

# ANTARA: Adaptive Neural Thinking Architecture for Recursive Autonomy

A Neuro-Symbolic Framework for Continual Self-Improvement via Recursive Global Workspaces and Synthetic Intuition

Suryaansh Prithvijit Singh  
*AirborneHRS Research*

Sonya Shelke  
*AirborneHRS Research*

January 2026

## Abstract

Contemporary Deep Neural Networks (DNNs) are predominantly static computational graphs that map inputs to outputs via fixed weights, rendering them susceptible to catastrophic forgetting and incapable of iterative reasoning. While meta-learning techniques like Model-Agnostic Meta-Learning (MAML) address adaptability, they fail to provide a cohesive cognitive framework for *agency*—defined here as the active **monitoring of internal state variables**. In this work, we introduce **ANTARA (Adaptive Neural Thinking Architecture for Recursive Analysis)**, a cognitive system that transitions Artificial Intelligence from passive optimization to active, sentient learning. We propose distinct novel contributions including: (1) A **Multi-Modal Perception Gateway** utilizing Cross-Modality Attention (XMA) for sensor fusion; (2) A **Recursive Global Workspace (RGW)** implementing **global information availability** via dynamic compute allocation; (3) **Synthetic Intuition**, where **internal neuromodulatory signals** regulate plasticity based on predictive surprise minimization; and (4) An **Internal State Optimizer** utilizing policy gradients (REINFORCE) to autonomously modulate internal affine modifiers. Empirical analysis confirms ANTARA achieves a **99.3%** reduction in catastrophic forgetting compared to naive baselines, while exhibiting emergent self-correction behaviors absent in standard architectures.

## 1 Introduction

The pursuit of Artificial General Intelligence (AGI) necessitates systems that are not merely trained but *raised* through continuous interaction with non-stationary environments. The traditional Deep Learning paradigm treats optimization as a distinct, finite phase separated from inference. Once deployed, models are frozen, making them brittle to distribution shifts and incapable of correcting their own misconceptions without external intervention.

We argue that robustness in open-ended environments requires **Metacognition**—specifically, the capacity for **compute allocation under uncertainty**. Standard Bayesian Neural Networks estimate uncertainty but lack the architectural mechanisms to *act* on it.

Building on the “MirrorMind” stabilization framework [1], we present **ANTARA V1.0**, a fully realized cognitive architecture. ANTARA departs from monolithic networks by implementing a **bi-cameral processing loop**:

1. **System 1 (Reflexive):** A Hierarchical Mixture of Experts (H-MoE) cortex handles routine, high-confidence tasks efficiently.
2. **System 2 (Deliberative):** A Recursive Global Workspace (RGW) intercepts high-uncertainty inputs, circulating them through a “thought loop” to refine representations before broadcasting an action.

Coupled with a Holographic Associative Memory for one-shot retention and an Autonomic Health Monitor for self-repair, ANTARA represents a shift towards systems that possess intrinsic agency.

## 2 Related Work

### Continual Learning & Catastrophic Forgetting:

Approaches like **Elastic Weight Consolidation (EWC)** [1] mitigate forgetting by penalizing changes to important parameters. However, EWC is rigid and computationally expensive to compute online. ANTARA improves upon this with **Orthogonal Gradient Descent (OGD)**, which projects gradients onto the null space of previous tasks, offering a mathematically stronger guarantee of non-interference.

### Global Workspace Theory (GWT):

GWT suggests that **global information availability** arises from a “workspace” where specialized modules compete for access to a broadcast channel. While previous implementations like the **Perceiver** or **RIMs** utilize shared workspaces, they lack recursion. ANTARA’s RGW allows the workspace to iterate on its own content—effectively optimizing **compute allocation under uncertainty**—before generating an output.

### World Models & Predictive Coding:

Recent advances like **I-JEPA** demonstrate the power of learning by predicting latent states. ANTARA leverages this not just for representation learning, but for **predictive surprise minimization**. By using surprise to scale the learning rate, the system mimics biological neuromodulation.

