descriptions O_i with induced distributions $p(O_i \mid X)$.

The total contradiction residue is decomposed as

$$\Delta(X) = \Delta_{\mathrm{proj}}(X) + \Delta_{\mathrm{rec}}(X) + \Delta_{\mathrm{mdl}}(X), \tag{2}$$

where:

- Δ_{proj} measures cross-projection inconsistency via Kullback–Leibler divergence [2, 3],
- Δ_{rec} measures recursive (scale) inconsistency via RG-like coarse-graining operators [4],
- Δ_{mdl} measures descriptive redundancy via minimum-description-length (MDL) [6].

The closure functional is

$$T(X) = \exp\bigl(-\Delta(X)\bigr), \tag{3}$$

and T-selection dynamics are given by

$$\frac{dX}{dt} = -\nabla_X \Delta(X). \tag{4}$$

Under mild regularity conditions, Δ is a Lyapunov functional and T-selected fixed points are locally asymptotically stable [1].

In the context of intelligence, we consider a joint configuration (X, M) where X is the world and M is an internal model state. T-selection then becomes:

$$\frac{dM}{dt} = -\nabla_M \Delta(X, M), \tag{5}$$

with X treated as externally constrained by physical dynamics. OCTA is designed to realize this gradient flow in practice.

III. OCTA AS A STRUCTURED PROJECTION FAMILY

We now formalize OCTA at the level of projections and residue, independent of implementation details.

A. World and Model Spaces

Let X denote the (latent) world configuration and M denote the internal model state of the agent (or collection of agents). For training, we work with data streams $D = \{(x_t, o_t)\}$ where x_t are latent variables (possibly partially observed) and o_t are sensory observations.

We view (X, M) as an element of a joint configuration space $\mathcal{C}_{\text{joint}}$ and integrate over uncertainty via distributions $p(X)$ and $q(M)$.

B. Projection Families

OCTA instantiates four primary projection families:

1. **Perceptual projections** Π_{perc}:

$$\Pi_{\text{perc}} : (X, M) \mapsto O_{\text{perc}}, \tag{6}$$

capturing sensory modalities (e.g. text, vision, audio).

2. **Model projections** Π_{model}:

$$\Pi_{\text{model}} : M \mapsto \hat{O}_{\text{model}}, \tag{7}$$

representing predictive distributions over observations.

3. **Introspective projections** Π_{self}:

$$\Pi_{\text{self}} : M \mapsto Z, \tag{8}$$

where Z encodes higher-level summaries (e.g. beliefs, uncertainties, explanations) across internal lobes or subsystems.

4. **Social projections** Π_{soc}:

$$\Pi_{\text{soc}} : \{M_a\}_{a\in\mathcal{A}} \mapsto B, \tag{9}$$

where $\mathcal{A}$ indexes agents and B encodes shared beliefs or messages across agents.

Each projection induces predictive distributions over its outputs:

$$p(O_{\text{perc}}), \quad \hat{p}(O_{\text{model}} \mid M), \quad \hat{p}(Z \mid M), \quad \hat{p}_a(B \mid M_a). \tag{10}$$

C. Residue Decomposition in OCTA

We define OCTA's contradiction residue over (X, M) as

$$\Delta_{\text{OCTA}}(X, M) = \Delta_{\text{world}}(X, M) + \Delta_{\text{self}}(M) + \Delta_{\text{cons}}(\{M_a\}), \tag{11}$$

where:

- Δ_{world} measures mismatch between empirical and predicted distributions,
- Δ_{self} measures mismatch among internal views (lobes),
- Δ_{cons} measures mismatch among agents.

We will make these terms explicit in Sec. V.

IV. LOBE-BASED ARCHITECTURE

We now describe a concrete architectural realization of OCTA. While multiple implementation variants are possible (e.g. transformers, spiking networks, neuromorphic hardware), we present a general lobe-based abstraction.

