DMWM: Dual-Mind World Model with Long-Term Imagination

Lingyi Wang[1]* **Rashed Shelim**[1] **Walid Saad**[1] **Naren Ramakrishnan**[2]
[1] Department of Electrical and Computer Engineering, Virginia Tech, USA
[2] Department of Computer Science, Virginia Tech, USA
Emails: {lingyiwang, rasheds, walids, naren}@vt.edu

Abstract

Imagination in world models is crucial for enabling agents to learn long-horizon policies in a sample-efficient manner. Existing recurrent state-space model (RSSM)-based world models depend on single-step statistical inference to capture the environment dynamics, and, hence, they cannot effectively perform long-term imagination tasks due to the accumulation of prediction errors. Inspired by the dual-process theory of human cognition, we propose a novel dual-mind world model (DMWM) framework that integrates logical reasoning to enable imagination with logical consistency. DMWM is composed of two components: an RSSM-based System 1 (RSSM-S1) component that handles state transitions in an intuitive manner and a logic-integrated neural network-based System 2 (LINN-S2) component that guides the imagination process through hierarchical deep logical reasoning. The inter-system feedback mechanism is designed to ensure that the imagination process follows the logical rules of the real environment. The proposed framework is evaluated on benchmark tasks that require long-term planning from the DMControl suite and the robotic platforms. Extensive experimental results demonstrate that the proposed framework yields significant improvements in terms of logical coherence, trial efficiency, data efficiency and long-term imagination over the state-of-the-art world models. The code is available at https://github.com/news-vt/DMWM.

1 Introduction

Imagination is a core capability of world models that allows agents to predict and plan effectively within internal virtual environments by using real-world knowledge [1, 2, 3, 4]. By predicting future scenarios in latent spaces, agents can evaluate potential outcomes without frequent real-world interactions thereby significantly improving data efficiency and minimizing trial-and-error costs. For complex tasks requiring long-term planning, imagination capabilities allow world models to evaluate the long-term consequences of diverse strategies and identify the optimal action plans. Consequently, the effectiveness of model-based decision-making approaches, such as model-based reinforcement learning (RL) [5, 6, 7, 8] and model predictive control (MPC) [9, 10, 11], can heavily depend on the quality of their imagination abilities.

One of the most widely used frameworks for world models is the so-called recurrent state-space model (RSSM) [2, 9, 12] and its variants [13, 14, 15, 16] that combine deterministic recurrent structures with stochastic latent variables to model environmental dynamics in a compact latent space. By doing so, RSSM models can capture the sequential dependencies and uncertainty of their target environment. However, RSSM cannot provide reliable, long-term predictions over extended imagination horizons due to the accumulation of prediction errors and the limitations of statistical inference [17, 18, 19]. In particular, although existing RSSM-based solutions can generate accurate short-term predictions

*Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

in a single-step manner, small errors inevitably propagate over longer time horizons [20], gradually amplifying over each step and resulting in significant deviations between the imagined and actual states. Moreover, RSSM schemes often optimize state-space representations through reconstruction loss or regression. This approach can lead to overfitting to observed patterns and cannot properly capture latent dynamics [18, 20, 21], particularly in complex, dynamic environments.

Several models [17, 18, 20, 22, 23, 24] have been proposed for long-term planning. For instance, trajectory-based models [20, 23] learn long-term dynamics by directly predicting future states to reduce compounding errors compared to single-step prediction models. Latent space-enhanced models [17, 18] incorporate abstracted long-term information, such as high-level goals and global knowledge, to mitigate error propagation in each single-step prediction. However, all of these approaches [17, 18, 20, 22, 23] still rely on statistical inference, which inevitably leads to prediction error accumulation and drift over long horizons. By skipping intermediate reasoning steps, these approaches [17, 18, 24] lose the logical consistency and interpretability required for complex, high-precision planning. Hence, existing approaches cannot provide robust and reliable imagination over an extended horizon size.

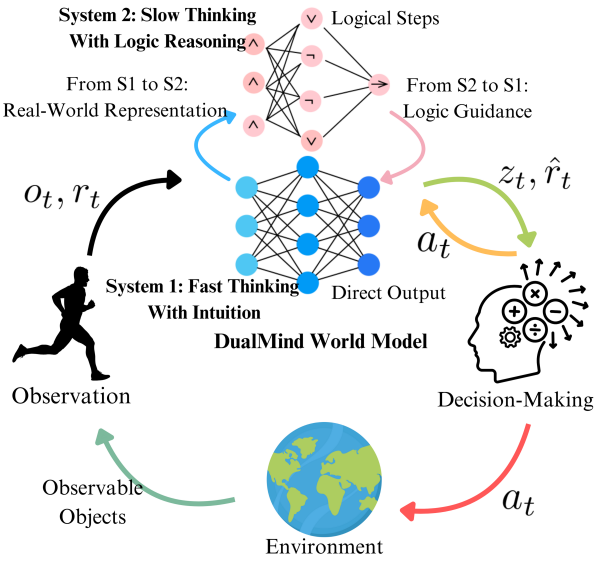

Figure 1: The proposed framework for DMWM.

Motivated by the challenges of long-term imagination and the limitations of existing long-term planning approaches, the main contribution of this paper is a novel dual-mind world model (DMWM) framework based on the dual-process theory of human cognition [25, 26, 27]. The proposed DMWM framework can achieve reliable and efficient imagination by synergistically combining the complementary strengths of System 1 and System 2 learning processes [28, 29]. By designing this framework shown in Figure 1, we make the following key contributions:

- We propose DMWM, a novel world model framework that integrates the dual-process theory of human cognition (System 1 and System 2) to endow agents with robust, long-term imagination capabilities driven by both data and logical reasoning. Particularly, based on the RSSM-based System 1 (RSSM-S1) component that predicts the state transitions in a fast, intuitive-driven manner, we further propose logic-integrated neural network-based System 2 (LINN-S2) component that is used, for the first time, to guide imagination through logical reasoning at a higher level.
- In LINN-S2, we introduce logical regularization rules to conduct logical reasoning within the state and action spaces by using operations such as $\wedge$, $\vee$, $\neg$ and $\rightarrow$. The logical rules allows the logical consistency and interpretability of the world model. Additionally, we propose a new recursive logic reasoning framework that extends local reasoning into globally consistent long-term planning, enabling the modeling of logical sequence dependencies in complex tasks.
- We design an effective inter-system feedback mechanism. In particular, LINN-S2 provides logical constraints to guide RSSM-S1 so as to ensure that predicted sequences are consistent with domain-specific logical rules. For the feedback based on real-world observations and latent representations from RSSM-S1, it updates the domain-specific logic of LINN-S2, thereby allowing dynamic refinement and adaptation. This novel inter-system feedback mechanism can thus allow the human-like, dual-process cognitive abilities for agents.
- We evaluate the proposed DMWM with actor-critic based RL and MPC in extensive experiments including DMControl and robotic tasks. Simulation results demonstrate that DMWM is able to respectively provide 14.3%, 5.5-fold, 32% and 120% improvement in logic consistency, trial efficiency, data efficiency and reliable imagination over an extended horizon size compared to baselines in complex tasks.