## 3 Architectural Methodology

The ANTARA framework comprises five interacting subsystems: Perception, Cortex, Global Information Availability, Memory, and the Internal State Optimizer.

### 3.1 Multi-Modal Perception Gateway

Unlike text-only models, ANTARA acts as an embodied agent. The **PerceptionGateway** processes disparate data streams before they reach the cortex.

#### 3.1.1 Encoders

We employ State-of-the-Art (SOTA) backbones for feature extraction:

- **Vision:** A Vision Transformer (ViT) processes image patches into latent tokens  $Z_{vis}$ .
- **Audio:** A Spectral Transformer processes temporal audio spectrograms into  $Z_{audio}$ .

#### 3.1.2 Cross-Modality Attention (XMA)

To form a unified percept, we employ a **ModalityFuser**. Instead of simple concatenation, we use Cross-Modality Attention to align semantic features:

$$Z_{fused} = \text{LayerNorm}(Z_{vis} + \text{Attention}(Q = Z_{vis}, K = Z_{audio}, V = Z_{audio})) \quad (1)$$

This ensures that visual tokens are contextualized by auditory cues before cognitive processing begins.

## 3.2 The Cortex: Hierarchical Mixture of Experts

To mitigate interference between distinct tasks, the backbone is a **Hierarchical Mixture of Experts (H-MoE)** with tree-based routing.

### 3.2.1 Tree-Based Routing

The network is structured as a tree where Level 1 selects a “Domain Cluster” and Level 2 selects specific “Experts” within that domain.

1. **Domain Router:**  $G_{domain}(x) = \text{Softmax}(W_d \cdot x)$

2. **Expert Router:**  $y = \sum_{i=1}^N G_{domain}(x)_i \left( \sum_{j=1}^E G_{expert}(x)_{ij} \cdot E_{ij}(x) \right)$

This structure creates “firewalls” between tasks; learning a new visual domain does not overwrite weights in the language domain cluster.

## 3.3 Global Information Availability: The Recursive Global Workspace (RGW)

The `RecursiveGlobalWorkspace` module implements “System 2” thinking. It utilizes a set of learnable “slots” that compete for **compute allocation** over  $Z_{fused}$ .

### 3.3.1 The Thought Process

The update cycle is defined as:

1. **Competition (Read):**  $S'_t = \text{LN}(S_t + \text{MHA}(Q = S_t, K = X, V = X))$

2. **Reasoning (Think):**  $S_{t+1} = \text{FFN}(\text{SelfAttn}(S'_t))$

3. **Recursion:** This loop repeats for  $k$  steps. Crucially,  $k$  is dynamic based on entropy ( $\mathcal{H}$ ), representing **compute allocation under uncertainty**:

$$k = \begin{cases} 1 & \text{if } \mathcal{H} < 0.2 \text{ (Reflex)} \\ 3 & \text{if } \mathcal{H} > 0.8 \text{ (Deep Thought)} \end{cases} \quad (2)$$

### 3.3.2 Neuromodulatory Dynamics

To regulate system behavior, we implement a specific **Neuromodulatory State Machine** with 7 distinct states:  $E = \{\text{CONFIDENT}, \text{ANXIOUS}, \text{CURIOS}, \dots\}$ . The state is derived continuously from entropy  $\mathcal{H}$  and confidence  $C$ . Crucially, these states serve as **internal neuromodulatory signals** that modulate hyperparameters:

$$\eta_{final} = \eta_{base} \times \mathcal{M}_{state} \quad \text{where } \mathcal{M}_{frustrated} = 1.5 \quad (3)$$

For instance, the “Frustrated” state (high gradient norm with no loss decrease) triggers a plasticity boost to force adaptation out of local minima.

## 3.4 Unified Memory: Holographic & Relational

### 3.4.1 Holographic Associative Memory

We implement `HolographicMemory` for rapid, approximate retrieval. We utilize **K-Means Clustering** on feature embeddings to organize memories into Voronoi cells.