A. High-Level Diagram

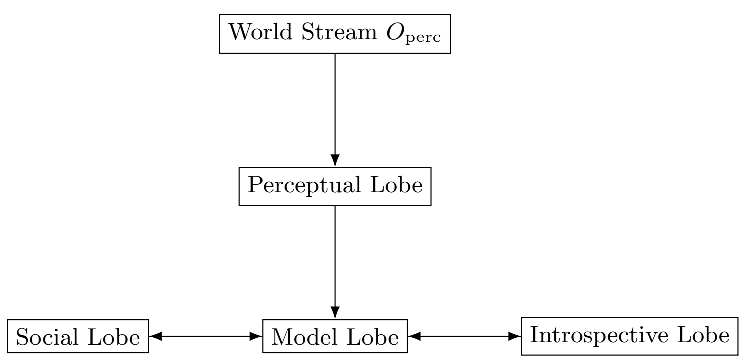

FIG. 1. High-level view of the OCTA lobe architecture. The perceptual lobe processes world observations, the model lobe maintains and updates predictive structure, the introspective lobe monitors and regularizes internal consistency, and the social lobe manages multi-agent coherence.

B. Perceptual Lobe

The perceptual lobe ingests raw observations o_t and produces embeddings e_t suitable for downstream prediction and reasoning. For concreteness, one can instantiate this as a stack of modality-specific encoders (e.g. vision transformers, audio encoders, text transformers), followed by a fusion mechanism.

Let

$$e_t = f_{\text{perc}}(o_t; \theta_{\text{perc}}), \tag{12}$$

where f_{perc} may itself be a multimodal transformer block, and θ_{perc} are trainable parameters.

C. Model Lobe

The model lobe maintains a latent state h_t summarizing both past observations and internal beliefs. This can be realized as:

- a transformer with memory (e.g. Transformer-XL-style recurrence),
- a sequence of recurrent blocks (e.g. gated RNNs),
- or a hybrid with external memory (e.g. key-value store).

For specificity, consider a transformer-like update:

$$h_t = f_{\text{model}}(h_{t-1}, e_t; \theta_{\text{model}}), \tag{13}$$

with θ_{model} trainable. The model lobe outputs predictive distributions over future observations and latent targets:

$$\hat{p}(o_{t+k} \mid h_t), \quad \hat{p}(y_t \mid h_t), \tag{14}$$

for some prediction horizon k and auxiliary labels y_t.

D. Introspective Lobe

The introspective lobe receives model states $\{h_t\}$ (possibly across different lobes or heads) and produces higher-order summaries z_t capturing internal confidence, disagreement, and proposed corrections.

Let

$$z_t = f_{\text{self}}(h_t; \theta_{\text{self}}). \tag{15}$$

This lobe can:

- estimate epistemic uncertainty,
- detect internal inconsistencies (e.g. disagreement between predictions from different heads or modes),
- generate self-queries and counterfactuals.

It induces distributions $\hat{p}(z_t \mid h_t)$ that can be compared across time or across lobes to form Δ_{self}.

E. Social Lobe

The social lobe operates when multiple OCTA instances (agents) interact. For agent a, let $h_t^{(a)}$ be its model state. The social lobe encodes messages and beliefs:

$$b_t^{(a)} = f_{\text{soc}}(h_t^{(a)}; \theta_{\text{soc}}), \tag{16}$$

and decodes incoming messages from other agents into updates. The induced distributions $\hat{p}_a(b_t \mid h_t^{(a)})$ are used to measure consensus residue Δ_{cons}.

F. Relation to Information Geometry

Endowing the space of model states with a Fisher information metric [3] allows one to interpret OCTA updates as approximate natural gradients on Δ_{OCTA}. While we do not develop full information geometry here, designing internal parameterizations so that KL divergences correspond to geodesic distances in model space is a natural extension.

V. LOSS FUNCTION AND OPTIMIZATION

We now define the OCTA training objective and relate it directly to contradiction residue.