2 Proposed DMWM Framework

In this section, we introduce the proposed DMWM framework inspired by the dual-process theory of human cognition. DMWM consists of RSSM-S1 and LINN-S2. First, we introduce RSSM-S1, which builds upon the RSSM architecture to learn the environment dynamics in a latent space and perform fast, intuitive state representations and predictions. Next, we introduce a novel LINN-S2 to capture the intricate logical relationships between the state space and the action space. By employing a hierarchical deep reasoning framework, LINN-S2 facilitates structured reasoning and enforces logical consistency over extended horizons. Finally, we explain the proposed inter-system feedback mechanism. The pipeline of the proposed DMWM framework is shown in Figure 1.

2.1 RSSM-based System 1

The RSSM-S1 component is based on DreamerV3 [7], which is represented by

$$\begin{aligned}
&\text{Deterministic State:} && h_t = f_\varphi\left(h_{t-1}, z_{t-1}, a_{t-1}\right), \\
&\text{Encoder:} && z_t \sim q_\varphi\left(z_t \mid h_t, o_t\right), \\
&\text{Stochastic State:} && \hat{z}_t \sim p_\varphi\left(\hat{z}_t \mid h_t\right), \\
&\text{Reward Predictor:} && \hat{r}_t \sim p_\varphi\left(\hat{r}_t \mid h_t, z_t\right), \\
&\text{Decoder:} && \hat{o}_t \sim p_\varphi\left(\hat{o}_t \mid h_t, z_t\right),
\end{aligned} \tag{1}$$

with the deterministic state h_t, observation o_t, predicted observation $\hat{o}_t$, stochastic state z_t, predicted stochastic state $\hat{z}_t$, action a_t and the predicted reward $\hat{r}_t$ at time step t. RSSM-S1 achieves an effective balance between deterministic and stochastic states and enables efficient data-driven prediction similar to the intuitive and automatic processes of System 1. The deterministic state captures data patterns, and the stochastic state models inherent uncertainty and dynamics for complex environments.

System 1 loss. RSSM-S1 is optimized by using the loss function of DreamerV3 [7]:

$$\begin{aligned}
&\mathcal{L}_{\text{S1}}\left(\varphi\right) = \mathcal{L}_{\text{pred}}\left(\varphi\right) + \varpi_{\text{dyn}}\,\mathcal{L}_{\text{dyn}}\left(\varphi\right) + \varpi_{\text{rep}}\,\mathcal{L}_{\text{rep}}\left(\varphi\right), \\
&\mathcal{L}_{\text{pred}}\left(\varphi\right) = -\ln p_\varphi\left(o_t \mid z_t, h_t\right) - \ln p_\varphi\left(r_t \mid z_t, h_t\right), \\
&\mathcal{L}_{\text{dyn}}\left(\varphi\right) = \mathrm{KL}\left[\mathrm{sg}\left(q_\varphi\left(z_t \mid h_t, o_t\right)\right) \| p_\varphi\left(z_t \mid h_t\right)\right], \\
&\mathcal{L}_{\text{rep}}\left(\varphi\right) = \mathrm{KL}\left[q_\varphi\left(z_t \mid h_t, o_t\right) \| \,\mathrm{sg}\left(p_\varphi\left(z_t \mid h_t\right)\right)\right],
\end{aligned} \tag{2}$$

where ϖ_{dyn} and ϖ_{rep} are respectively the weight factors of the dynamic loss $\mathcal{L}_{\text{dyn}}$ and the representation loss $\mathcal{L}_{\text{rep}}$, and $\mathrm{sg}(\cdot)$ represents the stop-gradient operator.

System 1 limitations. The RSSM framework can be viewed as a computational counterpart of the System 1 component in the dual-process theory of cognition [25] characterized by fast, intuitive-driven reasoning. However, it is inherently constrained by two critical limitations: *the absence of explicit logical reasoning* and *the inability to maintain coherence over extended temporal horizons*. While RSSM-S1 enables efficient and immediate response through pattern recognition and intuitive prediction, these strengths come at the cost of enforcing logical consistency or inferring causal relationships in complex scenarios. Moreover, RSSM-S1's focus on short-term processing limits its ability to integrate information across long periods, resulting in fragmented or inconsistent predictions in tasks that demand global planning or long-term foresight. These inherent limitations demonstrate the need to complement the capabilities of RSSM-S1 with mechanisms of structured reasoning and sustained temporal coherence, similar to System 2, to construct a more robust, cognitive world model framework.

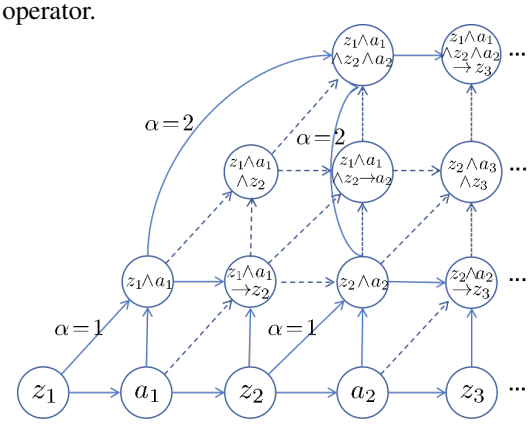

Figure 2: Logic reasoning for LINN-S2.

2.2 Logic-integrated neural network-based System 2

Next, we introduce the LINN-S2 component whose deep logical reasoning is shown in Figure 2. In LINN-S2, the states and actions are encoded as logic vector inputs, enabling logical deduction through operations such as negation ($\neg$), conjunction ($\wedge$), disjunction ($\vee$), and implication ($\to$). To establish these logical operations, the LINN framework [30] serves as the foundational module for

System 2. We propose a novel hierarchical logical reasoning framework and extended regularization rules for implication operations to capture and reason about structural relationships between the state and action spaces, thus endowing the world model with logical inference capability.

Neural modules for basic logic operations. In logical reasoning for world models, it is essential to uncover structural information and semantic relationships across different embedding spaces, specifically the action space and the state space. In [30], the authors proposed a straightforward concatenation-based method for LINN. However, the approach of [30] is limited by the fact that it limits that input vectors are from the same source space. Moreover, the straightforward concatenation-based method overlooks the semantic disparities between states and actions, risking the loss of critical information and failing to capture complex cross-space logical relationships.