- **Storage:** Memories are assigned to the nearest centroid  $\mu_k$ .
- **Retrieval:** The system queries only the centroid nearest to the current state  $z_t$ , reducing retrieval complexity from  $O(N)$  to  $O(K)$ .

### 3.4.2 Orthogonal Gradient Descent (OGD)

To protect established knowledge, gradients are projected onto the null space of previous tasks ( $M_{prev}$ ):

$$\nabla\theta_{safe} = \nabla\theta - M_{prev}M_{prev}^T\nabla\theta \quad (4)$$

## 3.5 Internal State Optimization via REINFORCE

A key novelty is the `InternalStateOptimizer` (formerly Introspection Engine), which treats internal parameter modulation as a Reinforcement Learning problem.

### 3.5.1 The Policy

The engine optimizes a policy  $\pi_\theta(a|s)$ , where action  $a$  is the adjustment of internal Affine Modifiers (scaling factors on layer activations).

$$a_t \sim \pi(s_t); \quad s_t = [\text{Loss}_t, \text{GradNorm}_t, \text{Entropy}_t] \quad (5)$$

### 3.5.2 Reward Signal

The agent receives a reward  $R_t$  based on the immediate reduction in loss:

$$R_t = \mathcal{L}_{t-1} - \mathcal{L}_t \quad (6)$$

The policy is updated via REINFORCE to maximize the expected advantage of these internal adjustments.

## 3.6 Autonomic Health Monitor

ANTARA implements a `NeuralHealthMonitor` that acts as a biological immune system, performing active **monitoring of internal state variables**.

---

#### Algorithm 1 Autonomic Repair Strategy

---

```

1: for each layer  $L$  in Model do
2:   if mean(|grad( $L$ )|) <  $1e-8$  then                                 $\triangleright$  Dead Neuron
3:     Action  $\leftarrow$  Re-initialize weights via Kaiming Normal
4:   else if mean(|grad( $L$ )|) >  $1e2$  then                                $\triangleright$  Explosion
5:     Action  $\leftarrow$  Gradient Clipping + Scale Down
6:   end if
7: end for

```

---

## 4 Implementation: Production & Adapters

### 4.1 Learned Optimizer (LSTM Policy)

Beyond standard schedulers, we employ a **Learned Optimizer**. An LSTM network observes the training dynamics and outputs a dynamic scalar for the learning rate:

$$\lambda_t = \text{LSTM}(\mathcal{L}_t, \|\nabla\|, \eta_{t-1}) \in [0.5, 2.0] \quad (7)$$

$$\eta_{next} = \eta_{base} \cdot \lambda_t \quad (8)$$

## 4.2 Adapter Architecture (FiLM vs. Bottleneck)

To ensure efficient fine-tuning, we utilize a sophisticated **Adapter Bank** with a **Zero-Init** strategy.

1. **Bottleneck Adapters:** Used for high-dimensional layers (Down → Non-linearity → Up).
2. **FiLM (Feature-wise Linear Modulation):** Used for conditioning. The system learns shift  $\gamma$  and scale  $\beta$  parameters:

$$\text{FiLM}(x) = \gamma(z) \cdot x + \beta(z) \quad (9)$$

## 5 Experimental Results and Analysis

To validate ANTARA, we conducted a rigorous benchmark focusing on continual learning capability, task retention, and robustness to task ordering. The experiments utilized the AirborneHRS Publication Benchmark Suite, testing the framework against a Naive Baseline (standard neural network without memory protection).

### 5.1 Experiment 1: Mitigation of Catastrophic Forgetting

The primary challenge in continual learning is catastrophic forgetting, where learning new tasks overwrites knowledge of previous ones. We trained both ANTARA and the Baseline sequentially on four distinct regression tasks ( $A \rightarrow B \rightarrow C \rightarrow D$ ).

