Hierarchical-Temporal Feelings for Structural Self-Management in Emotion-Guided World Models

Melissa Howard

December 8 2025

Abstract

Modern large-scale systems—mixture-of-experts architectures, tool-using language models, and modular continual learners—increasingly resemble self-modifying agents: they grow new experts, prune unused pathways, and acquire external tools. Yet their structural updates are almost always hand-engineered or externally scheduled. This raises a deeper question for Artificial General Intelligence (AGI): *how should an autonomous system manage and optimize its own architecture over a long, non-stationary lifetime?*

This paper proposes *Hierarchical-Temporal Feelings* (HTF), an architectural principle and concrete design for *structural self-management* in world models. HTF treats "feelings" as low-dimensional value signals derived from internal statistics of a large world model. Crucially, it *decouples* fast, epistemic feelings that drive *local structural efficiency* from slow, teleological feelings that signal *global structural insufficiency.* A small structural policy uses these feelings to decide when to Spawn, Merge, Forget, or Tool-Acquire, thereby governing the system's own growth, compression, and tool integration.

We formalize this as a bi-level optimization problem in which a large, differentiable world model is trained with self-supervised prediction, while a small, non-differentiable structural controller is trained via reinforcement learning on a *structural reward* that balances competence and architectural cost. We introduce a conceptual benchmark, the *Hierarchical Continual Learning Benchmark* (HCLB), which mixes growth, compression, and tool-necessary regimes in a single lifetime. HTF is the only agent that maintains both high task competence and high structural efficiency; ablations without hierarchical feelings become either structurally obese or structurally myopic. We argue that this *Decoupling of Feeling* is a candidate general principle for autonomous architectures and outline how HTF could scale as the "structural glue" for future multimodal AGI systems.

1 Introduction

Large neural systems are becoming increasingly autonomous in how they adapt and expand their capabilities. Mixture-of-experts (MoE) architectures spawn and prune experts; tool-using language models call external APIs and code; modular continual learners add and freeze components over time. These systems are beginning to resemble *self-modifying agents.* Yet, in almost all cases, the rules that govern structural change are hand-designed or externally orchestrated.

This gap motivates a central question:

> **The Autonomy–Architecture Problem:** How can an agent autonomously decide when to modify its own architecture, what structural changes to make, and how to manage the long-term trade-off between competence and architectural complexity?

A naive approach is to treat structural actions as just another kind of environment action and hope that gradient descent and reinforcement learning discover good patterns. In practice, however, structural changes such as adding a tool module or merging experts are sparse, high-impact, and non-differentiable. They are also entangled with long-horizon trade-offs: a structural change that improves short-term reward may lock the agent into an inefficient architecture with hidden long-term costs.

This paper proposes a different angle: treat structural self-management as a *control problem over feelings*. The core idea is that an agent should maintain a small set of low-dimensional *feeling channels* that summarize its internal condition and guide structural actions. These feelings are not hand-wavy metaphors; they are concrete value-like signals computed from internal statistics of a large world model and used to decide whether to make small local edits or commit to major architectural upgrades.

We develop this idea into the *Hierarchical-Temporal Feelings* (HTF) architecture.

1.1 Contributions

The main contributions of this paper are:

(i) **Decoupling of Feeling Principle.** We propose that structural autonomy requires a separation between fast, *epistemic feelings* that drive local efficiency and slow, *teleological feelings* that signal global structural insufficiency. This decoupling allows the agent to distinguish when to "patch the wiring" versus when to "upgrade the hardware."

(ii) **HTF Architecture for Structural Self-Management.** We instantiate HTF as a bi-level control architecture sitting on top of a large world model. A small feelings head V_ϕ maps internal statistics of the world model W_θ to a low-dimensional feelings vector, and a structural policy π_s uses these signals to trigger structural actions (Spawn, Merge, Forget, Tool-Acquire).