To address the need for cross-space logical reasoning in world models, we propose to explore the action embeddings and apply the Kronecker product for cross-space feature alignment, which preserves logic integrity and captures second-order relationships. The (imagined) state logical embedding v and the state logical embedding m are obtained with multilayer perception (MLP) by

$$v = \boldsymbol{W}_2^s f\left(\boldsymbol{W}_1^s(z) + \boldsymbol{b}_w^s\right), \quad m = \boldsymbol{W}_2^a f\left(\boldsymbol{W}_1^a(z \oplus a) + \boldsymbol{b}_w^a\right), \tag{3}$$

where $\oplus$ is the operation of vector concatenation, and $z \oplus a$ aims to capture the action logic within the state context. The formulations of operations $(\mathrm{AND}, \mathrm{OR}, \mathrm{NOT})$ are respectively given by

$$\mathrm{AND}(v, m) = \boldsymbol{W}_2^a f\left(\boldsymbol{W}_1^d(v \oplus m) + \boldsymbol{b}_w^d\right) + \mathrm{Conv2D}(v \otimes m, \boldsymbol{K}^d) + \boldsymbol{b}_k^d, \tag{4}$$

$$\mathrm{OR}(v, m) = \boldsymbol{W}_2^o f\left(\boldsymbol{W}_1^o(v \oplus m) + \boldsymbol{b}_w^o\right) + \mathrm{Conv2D}(v \otimes m, \boldsymbol{K}^o) + \boldsymbol{b}_k^o, \tag{5}$$

$$\mathrm{NOT}(v) = v + \boldsymbol{W}_2^n f\left(\boldsymbol{W}_1^n v + \boldsymbol{b}_w^n\right), \tag{6}$$

where $v \in \mathbb{R}^d$, $m \in \mathbb{R}^d$, $\boldsymbol{W}_1^l \in \mathbb{R}^{d \times 2d}$, $\boldsymbol{W}_2^l \in \mathbb{R}^{d \times d}$, $\boldsymbol{b}_w^l \in \mathbb{R}^d$, $\boldsymbol{b}_k^l \in \mathbb{R}^d$ are the parameters of the logical neural networks, $l \in \{d, o, n\}$, $\mathrm{Conv2D}(\cdot)$ represents the convolution neural network, $\boldsymbol{K}$ is the convolution core, $\otimes$ is the operation of Kronecker product, and $f(\cdot)$ is the activation function.

Logical operations capture intricate logical correlations that cannot be fully encapsulated by the geometric properties of the embedding space. For instance, the logical independence in $\mathrm{NOT}(v)$, reflected in the relationship between v and $\neg v$, does not correspond precisely to vector orthogonality. Logical operations are governed by logical axioms (e.g., identity, annihilator, and Complement), which impose algebraic constraints that transcend purely geometric interpretations, which are introduced in the logic regularization part.

Implication relationship for state reasoning. Based on the basic logical operations (AND, OR, NOT), we implement the implication operation ($p_\varphi \to q_\varphi$) to enable logical reasoning within the state and action logical embedding spaces. The implication operation is critical for assessing the rationality of a predicted state given the current imagination and action embeddings, which is derived through the equivalence relationship $p \to q \Longleftrightarrow \neg p \vee q$. Hence, the operation IMPLY for $v \to m$ is realized based on the fundamental operations of negation ($\neg$) and conjunction ($\wedge$), represented by

$$\mathrm{IMPLY}(v, m) = \mathrm{OR}(\mathrm{NOT}(v), m). \tag{7}$$

Table 1: Partital Logical Regularizers and Rules for Implication $\to$

Logical Rule	**Equation**	**Logic Regularizer** r_i
Identity	$w \to \mathbf{T} = \mathbf{T}$	$r_{11} = \sum_{w \in W} 1 - \mathrm{Sim}(\mathrm{OR}(\mathrm{NOT}(w), \mathbf{T}), \mathbf{T})$
Annihilator	$w \to \mathbf{F} = \neg w$	$r_{12} = \sum_{w \in W} 1 - \mathrm{Sim}(\mathrm{OR}(\mathrm{NOT}(w), \mathbf{F}), \mathrm{NOT}(w))$
Idempotence	$w \to w = \mathbf{T}$	$r_{13} = \sum_{w \in W} 1 - \mathrm{Sim}(\mathrm{OR}(\mathrm{NOT}(w), w), \mathbf{T})$
Complement	$w \to \neg w \equiv \neg w$	$r_{14} = \sum_{w \in W} 1 - \mathrm{Sim}(\mathrm{OR}(\mathrm{NOT}(w), \mathrm{NOT}(w)), \mathrm{NOT}(w))$

Logical regularizations. To ensure that neural modules accurately perform logical operations, the work in [31] introduced logic regularizers $\{r_i\}$ that enforce adherence to fundamental logical rules, thereby constraining module behavior. While neural networks can implicitly learn logical operations from data, the explicit incorporation of logical constraints improves model consistency, interpretability, and robustness while maintaining efficient neural computation. Derived from principles of AND, OR, and NOT, logic regularizers establish a unified inference framework with fundamental and advanced operations. This approach aligns model outputs with human-understandable reasoning, bridging neural networks and formal logic, and enabling effective learning, representation, and inference of logical formulas for interpretable reasoning in world models.

We extend the foundational framework of logical regularization in [30] by introducing implication regularizers to address advanced logical rules. As summarized in Table 1, the implication rules refine basic logical structures to handle intricate constructs. By explicitly encoding principles such as *contraposition* ($w \to v \equiv \neg v \to \neg w$) and *complement* ($w \to \neg w \equiv \neg w$), LINN-S2 improves its capacity to manage compound logical expressions while ensuring consistency across implication operations. The extended logical regularizers for IMPLY (r_{11}-r_{14}) are given in Table 1, combined with the basic logical regularizers for AND, OR, and NOT (r_1-r_{10}), collectively define the regularization loss function $\mathcal{L}_{\text{reg}} = \frac{1}{N_r}\sum_i r_i$, where r_i represents individual logical regularizers and N_r presents the total number of the regularizers. The complete table of the logical regularizers is given in Appendix G.

Proposed symbolic representation of hierarchical logical reasoning. We now present the symbolic representation of LINN-S2 for deep logical reasoning. Specifically, the logical reasoning process is formalized by using symbolic logic to represent logical relationships in sequences. By leveraging logical operations ($\wedge, \to$), LINN-S2 constructs a hierarchical logical reasoning framework that facilitates systematic and interpretable reasoning over sequential dependencies. The framework of the hierarchical logical reasoning is illustrated in Figure 2, which includes three key procedures given as follows.

1) Local logical composition: Each (imagined) state logical embedding v_t and action logical embedding m_t are combined by using the logical conjunction ($\wedge$) to capture the intrinsic logic as

$$c_t : v_t \wedge m_t, \quad \forall t < T. \tag{8}$$

The composition c_t establishes localized logical features based on the existing v_t and m_t.

2) Recursive implication reasoning: To ensure logical consistency across (imagined) states, each composition (c_t) undergoes an implication operation ($\to$) that aligns the logical information with the subsequent (imagined) state

$$\phi_t : c_t \to z_{t+1}, \quad \forall t < T. \tag{9}$$

The recursive formulation (9) encodes sequential dependencies and enforces consistency in predictive state transitions across the hierarchy. However, ϕ_t provides only single-step logical inference, and it lacks reasoning depth and comprehensiveness. To address this limitation, we propose incorporating deterministic historical information into the logical reasoning framework. The deep recursive implication reasoning is represented by

$$\phi_t^\alpha : c_{t-\alpha} \cdots \wedge c_{t-1} \wedge c_t \to z_{t+1}, \forall \alpha < t < T, \tag{10}$$

where α is the inference depth. In complex scenarios with long-term dependencies, the model utilizes logical relationships from past states to ensure coherent reasoning. Capturing logic across time steps retains historical information, strengthens sequential dependencies and enhances global consistency. This prevents inference bias from information loss during multi-step reasoning and ensures the reliability of long-term imagination.