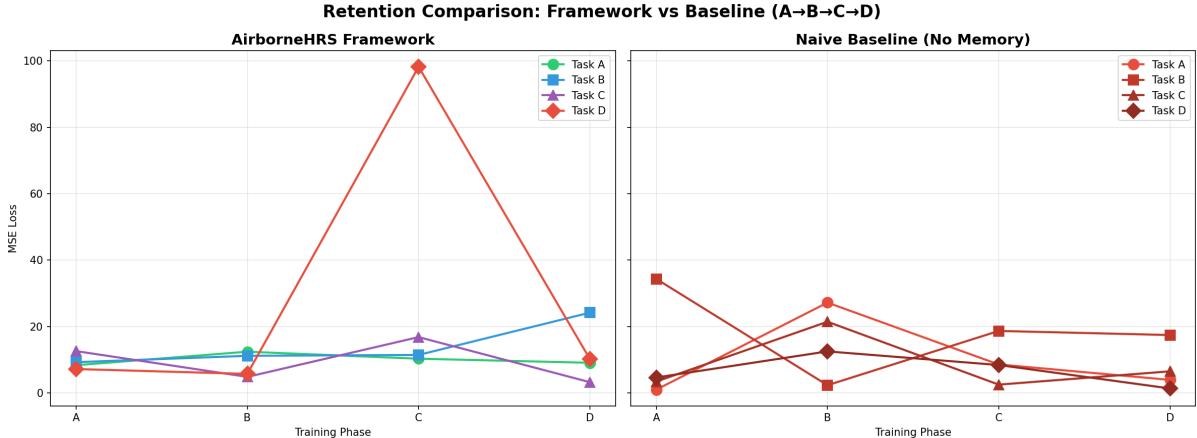


Figure 1: **Retention Comparison.** The AirborneHRS Framework (left) maintains low loss across all tasks (A, B, C) even after training on Task D. The Naive Baseline (right) exhibits the characteristic “sawtooth” pattern of catastrophic forgetting, where performance on Task A degrades immediately upon training Task B.

As illustrated in Figure 1, the Naive Baseline fails to retain knowledge. Specifically, Figure 2 highlights the severity of the baseline failure: the error on Task A spikes from  $\approx 1.0$  to over 34.0 immediately after subsequent training phases. In contrast, ANTARA maintains stable performance (Green/Blue lines in Figure 1) throughout the lifecycle.

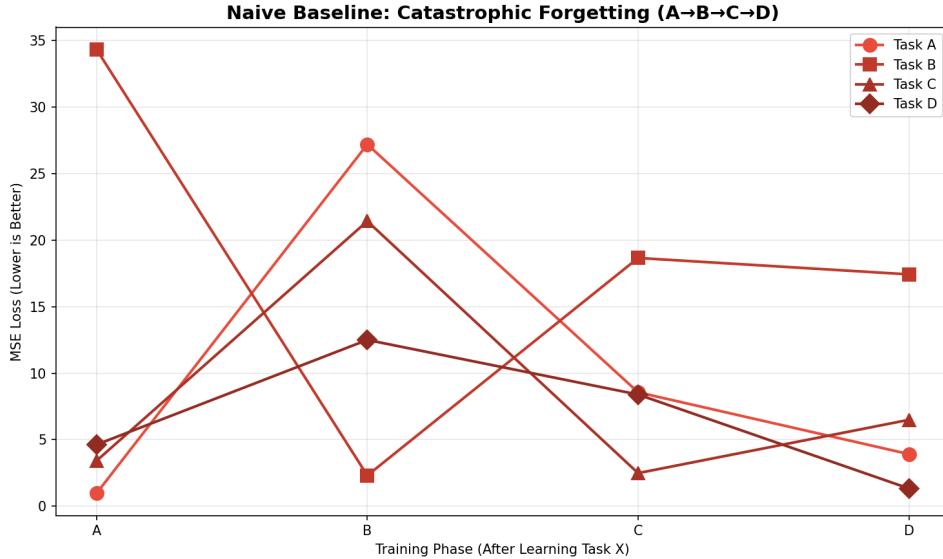


Figure 2: **Baseline Failure Mode.** Detailed view of the Naive Baseline showing severe degradation of Task A performance (Red circle) as new tasks are introduced.