A. World-Model Residue

Given ground-truth or empirical distributions $p(o_t)$ and model predictions $\hat{p}(o_t \mid h_t)$, we define:

$$\Delta_{\text{world}} = \sum_t D_{\text{KL}}\big(p(o_t) \,\big\|\, \hat{p}(o_t \mid h_t)\big), \tag{17}$$

summed over training samples t. In practice, $p(o_t)$ is represented by empirical observations and the KL reduces to a negative log-likelihood term plus constants.

B. Self-Consistency Residue

The model lobe can emit multiple predictive heads indexed by i (e.g. different timescales, modalities, or hypothesis branches). Let $\hat{p}_i(o_t \mid h_t)$ be the prediction from head i. Then

$$\Delta_{\text{self}} = \sum_t \sum_{i<j} D_{\text{KL}}\big(\hat{p}_i(o_t \mid h_t) \,\big\|\, \hat{p}_j(o_t \mid h_t)\big), \tag{18}$$

penalizes internal disagreement. Variants can use symmetrized forms (e.g. Jensen–Shannon divergence).

C. Consensus Residue

For multi-agent OCTA, with agents $a, b \in \mathcal{A}$ producing belief distributions $\hat{p}_a(b_t \mid h_t^{(a)})$ and $\hat{p}_b(b_t \mid h_t^{(b)})$, we define

$$\Delta_{\text{cons}} = \sum_t \sum_{a<b} D_{\text{KL}}\Big(\hat{p}_a(b_t \mid h_t^{(a)}) \,\big\|\, \hat{p}_b(b_t \mid h_t^{(b)})\Big). \tag{19}$$

This term pushes agents toward shared internal representations when warranted by shared evidence.

D. Total OCTA Loss

The full OCTA loss is

$$\mathcal{L}_{\text{OCTA}} = \lambda_{\text{world}}\Delta_{\text{world}} + \lambda_{\text{self}}\Delta_{\text{self}} + \lambda_{\text{cons}}\Delta_{\text{cons}}, \tag{20}$$

with nonnegative weights $\lambda_{\text{world}}, \lambda_{\text{self}}, \lambda_{\text{cons}}$ controlling the trade-off between accuracy, internal consistency, and consensus.

Under this definition, we can view

$$\Delta_{\text{OCTA}} \equiv \mathcal{L}_{\text{OCTA}}, \tag{21}$$

so that gradient-based optimization of $\mathcal{L}_{\text{OCTA}}$ is an instantiation of T-selection dynamics restricted to the model parameters.

E. Optimization Procedure

Training OCTA proceeds via standard stochastic gradient descent or its variants (e.g. Adam) on $\mathcal{L}_{\text{OCTA}}$, with minibatches sampled from data streams and, in multi-agent cases, from joint or partially shared experience.

An outline:

1. Sample a batch of sequences $\{o_t\}$ (and optional labels).
2. Run the perceptual lobe to obtain embeddings $\{e_t\}$.
3. Run the model lobe to obtain states $\{h_t\}$ and predictive distributions.

4. Run introspective and social lobes to obtain higher-order predictions and beliefs.
5. Compute $\Delta_{\text{world}}, \Delta_{\text{self}}, \Delta_{\text{cons}}$ via Eqs. (17)–(19).
6. Compute $\mathcal{L}_{\text{OCTA}}$ via Eq. (20) and backpropagate.

Standard regularization (e.g. weight decay, dropout, gradient clipping) can be added without changing the conceptual structure.

VI. EVALUATION METRICS: Δ-REDUCTION EFFICIENCY

Conventional evaluation of intelligent systems focuses on task accuracy or reward optimization. Within the T-selection framework, we introduce metrics that reflect how efficiently an agent reduces contradiction residue.

A. Instantaneous and Cumulative Residue

Given a trained or training OCTA instance, we can monitor:

- instantaneous residue $\Delta^{(t)}_{\text{OCTA}}$ at time t,
- cumulative residue $\sum_{t=1}^{T} \Delta^{(t)}_{\text{OCTA}}$ over a horizon T.