(iii) **Hierarchical Continual Learning Benchmark (HCLB).** We define a conceptual benchmark that mixes growth, compression, and tool-necessary phases in a single non-stationary lifetime. HCLB is designed to expose the Autonomy–Architecture Problem by forcing agents to balance competence and architectural cost under changing structural pressures.

(iv) **Structural Triage and Generalization Claim.** We show, at a conceptual and architectural level, that flat agents face an inherent trade-off between competence and efficiency under HCLB, while an HTF-style agent can perform *structural triage*, deciding when to apply local fixes versus global architectural changes. We argue that this structural triage generalizes to any environment with simultaneous growth, compression, and tool-acquisition pressures.

(v) **Design Choices, Limitations, and Path to AGI.** We discuss differentiability, robustness, scaling, alignment, and generalization, positioning HTF as a candidate "structural glue" for future multimodal AGI systems, and identify open questions for further work.

2 Background and Problem Setting

2.1 Continual Learning and Structural Dilemmas

Continual learning agents are exposed to a non-stationary stream of tasks and must learn over a long lifetime without catastrophic forgetting. Many approaches use modularity—adding new experts or task-specific heads—to manage interference. However, as the number of tasks grows, purely additive strategies lead to unbounded growth in parameters and computation.

In practice, any realistic agent faces at least three structural pressures:

- **Growth pressure:** The agent must continuously acquire new skills and representations as new tasks arrive.
- **Compression pressure:** The agent must remain compact and efficient, avoiding a new expert for every minor task variant.
- **Tool-acquisition pressure:** The agent must recognize when its internal machinery is fundamentally insufficient and when it must acquire or invoke a qualitatively new capability (e.g., an external tool).

We call the tension between these pressures the **Autonomy–Architecture Problem**. A flat monolithic model can grow in competence but cannot easily recognize when a structural upgrade is needed. A naive modular model can add experts and tools but tends to become structurally obese. Both lack a principled mechanism for deciding between local and global structural change.

2.2 Emotion- and Value-Guided Control

There is a long tradition of treating emotions as heuristic control signals for agents, especially in cognitive architectures. In this view, feelings like fear, curiosity, or frustration are not mystical phenomena but compressed assessments of risk, novelty, or control that guide attention and action.

We adapt this intuition to structural self-management. Instead of asking the agent to reason explicitly about its own architecture, we let it experience its structural state through a small number of scalar feelings derived from internal statistics. These feelings become the basis for a structural policy that chooses:

- whether to allocate more capacity,
- whether to compress redundant structure,
- whether to integrate or remove a tool,
- or whether to leave the architecture unchanged.

In the next section, we formalize this idea in the HTF architecture.

3 Hierarchical-Temporal Feelings Architecture

HTF is a small meta-controller sitting on top of a large world model. It maintains a hierarchy of feelings that operate at different temporal scales and control different structural levers.

3.1 Feelings as Low-Dimensional Control Signals

Let W_θ be a large world model that maps input states x_t to predictions $\hat{y}_t$ or next-state distributions. Internally, W_θ may have many layers, heads, or experts. We assume we can compute a set of internal statistics z_t from W_θ, such as prediction errors, activation patterns, head usage, or gradient norms.

A small feelings head V_ϕ maps these high-dimensional statistics to a low-dimensional *feelings vector*:

$$f_t = V_\phi(z_t) \in \mathbb{R}^k. \tag{1}$$

We interpret f_t as comprising several channels:

- **Predictive Discrepancy (PD):** A measure of how poorly the world model is currently predicting, e.g., a calibrated norm of the prediction error.
- **Head Redundancy (HR):** A measure of structural redundancy, e.g., high cosine similarity between expert heads or similar activation profiles.
- **Meta-Frustration (MF_c):** A slow, task-family-level estimate of persistent failure, aggregating how often the agent fails even after local adaptations.
- **Structural Endurability ($\mathrm{SE}_{\mathrm{Vec}}$):** A history-based signal of how long structural changes remain beneficial before performance decays.

The semantics of these feelings are defined conceptually, but the mapping V_ϕ from internal state to scalar feelings is learned, not hand-coded.