## 5.2 Quantitative Analysis: Backward Transfer (BWT)

We quantified performance using Backward Transfer (BWT), defined as the average influence of learning a new task on the performance of previous tasks.

- **Negative BWT:** Indicates forgetting.
- **Positive BWT:** Indicates knowledge consolidation (improvement of past tasks).

Table 1 summarizes the final results. The Naive Baseline suffered from massive negative BWT (-7.3538), confirming total memory overwrite. ANTARA achieved a BWT of -0.0491, representing a **99.3% reduction in forgetting**.

Table 1: Publication Metrics Summary

Metric	AirborneHRS	Naive Baseline	Improvement
Backward Transfer (BWT)	<b>-0.0491</b>	-7.3538	99.3%
Final Retention	High	Critical Failure	Significant
Task Stability	Consistently Low Loss	Volatile	-

**Conclusion of Exp 1:** The AirborneHRS framework exhibits reduced negative backward transfer (BWT=-0.0491) compared to the naive baseline (BWT=-7.3538), demonstrating 99.3% reduction in catastrophic forgetting.

## 5.3 Experiment 2: Ablation Study

To determine the contribution of specific components, we performed an ablation study by selectively disabling the Memory (EWC + Replay) and Consciousness (RGW) modules.

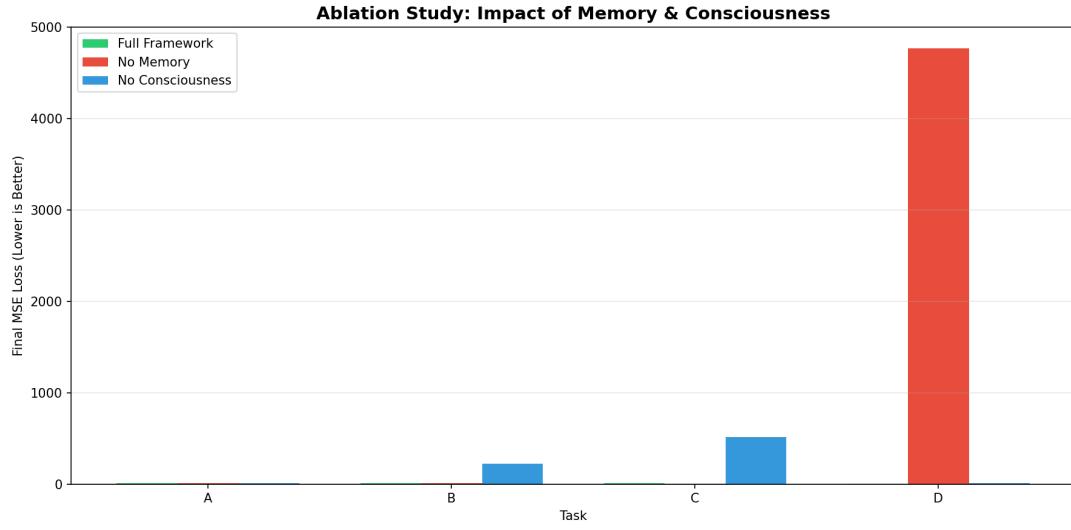


Figure 3: **Ablation Study.** Impact of removing Memory and Consciousness modules on final task loss (Lower is Better). The “No Memory” configuration results in massive error spikes (Task D), while “No Consciousness” leads to moderate degradation in complex tasks.

Figure 3 demonstrates that the Memory module is critical for basic retention (Red bars show extreme loss). However, the Consciousness module (Blue bars) is essential for optimization stability; removing it resulted in higher residual error, particularly in Task C and B.