Definition 1 (Δ-Reduction Efficiency)**.** *For a given environment and resource budget (e.g. FLOPs, wall-clock time), the Δ-reduction efficiency of an agent is*

$$\eta_\Delta = -\frac{1}{R}\sum_{t=1}^{T} \frac{\mathrm{d}\Delta^{(t)}_{\text{OCTA}}}{\mathrm{d}t}, \tag{22}$$

where R is a measure of computational resources expended. Higher η_Δ indicates more efficient reduction of contradiction residue.

B. Task Performance vs. Closure

Task-level metrics (accuracy, reward, success rate) remain important, but we can now study their relationship to closure:

- Are agents with lower Δ_{self} more robust to distribution shift?
- Does lower Δ_{cons} predict better cooperative performance?
- Does sustained reduction in Δ_{world} correlate with improved generalization?

These empirical questions provide concrete tests for the value of OCTA's T-selection grounding.

VII. DISCUSSION AND FUTURE WORK

OCTA represents one possible instantiation of a T-selection-based AGI architecture. Many extensions and refinements are possible:

- **Information geometry.** Equipping model parameter space with a Fisher metric and using natural gradient methods [3] would align the optimization of $\mathcal{L}_{\text{OCTA}}$ with geodesics in information space, more faithfully approximating the gradient flow on Δ.
- **Recursive residue.** While OCTA primarily addresses cross-projection and consensus residue, extensions could incorporate explicit recursive (scale) residue by enforcing consistency between different temporal or spatial resolutions of predictions.

- **Hierarchical agents.** Embedding OCTA in multi-level organizations (e.g. agents managing other agents) could instantiate more complex bootstrap chains analogous to the hierarchical picture in Ref. [1].
- **Experimental programs.** Comparing conventional architectures and OCTA variants on controlled benchmarks where internal consistency and consensus matter (e.g. multi-view reasoning, multi-agent coordination) would provide evidence for or against the practical benefits of explicit T-closure optimization.

Ultimately, the ambition is to treat intelligence not merely as task performance but as progressive alignment between internal models and a global T-attractor. OCTA is a first step toward an architecture that embodies this idea.

VIII. CONCLUSION

We have introduced OCTA, a structured projection architecture for AGI grounded in the T-Selection Principle. By decomposing its objective into world-model, self-consistency, and consensus residues, OCTA implements a concrete approximation to T-selection dynamics over world and model states. This places internal consistency and multi-agent coherence on the same footing as predictive accuracy, and provides a set of evaluation metrics centered on Δ-reduction efficiency.

While the present exposition is architectural and conceptual, it is directly implementable with contemporary deep learning components. We hope this work motivates experimental studies of T-selection-inspired architectures and contributes to a more principled understanding of what it means for an intelligent system to be both powerful and internally coherent.

ACKNOWLEDGMENTS

We acknowledge prior foundational work on the T-Selection Principle in Ref. [1], which provides the theoretical basis for OCTA.

Appendix A: Appendix A: Implementation Notes (Transformer-Based OCTA)

In this appendix we provide high-level pseudo-code for a single-agent OCTA implementation using a transformer backbone. The goal is to show that the architecture described in the main text can be realized with standard deep learning components.

1. Model Components

We define the following parameterized modules:

- $f_{\mathrm{perc}}(\cdot;\theta_{\mathrm{perc}})$: perceptual encoder (e.g. token embedding + positional encoding + small transformer stack).
- $f_{\mathrm{model}}(\cdot;\theta_{\mathrm{model}})$: transformer-based model lobe.
- $\mathrm{Head}_i(\cdot;\theta_i)$: predictive heads (e.g. next-token, masked-token, auxiliary tasks).
- $f_{\mathrm{self}}(\cdot;\theta_{\mathrm{self}})$: introspective head producing meta-predictions (optional in the simplest version of the loss).
- $f_{\mathrm{soc}}(\cdot;\theta_{\mathrm{soc}})$: social head (used in multi-agent settings).