3) Global logical chain: The global reasoning process integrates local consistency and recursive reasoning into a unified global logical chain L_T within the time period T

$$L_T^\alpha = \phi_1^\alpha \wedge \phi_2^\alpha \wedge \phi_3^\alpha \cdots \phi_{T-1}^\alpha \to \mathbf{T}, \tag{11}$$

where $\mathbf{T}$ represents the consistency condition, ensuring that the (imagined) state sequences align with the reasoning objectives. This formulation consolidates local information into a globally consistent reasoning framework.

In this way, LINN-S2 enhances the formal interpretation and mathematical rigor of world models along with several key features. First, the hierarchical structure encodes deep logical reasoning as a layered process, which allows logical consistency between state and action representations. Second, logical composition leverages logical operations for binding states and actions in a logical manner. This composition facilitates formal logic principles and enables modular, interpretable reasoning. Finally, the recursive implication ensures robustness and interpretability for long-term imagination.

System 2 loss. The loss function of the logic reasoning with inference depth α is given by

$$\mathcal{L}_{\text{log}}^\alpha = \frac{1}{T-1}\sum_t \text{Sim}(\phi_t^\alpha, \mathbf{T}) - \text{Sim}(\phi_t^\alpha, \mathbf{F}) = \frac{1}{T-1}\sum_t \text{Sim}(\phi_t^\alpha, \mathbf{T}) - \text{Sim}(\phi_t^\alpha, \text{NOT}(\mathbf{T})), \tag{12}$$

where the function Sim is the logic similarity metric that takes value between 0 and 1. In practice, $\mathbf{T}$ is a randomized fixed vertor, and $\mathbf{F}$ is obtained by $\mathrm{NOT}(\mathbf{T})$. We use the cosine similarity since the logical information of action and state spaces has been aligned in the logical vector space, given by

$$\mathrm{Sim}(v, m) = \sigma\left(\kappa(v \cdot m)/(\|v\|\|m\|)\right). \tag{13}$$

To ensure the logical consistency of basic operations of AND and OR by order-independence, i.e., $v \wedge m = m \wedge v$ and $v \vee m = m \vee v$, the inputs of AND and OR are randomly disrupted. Moreover, the ℓ_2-regularization term $\mathcal{L}_v$ [30] is employed to prevent the vector lengths from exploding, which could otherwise lead to trivial solutions (e.g., logical rules becoming ineffective) during optimization, and constraint on the model parameters to mitigate the risk of overfitting, represented by

$$\mathcal{L}_{\ell_2} = \sum_{v \in \mathcal{V}} \|v\|_F^2 + \sum_{m \in \mathcal{M}} \|m\|_F^2 + \sum_{w \in \mathcal{W}} \|w\|_F^2, \tag{14}$$

where $\mathcal{W}$ is the model parameter of LINN-S2. The loss function of System 2 is expressed as

$$\mathcal{L}_{\mathrm{S2}}(w) = \sum_{\alpha=0}^{\Lambda} \mathcal{L}_{\log}^{\alpha} + \beta_{\mathrm{reg}} \mathcal{L}_{\mathrm{reg}} + \beta_{\ell_2} \mathcal{L}_{\ell_2}, \tag{15}$$

where β_{reg} and β_{ℓ_2} represents the weight factors, and Λ denotes the maximum reasoning depth.

2.3 Inter-System Feedback Mechanism

Next, we develop a novel inter-system feedback mechanism to enable communications of observation signals and logical signals between System 1 and System 2.

Feedback from S1 to S2. During the real-environment interactions, LINN-S2 updates the domain-specific logical relationships by using the actual state transitions from RSSM-S1. Particularly, the state-action sequence $\{z_t, a_t, z_{t+1}\}_{t=0}^{T}$ captured by RSSM-S1 based on observations $\{o_t\}_{t=0}^{T}$ serves as labeled data, which is fed into LINN-S2 with the objective of minimizing the inference loss (12).

Feedback from S2 to S1. To embed LINN-S2-inspired logical reasoning into RSSM-S1, we propose a logic feedback that utilizes LINN-S2's logical consistency mechanisms to guide RSSM-S1. By unifying high-level logical structures with low-level representations, the proposed approach fosters a more robust and coherent world model for imagination. We particularly introduce the logical rules and rederive the variational evidence lower bound (ELBO) [5] with a logic inference term by

$$p_\varphi(o_{1:T}, z_{1:T} \mid a_{1:T}) = \prod_{t=1}^{T} p_\varphi(o_t \mid z_t) \tilde{p}_\phi(z_t \mid z_{t-1}, a_{t-1}), \tag{16}$$

where $\tilde{p}_\phi(z_t \mid z_{t-1}, a_{t-1}) \propto p_\varphi(z_t \mid z_{t-1}, a_{t-1}) \cdot \mathcal{C}(\phi(z_t, z_{t-1}, a_{t-1}))$ and $\mathcal{C}(\phi) = \mathrm{Sim}(\phi, \mathbf{T})$ is the logical consistency function that measures whether the logical rule ϕ is satisfied by imagination. The logical ELBO of the observation loss can be expressed as (Derivation in Appendix C)

$$\begin{aligned} \ln p_\varphi\left(o_{1:T} \mid a_{1:T}\right) = \int p_\varphi(o_{1:T}, z_{1:T} \mid a_{1:T})\, dz_{1:T} \geq \sum_{t=1}^{T} \Big(& \underbrace{\mathbb{E}_{q_1}\left[\ln p_\varphi\left(o_t \mid z_t\right)\right]}_{\text{Decoding}} \\ + \underbrace{\mathbb{E}_{q_1}\left[\ln \mathcal{C}(\phi(z_t, z_{t-1}, a_{t-1}))\right]}_{\text{Logic Inference}} & - \underbrace{\mathbb{E}_{q_2}\left[\mathrm{KL}\left[q_\varphi\left(z_t \mid o_{\leq t}, a_{<t}\right) \| p_\varphi\left(z_t \mid z_{t-1}, a_{t-1}\right)\right]\right]}_{\text{Prediction}} \Big). \end{aligned} \tag{17}$$

where $q_1 = q_\varphi\left(z_t \mid o_{\leq t}, a_{<t}\right)$, $q_2 = q_\varphi\left(z_{t-1} \mid o_{\leq t-1}, a_{<t-1}\right)$, and the state priors can be approximately obtained by past observations and actions $q_\varphi(z_{1:T} \mid o_{1:T}, a_{1:T}) = \prod_{t=1}^{T} q_\varphi(z_t \mid h_t, o_t)$.

With the proposed inter-system feedback, DMWM enables the agents to think in a human-like, dual-process cognitive way for more robust, reliable and long-term imagination.