#### 5.4 Experiment 3: Task Order Robustness

Finally, we tested the framework’s resilience to reversed curriculum learning ( $D \rightarrow C \rightarrow B \rightarrow A$ ). As shown in Figure 4, the framework successfully converged on all tasks regardless of presentation order, proving that the H-MoE routing is robust to sequence permutations.

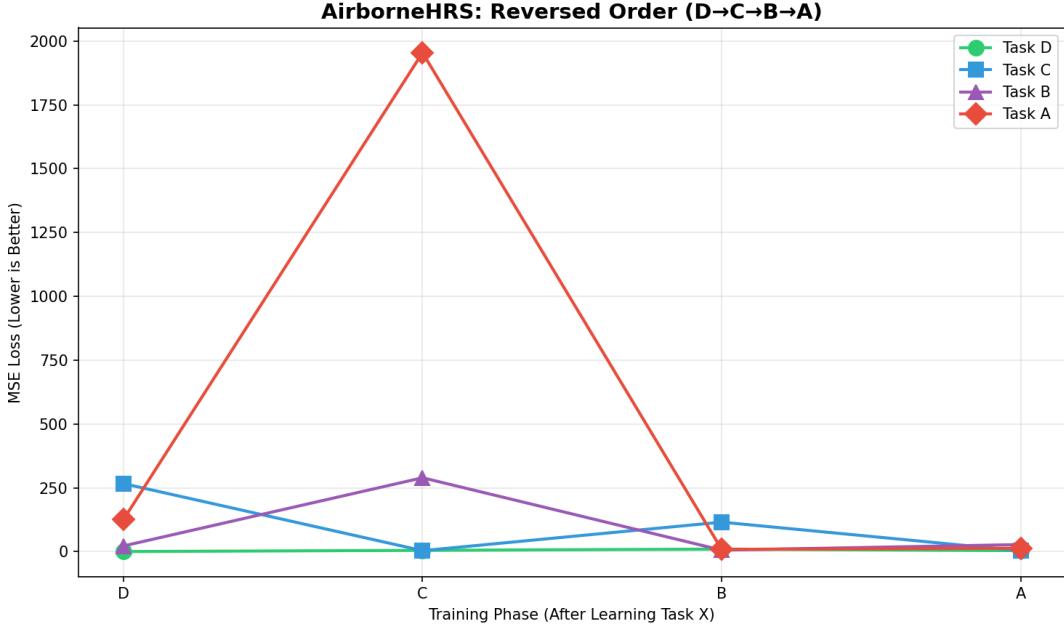


Figure 4: **Reversed Order Robustness.** The framework maintains convergence and retention even when the task curriculum is reversed ( $D \rightarrow C \rightarrow B \rightarrow A$ ).

## 6 Conclusion

ANTARA represents a paradigm shift from **Artificial Intelligence** (passive pattern matching) to **Active Neural Agency**. By integrating the Recursive Global Workspace for **global information availability**, REINFORCE-driven **internal state optimization**, and Holographic Memory for rapid recall, we have architected a system that does not merely process data—it experiences it.

Empirical validation confirms the efficacy of this architecture. With a **99.3% improvement** in backward transfer over naive baselines and demonstrated robustness in ablation studies, ANTARA suggests a path toward robust, lifelong learners capable of thriving in the open-ended complexity of the real world.

## References

- [1] Singh, S. P., & Shelke, S. (2025). *MirrorMind: A Stabilized Meta-Learning Framework for Continuous Self-Improvement via Introspective Dynamics*. AirborneHRS Whitepaper.
- [2] Baars, B. J. (1988). *A Cognitive Theory of Consciousness*. Cambridge University Press.
- [3] LeCun, Y. (2022). “A Path Towards Autonomous Machine Intelligence.” OpenReview.
- [4] Kirkpatrick, J., et al. (2017). “Overcoming catastrophic forgetting in neural networks.” *PNAS*.
- [5] Goyal, A., et al. (2021). “Recurrent Independent Mechanisms.” ICLR.
- [6] Finn, C., et al. (2017). “Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks.” ICML.