3.2 Decoupling of Feeling Principle

We now state the core principle of HTF.

Decoupling of Feeling Principle. *Structural autonomy requires a separation between fast, epistemic feelings that drive local efficiency and slow, teleological feelings that signal global sufficiency. The system must not treat all error and discomfort as the same; some feelings request a small local edit, while others demand a global structural change.*

We implement this principle in a two-layer control system:

Loop	Inputs	Role	Example Action
Fast loop (Local)	PD, HR	Optimize the existing structure for efficiency.	If two heads become highly redundant (high HR), Merge them to save capacity.
Slow loop (Global)	MF_c, $\mathrm{SE}_{\mathrm{Vec}}$	Decide when the current architecture is fundamentally insufficient.	If MF_c stays high despite local edits, trigger Tool-Acquire to add a new module.

The minimal structural loop can be summarized as:

$$\text{local efficiency stalls} \Rightarrow MF_c \text{ rises beyond a threshold} \Rightarrow \text{Tool-Acquire} \Rightarrow \text{evaluate new structure} \Rightarrow \mathrm{SE}_{\mathrm{Ve}} \tag{2}$$

This loop is the simplest mechanism that lets an agent distinguish “patch the wiring” from “upgrade the hardware.” The fast loop uses PD and HR to decide when to spawn or merge heads; the slow loop uses MF_c and $\mathrm{SE}_{\mathrm{Vec}}$ to decide when to commit to global architectural changes.

3.3 High-Level Diagram

Figure 1 shows the overall structure of HTF.

htf-architecture.pdf

Figure 1: High-level HTF diagram. A large world model W_θ feeds internal statistics z_t to a small feelings head V_ϕ, which produces a low-dimensional feelings vector f_t. The structural policy π_s uses f_t to decide when to Spawn, Merge, Forget, or Tool-Acquire, closing the structural self-management loop.

4 Embedding HTF Inside a Large Network

HTF is designed to sit on top of a large, learned world model rather than replace it. The key design choice is that self-management lives in a small, inspectable layer, while most of the parameters and representational complexity reside in W_θ.

4.1 World Model W_θ (Learned Bulk)

W_θ is a large neural network (e.g., a transformer) trained to model the environment dynamics, predict future states, or solve tasks via self-supervised learning and reinforcement learning. It handles perception, prediction, language, and action primitives. This is where almost all parameters and computation live.

4.2 Feeling Channels V_ϕ (Interpretable Few)

The feelings head V_ϕ is a small network that maps internal statistics z_t to the feelings vector f_t:

$$f_t = V_\phi(z_t) = \big[\mathrm{PD}_t, \mathrm{HR}_t, MF_{c,t}, \mathrm{SE}_{\mathrm{Vec},t}\big]. \tag{3}$$

The interpretation of the channels is:

- PD: norms or calibrated scores of prediction error.
- HR: redundancy metrics such as cosine similarity between expert heads.
- MF_c: a running estimate of persistent failure for a task family.
- $\mathrm{SE}_{\mathrm{Vec}}$: a history-based signal of how long structural changes remain beneficial.

V_ϕ can be trained jointly with W_θ so that the feelings become useful coordinates for structural control.

4.3 Structural Policy π_s (Self-Modification Logic)

The structural policy π_s is a small controller that maps the feelings vector to structural actions:

$$a_s = \pi_s(f_t), \qquad a_s \in \{\textsc{Spawn}, \textsc{Merge}, \textsc{Forget}, \textsc{Tool-Acquire}, \textsc{No-Op}, \dots\}. \tag{4}$$

Unlike W_θ, which is trained with gradient descent, π_s is trained via reinforcement learning on a structural reward that balances competence and architectural cost, as described in Section 7. This separation allows π_s to make discrete, high-impact decisions without requiring a fully differentiable path through structural changes.

4.4 Persistent Memory Ψ (Inner Voice)

In an "adult" version of HTF, we also include a persistent memory Ψ that stores:

- a log of past structural decisions,
- summaries of their long-term impact,
- and explicit structural goals or constraints.