3 Experimental Results and Analysis

In this section, we conduct extensive experiments to address the following key questions: (a) Can our model effectively capture logical relationships in dynamic environments?, (b) Does enhanced logical consistency enable our model to achieve higher task rewards under limited environment trials and data?, and (c) Over an extended horizon, can our model generate reliable long-term imagination?.

Experimental setup. The training environments consist of 20 continuous control tasks from DeepMind control (DMC) suite, 4 robotic tasks from ManiSkill2 platform, and 4 robotic tasks from

MyoSuite platform. We evaluate DMWM using two model-based decision-making approaches: An actor-critic reinforcement learning method and a gradient-based model predictive control (Grad-MPC) approach [11], referred to as DMWM-AC and DMWM-GD, respectively. Training details for DMWM-AC and DMWM-GD are provided in Appendix D. For comparison purposes and benchmarking, we include DreamerV3 [7], Dreamer-enabled Grad-MPC [11], and TD-MPC2 [13] as baselines. To highlight the limitations of using a single RSSM for world cognition, we compare our method against two state-of-the-art RSSM variants: Hieros [3] and HRSSM [15]. HRSSM improves representation robustness via masking and bisimulation to mitigate visual noise interference, while Hieros enhances long-term modeling and exploration efficiency through S5WM and hierarchical strategies. Details on environment and model settings are provided in Appendix E.

Logical consistency. Figure 3 shows the logical correlations between individual state-action pairs ($s_i \wedge a_j$) and the target state (s_{30}). The diagonal shows strong correlations for one-step pairs $s_i \wedge a_i \rightarrow s_{30}$, which is the key path of localized reasoning. The off-diagonal elements show interdependency across different state-action pairs ($s_i \wedge a_j, i \neq j$) that propagate global logical information. The observed patterns highlight the need for deep logical reasoning to capture both short-term and long-term logical dependencies.

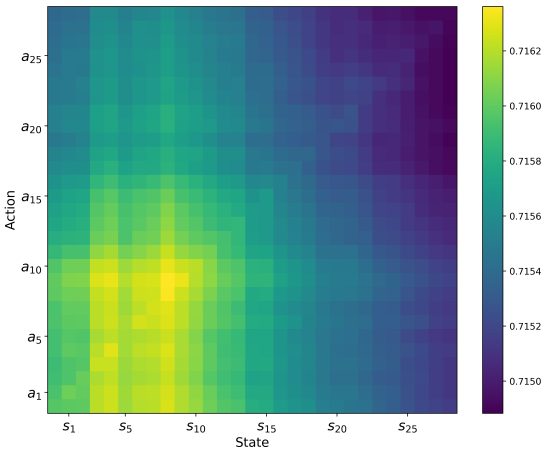

Figure 3: Heatmap of deep logic correlations for sequential imagination $s_0 \wedge a_0 \wedge ...s_{29} \wedge a_{29} \rightarrow s_{30}$ with reasoning depth $\alpha = 30$. The horizontal axis indicates the past states s_i and the vertical axis indicates past actions a_j. The color of points represents the logical strength of long-term state-action pairs.

Table 2 compares the logical consistency of the proposed DMWM against various baselines on DMC tasks. Logical consistency data for 20 tasks with different horizon sizes is provided in Appendix H.1. The proposed DMWM achieves state-of-the-art logical consistency in both mean logic loss and stability across all DMC environments. DMWM respectively achieves 14.3%, 2.6% and 3.3% improvement in logic consistency compared to Dreamer, Hieros, and HRSSM. For instance, while the masking and hierarchical strategies in Hieros and HRSSM can reduce single-step propagation errors by mitigating environment noise, they still have difficulty in addressing long-term imagination due to predictive deviation in statistical inference and error accumulation. This highlights the limitations of relying solely on System 1 and the need for logical consistency from System 2 for robust imagination.

Table 2: Performance comparison of our approach with various baselines on DMC tasks in terms of logic consistency. The mean and variance of logical consistency are reported over 100 test episodes with the horizon size $H = 30$. Complete results of logical consistency over 20 tasks with varying horizon size are concluded in Appendix H.1.

Env	Dreamer [7]	Hieros [3]	HRSSM [15]	DMWM (**Proposed**)
Cartpole Balance	0.683 ± 0.057	0.711 ± 0.032	0.713 ± 0.041	$\mathbf{0.727 \pm 0.023}$
Pendulum Swingup	0.611 ± 0.137	0.709 ± 0.054	0.699 ± 0.079	$\mathbf{0.730 \pm 0.037}$
Reacher Hard	0.608 ± 0.121	0.702 ± 0.062	0.703 ± 0.072	$\mathbf{0.730 \pm 0.042}$
Finger Turn Hard	0.627 ± 0.131	0.698 ± 0.061	0.703 ± 0.073	$\mathbf{0.725 \pm 0.029}$
Cheetah Run	0.643 ± 0.131	0.689 ± 0.113	0.695 ± 0.087	$\mathbf{0.725 \pm 0.049}$
Cup Catch	0.652 ± 0.087	0.701 ± 0.072	0.714 ± 0.061	$\mathbf{0.728 \pm 0.021}$
Walker Walk	0.612 ± 0.140	0.696 ± 0.063	0.701 ± 0.073	$\mathbf{0.730 \pm 0.034}$
Quadruped Walk	0.656 ± 0.092	0.701 ± 0.067	0.703 ± 0.072	$\mathbf{0.723 \pm 0.039}$
Hopper Hop	0.633 ± 0.127	0.704 ± 0.087	0.701 ± 0.092	$\mathbf{0.722 \pm 0.038}$

Trial efficiency. Figure 4 presents test returns under limited environment trials. The number of environment trials serves as the x-axis to quantify exploration efficiency by measuring performance under the same environment exploration opportunities, which is a crucial factor for real-world applications and high-cost simulations. The complete results for 20 DMC tasks under limited environment trials are provided in Appendix H.2. The results show that the proposed DMWM-AC and DMWM-MPC significantly outperform baseline methods across most tasks with limited

environment trials, particularly in complex environments such as Cheetah Run and Finger Turn Hard, and Quadruped Run. In contrast, Dreamer and GD-MPC exhibit limited performance in high-dimensional control tasks with lower learning efficiency and stability. As shown in Figure 4, DMWM approaches achieve an average 5.5-fold improvement in test return under limited environment trials compared to baseline methods. These observations highlight that leveraging logical information from the environment enhances exploration efficiency, especially in complex tasks, by enhancing DMWM's capability in long-term imagination.

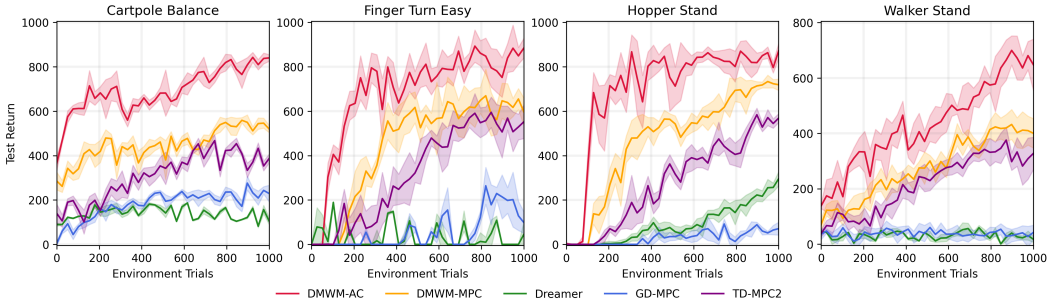

Figure 4: Performance comparison of results on 4 DMC tasks under environment trials that indicate the number of times that models explore the environments. The vertical axis indicates the average return over 100 test episodes. Complete results on 20 DMC tasks are concluded in Appendix H.2.

