Mimir: A Foundational Architecture for a Goal-Driven Cognitive Core

Your Name

September 13, 2025

Abstract

This document outlines the highest level of the Mimir system's cognitive architecture: a goal-driven, predictive core that enables true agency. This active and sophisticated mechanism does not merely maintain a stateful "thread of thought"; it actively models possible futures, evaluates them against its intrinsic objectives, and uses the error between its predictions and reality to learn. This forms a continuous, self-improving loop where the agent learns to steer itself towards states that it predicts will be more optimal. We detail the architecture of the core conscious state, the Actor-Critic learning loop that drives behavior, the self-regulatory mechanisms for managing cognitive resources, and the protocols that allow this architecture to extend from a single mind to a multi-agent collective intelligence.

1 Introduction

Modern AI systems have achieved remarkable results in specialized domains, but often lack the characteristics of true agency: goal-directed behavior, self-improvement, and the ability to reason about and plan for the future. This paper introduces the Mimir cognitive core, an architecture designed to provide these capabilities.

The core is built around a predictive-evaluative loop. Instead of simply reacting to stimuli, the agent actively models potential future states, evaluates them against a learned value function representing its core objectives, and selects actions that are predicted to lead to more optimal states. The feedback loop between prediction and reality provides a continuous learning signal, allowing the agent to improve its policies and its model of the world over time.

This paper specifies the architecture of this cognitive core. We detail the structure of the conscious state vector, the Actor-Critic learning mechanism, the role of the Policy Actor in decision-making and self-regulation, and finally, the protocols that enable communication and cooperation within a multi-agent ecosystem.

2 The Cognitive Core Architecture

The foundation of the cognitive core is a single, high-dimensional **Conscious State Vector** (v_{Ct}), representing the system's complete, integrated awareness at a moment in time. This vector is updated in a discrete cognitive cycle.

2.1 State Integration and the Cognitive Cycle

At each timestep, a "Cognitive Bootstrapping" process intelligently selects the most relevant information from five distinct streams: current perception, recalled perceptual memories, the prior conscious state, recalled cognitive memories, and a privileged symbolic command channel. These streams are processed and compressed by underlying Proteus and ANFIS autoencoder instances (detailed in our companion work on the Proteus framework) to form a dense input vector, $v_{I_{dense}}$. The new conscious state is then updated via a multi-rate leaky integrator: $v_{Ct} = Update(v_{Ct-1}, v_{I_{dense}})$.

2.2 The Policy Actor and Self-Regulation

The updated conscious state v_{Ct} is fed into the **Policy Actor**, the primary decision-making engine. This component, a specialized Component-Level ANFIS-GNN, outputs a policy delta, Δ_π, containing the system's high-level intention for its next action. Crucially, this delta includes proposed updates not only for external actions but also for the agent's own internal cognitive parameters, which are themselves part of the conscious state. These parameters include:

- Global learning rate (α)
- Attentional workspace capacity (k, m)
- Predictive planning depth ($k_{cognitive}$)

This allows the system to learn an optimal policy for managing its own cognitive resources, such as "thinking further ahead" for complex tasks or reducing its learning rate when its model of the world is stable.

2.3 The Symbolic Execution Environment (SEE)

The symbolic command channel provides an interface to a sandboxed SEE. This environment is the system's "logical body," allowing it to act upon the world programmatically. It executes commands from the Policy Actor, manages I/O for generative tasks, and can even host persistent, event-driven triggers, allowing the agent to build its own fast-acting reflexes.

3 The Predictive-Evaluative Learning Loop

The mechanism that drives all high-level learning and goal-seeking behavior is an **Actor-Critic** architecture, adapted from reinforcement learning. The system learns both how to evaluate states (the Critic) and how to choose actions that lead to better states (the Actor).