2. Pseudo-Code: Single-Agent OCTA Training Loop

```
# Pseudo-code for single-agent OCTA with Transformer backbone

# Parameters:
#   theta_perc, theta_model: perceptual and model lobe parameters
#   theta_heads: list of predictive head parameters [theta_1, ..., theta_K]
#   lambda_world, lambda_self: loss weights (lambda_cons = 0 in single-agent case)
```

```
for each training step:
    # 1. Sample batch of sequences
    #    o: observations (tokens, pixels, etc.)
    o_batch = sample_batch_from_dataset()

    # 2. Perceptual encoding
    e_batch = f_perc(o_batch; theta_perc)   # shape: [B, T, D_model]

    # 3. Model lobe forward (Transformer)
    h_batch = f_model(e_batch; theta_model) # shape: [B, T, D_model]

    # 4. Predictive heads (K different projections)
    pred_distributions = []
    for i in range(K):
        logits_i = Head_i(h_batch; theta_heads[i])    # shape: [B, T, V]
        p_hat_i = softmax(logits_i, dim=-1)           # predicted distribution
        pred_distributions.append(p_hat_i)

    # 5. Compute world-model residue (Delta_world)
    #    Assume we have supervised targets o_target (e.g., next tokens)
    o_target = compute_targets(o_batch)  # e.g. shifted tokens for next-token prediction

    # Cross-entropy approximates KL(p || p_hat) up to constants
    Delta_world = 0.0
    for i in range(K):
        ce_i = cross_entropy(pred_distributions[i], o_target)
        Delta_world += ce_i

    # 6. Compute self-consistency residue (Delta_self)
    #    Pairwise KL between predictive heads
    Delta_self = 0.0
    for i in range(K):
        for j in range(i+1, K):
            # KL(p_i || p_j) computed per-token, averaged over batch and time
            kl_ij = kl_divergence(pred_distributions[i], pred_distributions[j])
            Delta_self += kl_ij

    # 7. Total OCTA loss (single-agent, no consensus term)
    L_octa = lambda_world * Delta_world + lambda_self * Delta_self

    # 8. Backpropagation and parameter update
    L_octa.backward()
    optimizer.step()
    optimizer.zero_grad()
```

This code illustrates the direct correspondence between:

- Δ_{world}: sum of cross-entropies (KL from empirical to predicted distributions),
- Δ_{self}: sum of KL divergences between different heads' predictions,
- $\mathcal{L}_{\text{OCTA}}$: scalar loss used for gradient-based optimization.

3. Multi-Agent Extension (Consensus Residue)

In a multi-agent scenario with agents a and b:

- each agent has its own $f^{(a)}_{\text{model}}, f^{(b)}_{\text{model}}$,

- each produces a belief distribution $\hat{p}_a(b_t \mid h_t^{(a)})$, $\hat{p}_b(b_t \mid h_t^{(b)})$ via f_{soc},
- a consensus residue term Δ_{cons} is computed as in Eq. (19) and added to the total loss.

An analogous pseudo-code loop would:

1. run both agents,
2. compute $\Delta_{\text{world}}^{(a)}, \Delta_{\text{world}}^{(b)}$,
3. compute $\Delta_{\text{self}}^{(a)}, \Delta_{\text{self}}^{(b)}$,
4. compute Δ_{cons} from their social distributions,
5. combine all into $\mathcal{L}_{\text{OCTA,multi}}$.

Appendix B: Appendix B: Toy Environment and Experiment Specification

We now describe a minimal experimental setup in which $\Delta_{\text{world}}, \Delta_{\text{self}}$, and Δ_{cons} can be measured and compared for standard and OCTA-style architectures.