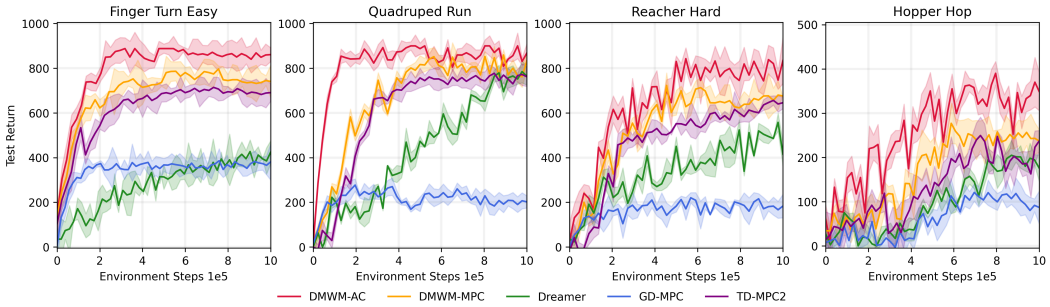

Figure 5: Performance comparison on 4 DMC tasks under environment steps that indicate the number of environment interactions. The vertical axis denotes the average test return over 100 episodes. Complete test results on 20 DMC tasks are provided in Appendix H.3.

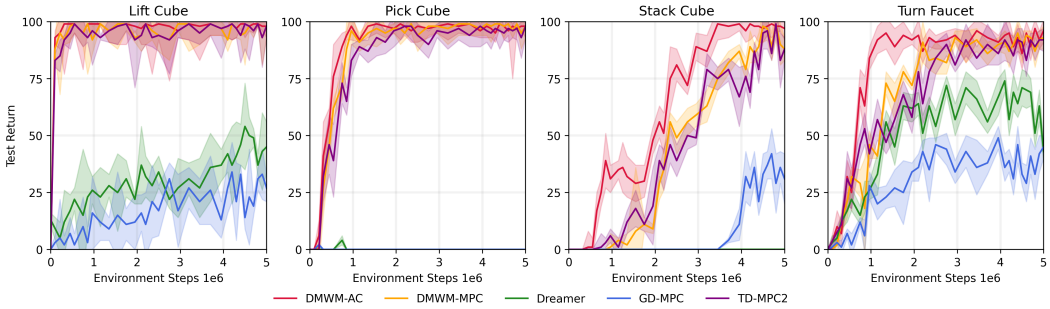

Figure 6: Performance comparison of test results on 4 ManiSkill2 robotic tasks under environment steps. The vertical axis denotes the average test return over 100 episodes.

Data efficiency. Figures 5-7 present test results under limited environment steps for the purpose of quantifying the data efficiency. The complete results for 20 DMC tasks under limited environment steps are provided in Appendix H.3. DMWM approaches consistently outperform Dreamer and GD-MPC across most tasks in terms of convergence speed and final returns, particularly in high-dimensional dynamics such as Cheetah Run and Quadruped Run. These results highlight DMWM's superior long-term planning and dynamic modeling capabilities, enabling more efficient utilization of environment data. For tasks that require simple dynamic modeling like Cartpole Balance and Cup Catch, all methods converge rapidly. In contrast, for complex tasks, such as Reacher Hard and Hopper Hop, the proposed DMWM approaches achieve stable performance and an average improvement of 32% in test return under limited data compared to Dreamer and GD-MPC.

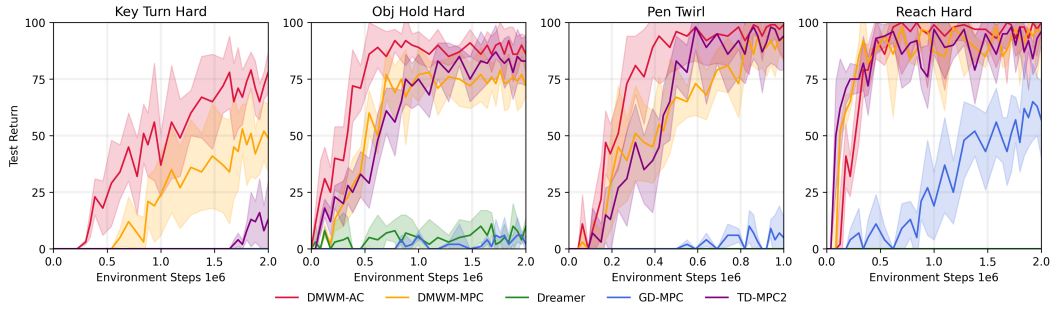

Figure 7: Performance comparison of test results on 4 MyoSuite robotic tasks under environment steps. The vertical axis denotes the average test return over 100 episodes.

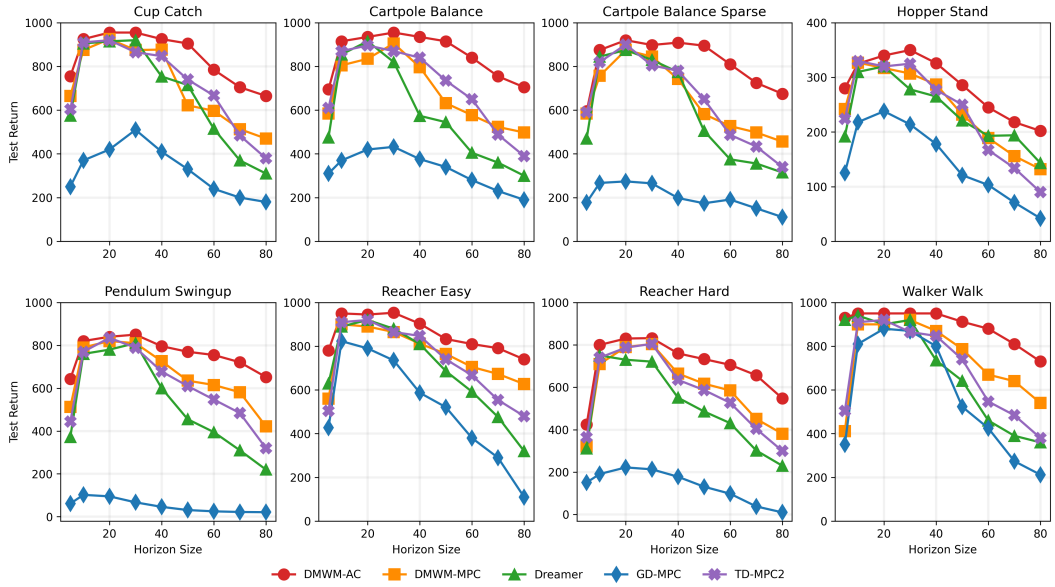

Figure 8: Performance comparison on 8 DMC tasks across different horizon size. Complete results on 20 DMC tasks across varying horizon size are provided in Appendix H.4.