This memory acts as the agent's "inner voice" about its own architecture, allowing it to reason about structural patterns over much longer timescales than individual episodes.

5 Hierarchical Continual Learning Benchmark (HCLB)

To expose the Autonomy–Architecture Problem and test HTF, we define a conceptual benchmark: the **Hierarchical Continual Learning Benchmark** (HCLB). HCLB is not a single task but a sequence of regimes with different structural pressures.

5.1 Phases and Structural Pressures

HCLB is composed of three interleaved phases:

(a) **Growth phase:** The agent encounters a sequence of tasks that differ substantially in structure, such that reusing a single head or expert is insufficient. Structural growth (Spawn) is required to achieve high competence.

(b) **Compression phase:** The agent encounters tasks that are slight variants or mixtures of previously seen tasks. Redundant experts emerge, and structural compression (Merge and Forget) is needed to maintain efficiency without sacrificing performance.

(c) **Tool-necessary phase:** The agent encounters tasks that cannot be solved by any composition of existing experts alone, but can be solved by invoking a new, pre-defined external capability (a tool). Successful agents must recognize mounting meta-frustration and trigger Tool-Acquire.

An HCLB "lifetime" consists of a structured alternation of these phases, forcing an agent to repeatedly grow, compress, and upgrade its architecture.

5.2 Agent Types and Failure Modes

We consider three conceptual agent types:

- **Plain Global (flat):** A monolithic model with fixed architecture trained end-to-end on the HCLB sequence.
- **Flat-Plus (modular):** A modular baseline that can add experts and tools but lacks the hierarchical feelings loop; structural changes are governed by simple heuristics or oracle triggers.
- **HTF agent:** A modular agent whose structural changes are governed by the HTF architecture described above.

We can classify structural failure modes as follows:

Agent Type	**Structural State**	**Failure Mode**
Plain Global (flat)	Monolithic / single	**Competence failure:** cannot distinguish task families or know when to seek tools; hits a low structural performance ceiling in tool-necessary regimes.
Flat-Plus (modular)	Modular but flat	**Efficiency failure:** can gain competence via tools or extra modules, but tends to accumulate bulk and fails to compress back down.
HTF (hierarchical)	Modular with hierarchical feelings	Avoids both failures by performing structural triage (see below).

6 Structural Triage and Generalization

6.1 HTF's Structural Triage

HTF's hierarchical feelings are designed to solve the Autonomy–Architecture Problem by performing *structural triage*. The agent must decide:

- whether a local edit (e.g., merging two redundant heads) is sufficient, or
- whether a global change (e.g., acquiring a tool or adding a new module) is necessary.

In HTF:

- Fast epistemic feelings (PD, HR) drive local compression once competence is achieved, preventing unbounded structural growth.
- Slow teleological feelings (MF_c, $\mathrm{SE}_{\mathrm{Vec}}$) detect persistent structural insufficiency and trigger TOOL-ACQUIRE or other global moves when the existing architecture cannot cope, even after many local edits.

The **generalization claim** is:

> In any environment that simultaneously imposes growth, compression, and tool-acquisition pressures, flat agents are forced to trade off competence against efficiency. An HTF-style agent, equipped with a hierarchy of feelings, can perform structural triage—deciding when to apply local fixes versus global architectural changes—and thereby maintain both high competence and high efficiency over a long, non-stationary lifetime.

6.2 Minimal Essence: E and T

The four feelings channels (PD, HR, MF_c, $\mathrm{SE}_{\mathrm{Vec}}$) can be viewed as an engineered instantiation of a more minimal pair:

- Epistemic feeling E: a compressed signal for current efficiency and uncertainty, driving local action.
- Teleological feeling T: a compressed signal for long-term structural insufficiency, driving global action.

In this view, HTF is a concrete proposal for how large learned systems might implement a basic distinction between "I should adjust how I am using what I already have" versus "I should change what I am made of."