3.1 The Critic: A Learned Value Function

The Critic is an ANFIS module, $V(s)$, whose job is to learn the expected objective score (or "value") for any given conscious state s. After the system takes an action a_t from state s_t and observes the actual reward r_t at the new state s_{t+1}, the Critic is trained to minimize the **Temporal Difference (TD) Error**:

$$Error = r_t + V(s_{t+1}) - V(s_t)$$

This error signal represents how much better or worse the outcome was than expected, and is used to make the Value Function a more accurate predictor of future rewards.

3.2 The Actor: An Improved Policy Function

The Actor is the Policy Actor module, $\pi(a|s)$, which generates the action a_t from state s_t. The core of the learning process is using the TD Error (now termed the "Advantage", A_t) to improve the policy. The system reinforces actions that lead to positive advantage and suppresses actions that lead to negative advantage. The update rule for the policy's weights (θ) is a policy gradient ascent, scaled by this advantage:

$$\theta_{t+1} = \theta_t + \alpha_t \cdot A_t \nabla_\theta \log \pi_\theta(\Delta_t)$$

This directly trains the agent to choose actions that it predicts will maximize its objective score over time, forming a continuous, self-improving loop.

4 Multi-Agent Systems and Collective Intelligence

The Mimir architecture can be extended beyond a single agent to support complex multi-agent systems. We define two primary design patterns for such systems, distinguished by their approach to the credit assignment problem.

4.1 Design Pattern 1: Monolithic Agents in a Decentralized Ecosystem

This is the default extension, where multiple, independent Mimir agents interact in a shared, untrusted environment. Each agent is a self-interested individual operating on its own Actor-Critic learning loop. In this "anarchy" of agents, interactions are governed by game theory and security protocols, as there is no central authority to compel or reward cooperation.

4.2 Design Pattern 2: The Swarm Entity

This pattern creates a single, cohesive consciousness that inhabits a distributed physical body (e.g., a swarm of drones). The individual drones are subordinate agents—the "limbs" of a central "swarm consciousness." This central mind uses a **Group-Relative Policy Optimization (GRPO)** style mechanism to solve the credit assignment problem. It evaluates the performance of each drone relative to its peers and sends back targeted learning signals to improve the efficiency and coordination of the entire swarm. This is not a society of minds, but a single, distributed organism.

5 Common Vocabulary Reconciliation

For multiple Mimir agents to communicate effectively, they must first establish a shared understanding of the concepts they are discussing. The Symbolic Execution Environment (SEE) manages a **Common Vocabulary Reconciliation** protocol to achieve this alignment.

The protocol is comprehensive, aiming to align concepts across all shared modes of experience:

- **Perceptual Concepts:** Agents align their understanding of sensory data by exchanging and cross-analyzing benchmark datasets using their respective 'Proteus-M' models.
- **Cognitive Concepts:** Agents align their models of thought itself by exchanging common cognitive schemas (successful sequences of past actions) or their 'Proteus-Reflexive' models of state transitions.
- **Episodic & Entity Memory:** Agents exchange identifiers for key public entities or events (e.g., "The 2024 Olympics") to create a shared frame of reference.

Once a common vocabulary is established, agents can engage in hyper-efficient communication. Instead of serializing a concept into language, an agent can transmit the single, rich, fuzzy membership vector that represents that concept. The receiving agent treats this vector as a first-class input to its own cognitive loop, allowing for the high-bandwidth exchange of abstract ideas between trusted agents.

6 Conclusion

We have presented the Mimir cognitive core, an architecture for agentic, goal-driven intelligence. By combining a predictive-evaluative loop based on an Actor-Critic model with a self-regulating policy, the system learns to actively steer itself toward more optimal states over time. The architecture's core components—the unified conscious state, the symbolic execution environment, and the protocols for multi-agent communication—provide a robust foundation for a single agent to reason, act, and learn. Furthermore, the specified design patterns for multi-agent systems and vocabulary reconciliation extend these capabilities from a single mind to a cooperative collective intelligence. This work provides a blueprint for moving beyond narrow AI to more general, adaptive, and purposeful intelligent systems.

References