Imagination ability over extended horizon size. As shown in Figure 8, although a long-horizon prediction introduces cumulative errors, DMWM-AC and DMWM-MPC consistently achieve high test returns across most tasks and remain stable performance even over extended horizons. In stability-critical tasks, such as Cartpole Balance and Pendulum Swingup, DMWM approaches maintain strong performance across a wide range of prediction horizons, whereas Dreamer and GD-MPC are more susceptible to degradation due to prediction errors. For extended horizon size of $H > 30$ in complex control tasks, DMWM approaches achieve an average 120% improvement in test return compared to Dreamer and GD-MPC. These results emphasize the crucial role of logical reasoning in long-term imagination. Across most tasks, DMWM approaches demonstrate superior performance over long-term horizons.

Impact of logic inference depth. Figure 9 presents test returns with inference depth $\alpha = 10, 30, 50$ and the complete results for 20 DMC tasks are provided in Appendix H.5. From Figure 9, we observe that a bigger reasoning depth α can produce performance gains in long-horizon decision making along with diminishing marginal returns from increasing logical inference depth and increased computational overhead. Moreover, Figure 9 also shows that the proposed deep logic inference can effectively capture the logic over the long-horizon trajectories for more robust imagination.

4 Related Works

World models are critical components in model-based intelligent systems, which enable learning, reasoning and decision-making in complex environments [5, 6, 7, 32, 33, 34, 35, 36, 37]. By

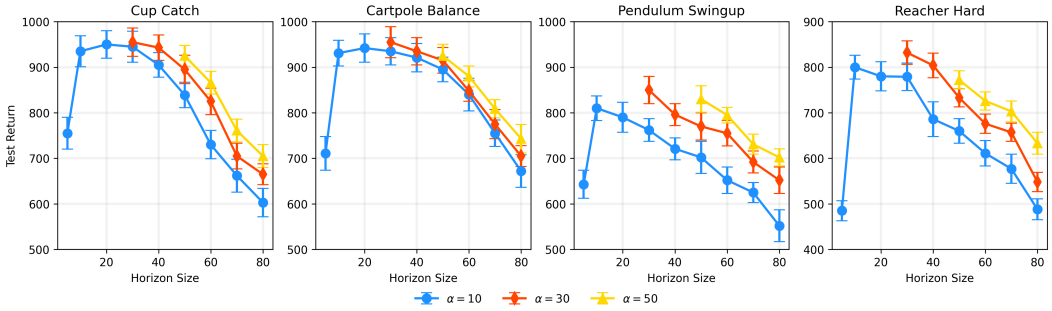

Figure 9: Performance comparison of results on 4 DMC tasks under environment trials that indicate the number of times that models explore the environments. The vertical axis indicates the average return over 100 test episodes. Complete results on 20 DMC tasks are concluded in Appendix H.5.

simulating the environment, these models imagine future states, plan behaviors and optimize strategies without relying heavily on real-world trial and error. Existing frameworks for world modeling, such as RSSM [5, 6, 7], generative models [32, 33, 34], and large language models (LLM) [35, 36, 37], have demonstrated varying strengths in tasks, such as prediction and efficient behavior planning. The Dreamer series [5, 6, 7], for instance, has advanced model-based reinforcement learning by improving task generalization and sample efficiency through latent space modeling and planning. Additionally, diffusion model-based world models [32, 33, 34] can generate diverse and high-quality outputs but cannot ensure long-term logical consistency. On the other hand, LLM-based models [35, 36, 37] can perform limited logical reasoning and task decomposition but face challenges in dynamic environment modeling and resource consumption.

Furthermore, logic neural reasoning is a promising approach to enhance the generalization ability and long-term imagination of world models [38, 29, 39]. Previous research has explored logic neural networks and probabilistic logic [40, 30, 41], but these methods have struggled with dynamic environments and evolving logical variables. Our approach, by contrast, enables the automatical and implicit logic inference, thus providing superior generalization and robustness in complex, dynamic task settings. Hence, in contrast to the existing world models, our work introduces a novel dual-mind framework that combines the efficient sampling of RSSM with logic-driven reasoning to enhance long-term imagination with logical consistency, thus addressing a critical research gap for logic-driven robust world models.

5 Conclusion and Limitations

Conclusion. In this paper, we have proposed DMWM, a novel world model framework for reliable long-term imagination inspired by the dual-process theory of human cognition. The proposed DMWM combines the fast, intuitive-driven S1 with the structured, logic inference-driven S2. We have designed an efficient inter-system feedback mechanism that enhances the logic consistency and adaptability of imagination trajectories. Extensive evaluations on DMControl and robotic tasks across diverse benchmarks demonstrated significant improvements in logical consistency, data-efficiency and reliable imagination over an extended horizon size.

Limitations and future work. A key limitation of our approach lies in its reliance on predefined simple domain-specific logical rules, which restricts its adaptability to environments where such rules are ambiguous or constantly evolving. This dependency limits the framework's ability to generalize to novel tasks. Future work could focus on enabling the model to autonomously learn and adapt logical rules from data by exploring causal relationships. This, in turn, can reduce the need for explicit manual definitions and enhance flexibility in dynamic and complex environments.

Acknowledgments

This work was supported by the US National Science Foundation under Grants CNS-2225511, IIS-2509636, DBI-2412389, CMMI-2240402, IIS-2312794, and CCF-1918770. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsor(s).

References

[1] Zhixuan Lin, Yi-Fu Wu, Skand Peri, Bofeng Fu, Jindong Jiang, and Sungjin Ahn. Improving generative imagination in object-centric world models. In *International conference on machine learning*, pages 6140–6149. PMLR, 2020.

[2] Guangxiang Zhu, Minghao Zhang, Honglak Lee, and Chongjie Zhang. Bridging imagination and reality for model-based deep reinforcement learning. *Advances in Neural Information Processing Systems*, 33:8993–9006, 2020.

[3] Paul Mattes, Rainer Schlosser, and Ralf Herbrich. Hieros: Hierarchical imagination on structured state space sequence world models. *arXiv preprint arXiv:2310.05167*, 2023.

[4] Lior Cohen, Kaixin Wang, Bingyi Kang, and Shie Mannor. Improving token-based world models with parallel observation prediction. *arXiv preprint arXiv:2402.05643*, 2024.

[5] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

[6] Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

[7] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, pages 1–7, 2025.

[8] Yucen Wang, Shenghua Wan, Le Gan, Shuai Feng, and De-Chuan Zhan. Ad3: Implicit action is the key for world models to distinguish the diverse visual distractors. *arXiv preprint arXiv:2403.09976*, 2024.

[9] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pages 2555–2565. PMLR, 2019.

[10] Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *arXiv preprint arXiv:2203.04955*, 2022.

