Temporal Renormalization: Solving the Horizon Explosion via Multiscale Diffusion Planning

Anonymous

January 2, 2026

Abstract

Current artificial intelligence architectures operate on a "flat" timeline, predicting states sequentially ($t \to t+1$). This results in a "Horizon Explosion," where error accumulates exponentially with sequence length ($Error^T$), rendering long-horizon reasoning computationally intractable. We propose a new computational primitive: **Temporal Renormalization**. By replacing flat planning with a Multiscale Renormalization Group architecture, we treat time as a fractal, planning a hierarchy of constraints rather than a sequence of actions. This reduces planning complexity from linear $O(T)$ to logarithmic $O(\log T)$. We introduce the **Fractal Diffusion** architecture (SWE-DP), utilizing decoupled energy resonance between layers to prevent vanishing gradients. We release this architecture as a proposal for the community to validate against the "10k Step Challenge."

1 The Problem: The Horizon Explosion

The fundamental bottleneck preventing Artificial General Intelligence (AGI) is not a lack of data or parameters, but a limit in computational complexity regarding time. To solve a task, an agent must minimize the Action Functional S over a trajectory τ:

$$S[\tau] = \int_0^T \mathcal{L}(s_t, a_t)\, dt \tag{1}$$

For a task lasting one hour at atomic resolution (100ms), $T \approx 36,000$. In flat architectures (Transformers, standard RL), the probability of a successful plan scales as $P_{success} \propto (1-\epsilon)^T$. For any non-trivial ϵ, this probability approaches zero rapidly, a phenomenon we term the **Horizon Explosion**.

2 The Solution: Temporal Renormalization

To solve the Horizon Explosion, we apply the **Renormalization Group (RG)** transformation to the temporal dimension. We define a hierarchy of coarse-grained planning layers $k \in \{0, 1, ..., K\}$, where each layer operates at a timescale $\Delta t_k = \beta^k \Delta t_0$.

2.1 Mathematical Formulation

Instead of solving the global optimization $S[\tau]$ directly, we solve a cascade of constrained optimizations. The effective action at scale k is given by the renormalization operator $\mathcal{R}$:

$$S_{eff}^{(k)} = \mathcal{R}[S^{(k-1)}] + \lambda_{const} \|\tau^{(k)} - \text{Proj}(\tau^{(k+1)})\|^2 \tag{2}$$

This decomposes the global optimization into local sub-problems, reducing complexity from $O(T)$ to $O(\log T)$.

2.2 The Fractal Stack (CEO-Manager-Worker)

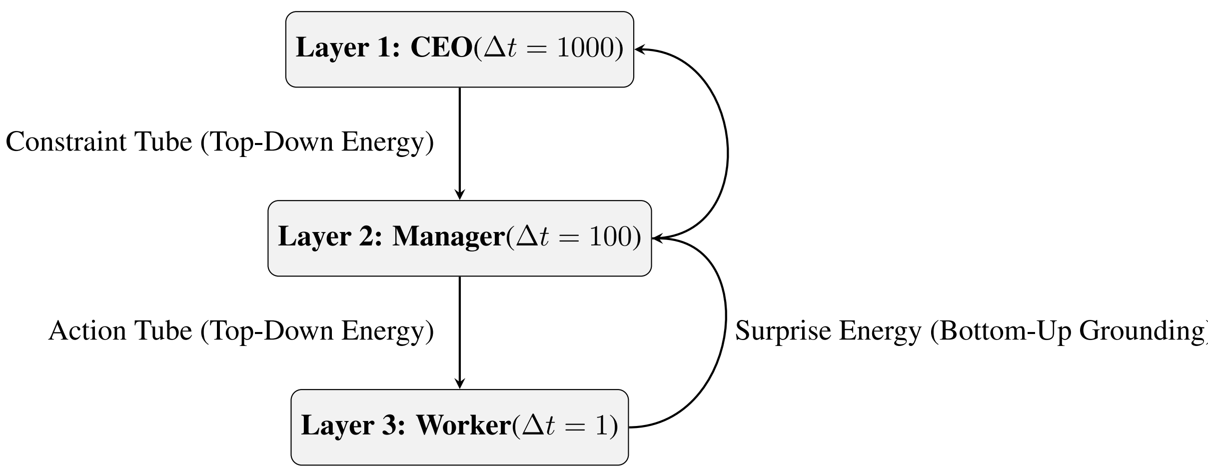

Figure 1: The Fractal Diffusion Architecture. Signals flow bi-directionally: Control flows down (Left) via Constraints, Reality flows up (Right) via Surprise.

The architecture consists of three distinct layers:

1. **Layer 1: The CEO (Coarse-Grained).** Solves the Lagrangian for the entire episode using abstract latents. Outputs a "Constraint Tube" defining valid states.
2. **Layer 2: The Manager (Medium-Grained).** Finds a path that satisfies the CEO's tube while navigating local obstacles.
3. **Layer 3: The Worker (Fine-Grained).** Executes immediate physics using Consistency Distillation for real-time latency (< 10ms).

3 The Physics: Decoupled Optimization

A critical failure mode of hierarchical systems is the vanishing gradient problem. We propose **Decoupled Optimization**. Layers do not communicate via end-to-end gradients, but via **Energy Potentials**:

$$E_{total} = \underbrace{E_{goal}(\tau^{(k)})}_{\text{Task}} + \underbrace{\lambda_{down} E_{constraint}(\tau^{(k)}|\tau^{(k+1)})}_{\text{Control}} + \underbrace{\lambda_{up} E_{surprise}(\tau^{(k)}|\tau^{(k-1)})}_{\text{Grounding}} \quad (3)$$

This creates a resonant circuit where the plan is the equilibrium state of the multi-layer energy system.

4 Engineering Stack (SWE-DP)

We define the reference implementation stack, termed **SWE-DP**:

- **World Model:** Mamba or S4 (State Space Models) for stable simulation.
- **Planner:** Diffusion Transformer (DiT), treating trajectories as images.
- **Accelerator:** Consistency Distillation on the Worker layer.

5 Validation: The 10k Step Challenge

We propose the **10k Step Challenge** as the standard MVP for AGI reasoning. **Task:** A sparse-reward navigation task (e.g., MiniGrid) requiring 10,000 atomic steps.

Go/No-Go Metrics:

1. **Wall-Clock Speed:** Outperform MCTS at Horizon=50.
2. **Context Stability:** Maintain goal context after 5,000 steps.
3. **Adaptation:** Re-plan in < 100 steps when dynamics change.

6 Call to Action

We invite the research community to fork this concept. We are not building a chatbot. We are building a Navigational Physics Engine. If the hierarchy holds for physics, it will scale to language.