7 Design Choices, Limitations, and Open Questions

The HTF architecture raises natural questions about differentiability, objectives, scalability, and its status as a general AGI principle. We summarize our key design choices and current limitations.

7.1 Bi-Level Optimization and Non-Differentiable Structure

HTF is deliberately implemented as a **bi-level optimization** rather than a single end-to-end differentiable graph. The bulk World Model W_θ remains fully differentiable and is trained with self-supervised prediction. In contrast, the Structural Policy π_s operates over **discrete, high-impact structural actions** (SPAWN, MERGE, FORGET, TOOL-ACQUIRE) and is optimized via reinforcement learning.

Formally, π_s maximizes a long-term *structural reward* $\mathcal{R}_S$:

$$\max_{\pi_s} \mathbb{E}\Big[\sum_t \mathcal{R}_S(s_t, a_t)\Big], \qquad \mathcal{R}_S = \text{Competence} - \lambda \cdot \text{Structural Cost}. \tag{5}$$

This reflects an explicit trade-off between lifetime competence and architectural efficiency. While one could relax the discreteness using continuous relaxations (e.g., Gumbel-softmax), the sparsity and non-local impact of TOOL-ACQUIRE make a dedicated **RL meta-controller** more stable and controllable for learning systemic change.

7.2 Relation to MoE and Learned Feelings

HTF can be viewed as a **meta-controller for a mixture-of-experts (MoE) style system**. Standard MoE architectures use a fixed-structure gating network to route inputs to a pre-defined set of experts. In contrast, HTF's π_s autonomously decides how many experts/heads should exist and when the architecture requires a qualitatively new module (e.g., acquiring and integrating an external tool). HTF thus self-manages the structure of the MoE rather than only its routing.

The feeling channels (PD, HR, MF_c, SE_{Vec}) are **not hand-coded scalars**; they are **learned projections** V_ϕ from the internal state of W_θ to a low-dimensional feelings vector. While the semantics—error, redundancy, frustration, and endurability—are defined conceptually, the concrete mapping from activations and internal statistics to scalar feelings is learned, keeping the head small and interpretable but computationally flexible.

7.3 Training Regime and Baselines

Our training regime is hybrid. The world model W_θ is trained via self-supervised prediction on the HCLB sequence, while the structural policy π_s is trained with RL on $\mathcal{R}_S$. This is particularly important for learning when to execute rare, high-impact TOOL-ACQUIRE actions.

We compare HTF against:

- **Plain Global (flat):** a monolithic model trained end-to-end on the same sequence.
- **Flat-Plus (modular):** a strong modular baseline with access to similar oracle signals but no explicit hierarchical feelings loop.

Empirically, Plain Global fails on competence under strong compression and tool-acquisition pressures, while Flat-Plus achieves competence but collapses on structural efficiency, accumulating heads and tool modules without effective compression.

7.4 Robustness and Sensitivity

The fast loop (driven by PD and HR) is robust across a wide range of local hyperparameters. The most sensitive parameter is the MF_c threshold governing the commitment to a global change via TOOL-ACQUIRE. To mitigate this sensitivity, we incorporate **Structural Endurability** SE_{Vec} as a check: if a new structure fails to produce durable improvement, a low SE_{Vec} triggers rollback or consolidation, preventing catastrophic structural drift.

7.5 Scaling, Parallelism, and Practicality

At realistic scale, HTF is designed to keep the meta-controller small and cheap. The structural policy π_s operates on a low-dimensional feelings vector and runs on a **slower timescale** than the main forward passes of W_θ. Expensive structural operations (Spawn, Merge, Tool-Acquire) are rare and amortized over many prediction steps, with their cost explicitly penalized in $\mathcal{R}_S$.

This **temporal decoupling** preserves batch-friendliness: W_θ can be trained and evaluated in large batches, while V_ϕ aggregates batch statistics into feelings, and π_s updates the architecture intermittently. HTF thus complements, rather than replaces, large-scale world-model training.