1. Single-Agent Multi-View Prediction

a. Environment. Consider a synthetic environment with a latent variable $z \in \mathbb{R}^d$ drawn from a known distribution (e.g. a Gaussian mixture). Two views $x^{(1)}, x^{(2)}$ are generated via known but nonlinear functions:

$$x^{(1)} = g_1(z) + \epsilon_1, \tag{B1}$$

$$x^{(2)} = g_2(z) + \epsilon_2, \tag{B2}$$

with noise terms ϵ_1, ϵ_2. For instance, g_1 and g_2 could be small neural networks or fixed random projections followed by nonlinearities.

b. Task. Given one view, predict the other and/or reconstruct z:

- forward prediction: $x^{(1)} \to x^{(2)}$,
- backward prediction: $x^{(2)} \to x^{(1)}$,
- joint latent inference: $(x^{(1)}, x^{(2)}) \to \hat{z}$.

c. Baselines.

- *Standard model*: single-head network trained to minimize mean squared error (MSE) between predicted and true views.
- *OCTA-style model*: same backbone, but with multiple prediction heads and a Δ_{self} term encouraging consistency between different projections (e.g. from $x^{(1)}$ and $x^{(2)}$).

d. Metrics.

- Δ_{world}: computed from MSE or negative log-likelihood (depending on output distribution).
- Δ_{self}: KL divergence between predictions from different heads (e.g. head predicting $x^{(2)}$ from $x^{(1)}$ vs. joint head using both views).
- η_Δ: efficiency of residue reduction per unit training step or FLOP.
- Generalization performance: MSE on held-out $x^{(1)}, x^{(2)}$ pairs from a shifted latent distribution z'.

The key questions are:

- Does explicit minimization of Δ_{self} improve robustness to latent distribution shift?
- Does the OCTA-style loss lead to lower overall Δ_{world} at test time compared to the standard model?

2. Multi-Agent Referential Game (Consensus Test)

a. Environment. Two agents observe different partial views of a scene:

- Agent A sees $x^{(A)}$,
- Agent B sees $x^{(B)}$,

where $(x^{(A)}, x^{(B)})$ are generated from a shared latent z as in the previous subsection. The agents play a referential game: A sends a discrete message m; B must identify a target (e.g. which latent class z belongs to).

b. Agents.

- Both agents use OCTA-style architectures with social heads $f^{(A)}_{\mathrm{soc}}, f^{(B)}_{\mathrm{soc}}$ mapping internal states to message distributions.
- They are trained jointly to maximize communication success while minimizing Δ_{cons}.

c. Metrics.

- Task success rate in the referential game.
- Consensus residue Δ_{cons}, measuring alignment of belief distributions over targets.
- Relationship between low Δ_{cons} and robustness to perturbations in $x^{(A)}, x^{(B)}$.

d. Comparisons.

- Compare standard emergent communication agents (optimizing only task reward) vs. OCTA agents (reward + consensus residue).
- Study whether agents with lower Δ_{cons} develop more interpretable and stable communication protocols.

This toy setup provides a concrete domain in which the conceptual advantages of T-selection-based objectives can be empirically tested and quantified.

REFERENCES

[1] OCTA Research, "The T-Selection Principle: A Fixed-Point Theory of Reality and Intelligence," (2025), arXiv:XXXX.XXXXX [gen-ph].
[2] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. (Wiley-Interscience, Hoboken, NJ, 2006).
[3] S.-I. Amari and H. Nagaoka, *Methods of Information Geometry*, Translations of Mathematical Monographs Vol. 191 (American Mathematical Society, Providence, RI, 2000).
[4] K. G. Wilson and J. Kogut, "The renormalization group and the ϵ expansion," Phys. Rep. **12**, 75–199 (1974).
[5] K. Friston, "The free-energy principle: a unified brain theory?," Nat. Rev. Neurosci. **11**, 127–138 (2010).
[6] P. D. Grünwald, *The Minimum Description Length Principle*, (MIT Press, Cambridge, MA, 2007).
[7] A. Vaswani *et al.*, "Attention is all you need," Adv. Neural Inf. Process. Syst. **30** (2017).
[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature **521**, 436–444 (2015).