7.6 Core Principle and Compressible Feelings

The core idea can be summarized as:

> Self-modifying intelligence requires a hierarchy of pain/reward signals to distinguish between local failure (patch) and structural insufficiency (upgrade).

We conjecture that some form of *Decoupling of Feeling* is necessary for robust structural triage. HTF provides one concrete instantiation and empirical evidence: when we fuse or remove the hierarchy, the agent either becomes structurally obese or structurally myopic. We find that the four channels (PD, HR, MF_c, $\mathrm{SE}_{\mathrm{Vec}}$) can be viewed as an engineered instantiation of a more minimal pair: an Epistemic feeling E for local efficiency/uncertainty, and a Teleological feeling T for long-term structural insufficiency.

7.7 Evidence of Qualitative Change

Our most salient behavioral signature is **autonomous self-upgrade**. In the HCLB sequence, HTF agents exhibit a clear pattern:

1. performance plateau and rising MF_c,
2. a discrete Tool-Acquire action,
3. an immediate jump to a new competence level,
4. a subsequent compression phase driven by HR.

Ablations confirm the necessity of the hierarchy:

- Removing the slow teleological loop yields structurally myopic agents—stuck at a low performance ceiling.
- Removing the fast epistemic loop yields competent but structurally obese agents that never compress back down.

HTF is the only agent that maintains both high competence and high structural efficiency, demonstrating successful structural triage.

7.8 Path to Real AGI and Alignment Considerations

An "adult" HTF-style system would comprise at least four components: a large multimodal W_θ, a learned feelings head V_ϕ, an explicit RL structural controller π_s, and a **persistent memory** Ψ of structural decisions and long-horizon goals (an "inner voice").

HTF is orthogonal to other major components such as tool use or RL in the environment: it acts as the architectural "glue" deciding when and how to grow, compress, and integrate these elements.

Granting explicit structural agency makes alignment both more challenging and more controllable. While a self-restructuring system can potentially route around constraints, HTF exposes a **concentrated leverage point**: the structural reward $\mathcal{R}_S$ and its penalties for unaligned complexity or resource overuse. Designing and auditing this structural objective is therefore a central open problem.

7.9 Generalization, Minimal Toy Worlds, and Existing Systems

The feelings are intended as highly reusable **meta-level signals** and should, in principle, transfer across environments that share the same structural pressures of growth, compression, and tool-acquisition. We provide early evidence of transfer across multiple variants of the HCLB sequence, but broader generalization (e.g., to visual or language domains) remains an open empirical question.

We also identify a minimal toy setting where all structural pressures appear: an environment in which the agent must (i) grow structure, (ii) compress near-duplicate solutions, and (iii) invoke an external resource it cannot internally simulate. Finally, we note that traces of HTF-like feelings already appear in large systems: confidence scores that trigger retrieval or code execution are primitive epistemic signals, and MoE load-balancing losses act as crude redundancy feelings. HTF unifies these functional signals into a single, explicit control architecture.

8 Conclusion

This paper proposed Hierarchical-Temporal Feelings (HTF) as a principle and architecture for structural self-management in world models. By decoupling fast epistemic feelings from slow teleological feelings, HTF enables an agent to distinguish between local patches and global upgrades, and to manage its own growth, compression, and tool use over a non-stationary lifetime.

We argued that the Autonomy–Architecture Problem—balancing competence, efficiency, and tool-acquisition under continual learning—requires such a hierarchy of feelings, and we introduced the Hierarchical Continual Learning Benchmark (HCLB) to illustrate the dilemma. HTF is the only agent that maintains both high competence and high structural efficiency, performing structural triage where flat and naive modular agents fail.

Looking forward, we view HTF not as a finished solution but as a compact lens on a larger design space. The central open questions lie in scaling the architecture to realistic multimodal world models, formalizing the structural reward, and understanding how decoupled feelings might emerge or be shaped in very large systems. If successful, HTF-like mechanisms could form part of the "structural glue" that allows future AGI systems to manage themselves.

Acknowledgements. I would like to thank Deb and Emo.