[11] Jyothir SV, Siddhartha Jalagam, Yann LeCun, and Vlad Sobal. Gradient-based planning with world models. *arXiv preprint arXiv:2312.17227*, 2023.

[12] Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *Advances in Neural Information Processing Systems*, 36, 2024.

[13] Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. *arXiv preprint arXiv:2310.16828*, 2023.

[14] Qi Wang, Junming Yang, Yunbo Wang, Xin Jin, Wenjun Zeng, and Xiaokang Yang. Making offline rl online: Collaborative world models for offline visual reinforcement learning. *Advances in Neural Information Processing Systems*, 37:97203–97230, 2024.

[15] Ruixiang Sun, Hongyu Zang, Xin Li, and Riashat Islam. Learning latent dynamic robust representations for world models. *arXiv preprint arXiv:2405.06263*, 2024.

[16] Qi Wang, Zhipeng Zhang, Baao Xie, Xin Jin, Yunbo Wang, Shiyu Wang, Liaomo Zheng, Xiaokang Yang, and Wenjun Zeng. Disentangled world models: Learning to transfer semantic knowledge from distracting videos for reinforcement learning. *arXiv preprint arXiv:2503.08751*, 2025.

[17] Nan Rosemary Ke, Amanpreet Singh, Ahmed Touati, Anirudh Goyal, Yoshua Bengio, Devi Parikh, and Dhruv Batra. Modeling the long term future in model-based reinforcement learning. In *International Conference on Learning Representations*, 2018.

[18] Anthony Simeonov, Yilun Du, Beomjoon Kim, Francois Hogan, Joshua Tenenbaum, Pulkit Agrawal, and Alberto Rodriguez. A long horizon planning framework for manipulating rigid pointcloud objects. In *Conference on Robot Learning*, pages 1582–1601. PMLR, 2021.

[19] Joseph Clinton and Robert Lieck. Planning transformer: Long-horizon offline reinforcement learning with planning tokens. *arXiv preprint arXiv:2409.09513*, 2024.

[20] Nathan Lambert, Albert Wilcox, Howard Zhang, Kristofer SJ Pister, and Roberto Calandra. Learning accurate long-term dynamics for model-based reinforcement learning. In *2021 60th IEEE Conference on decision and control (CDC)*, pages 2880–2887. IEEE, 2021.

[21] Shiqian Li, Kewen Wu, Chi Zhang, and Yixin Zhu. On the learning mechanisms in physical reasoning. *Advances in Neural Information Processing Systems*, 35:28252–28265, 2022.

[22] Jianping Zhu, Xin Guo, Yang Chen, Yao Yang, Wenbo Li, Bo Jin, and Fei Wu. Adaptive meta-learning probabilistic inference framework for long sequence prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17159–17166, 2024.

[23] Artyom Sorokin, Nazar Buzun, Leonid Pugachev, and Mikhail Burtsev. Explain my surprise: Learning efficient long-term memory by predicting uncertain outcomes. *Advances in Neural Information Processing Systems*, 35:36875–36888, 2022.

[24] Jiajian Li, Qi Wang, Yunbo Wang, Xin Jin, Yang Li, Wenjun Zeng, and Xiaokang Yang. Open-world reinforcement learning over long short-term imagination. *arXiv preprint arXiv:2410.03618*, 2024.

[25] Jonathan St BT Evans and Keith E Stanovich. Dual-process theories of higher cognition: Advancing the debate. *Perspectives on psychological science*, 8(3):223–241, 2013.

[26] Keith Frankish. Dual-process and dual-system theories of reasoning. *Philosophy Compass*, 5(10):914–926, 2010.

[27] Jonathan St BT Evans. In two minds: dual-process accounts of reasoning. *Trends in cognitive sciences*, 7(10):454–459, 2003.

[28] Wenyue Hua and Yongfeng Zhang. System 1+ system 2= better world: Neural-symbolic chain of logic reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 601–612, 2022.

[29] Shiqian Li, Yuxi Ma, Jiajun Yan, Bo Dai, Yujia Peng, Chi Zhang, and Yixin Zhu. A simulation-heuristics dual-process model for intuitive physics. *arXiv preprint arXiv:2504.09546*, 2025.

[30] Shaoyun Shi, Hanxiong Chen, Weizhi Ma, Jiaxin Mao, Min Zhang, and Yongfeng Zhang. Neural logic reasoning. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1365–1374, 2020.

[31] Hanxiong Chen, Shaoyun Shi, Yunqi Li, and Yongfeng Zhang. Neural collaborative reasoning. In *Proceedings of the Web Conference 2021*, pages 1516–1527, 2021.

[32] Chang Chen, Yi-Fu Wu, Jaesik Yoon, and Sungjin Ahn. Transdreamer: Reinforcement learning with transformer world models. *arXiv preprint arXiv:2202.09481*, 2022.

[33] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. Drivedreamer: Towards real-world-driven world models for autonomous driving. *arXiv preprint arXiv:2309.09777*, 2023.

[34] Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *arXiv preprint arXiv:2405.12399*, 2024.

[35] Kolby Nottingham, Prithviraj Ammanabrolu, Alane Suhr, Yejin Choi, Hannaneh Hajishirzi, Sameer Singh, and Roy Fox. Do embodied agents dream of pixelated sheep: Embodied decision making using language guided world modelling. In *International Conference on Machine Learning*, pages 26311–26325. PMLR, 2023.

[36] Lionel Wong, Gabriel Grand, Alexander K Lew, Noah D Goodman, Vikash K Mansinghka, Jacob Andreas, and Joshua B Tenenbaum. From word models to world models: Translating from natural language to the probabilistic language of thought. *arXiv preprint arXiv:2306.12672*, 2023.

[37] Jiannan Xiang, Tianhua Tao, Yi Gu, Tianmin Shu, Zirui Wang, Zichao Yang, and Zhiting Hu. Language models meet world models: Embodied experiences enhance language models. *Advances in neural information processing systems*, 36, 2024.

[38] Joao Ferreira, Manuel de Sousa Ribeiro, Ricardo Gonçalves, and Joao Leite. Looking inside the black-box: Logic-based explanations for neural networks. In *Proceedings of the international conference on principles of knowledge representation and reasoning*, volume 19, pages 432–442, 2022.

[39] Tao Li and Vivek Srikumar. Augmenting neural networks with first-order logic. *arXiv preprint arXiv:1906.06298*, 2019.

[40] Meng Qu and Jian Tang. Probabilistic logic neural networks for reasoning. *Advances in neural information processing systems*, 32, 2019.

[41] Ryan Riegel, Alexander Gray, Francois Luus, Naweed Khan, Ndivhuwo Makondo, Ismail Yunus Akhalwaya, Haifeng Qian, Ronald Fagin, Francisco Barahona, Udit Sharma, et al. Logical neural networks. *arXiv preprint arXiv:2006.13155*, 2020.

[42] Quentin Garrido, Mahmoud Assran, Nicolas Ballas, Adrien Bardes, Laurent Najman, and Yann LeCun. Learning and leveraging world models in visual representation learning. *arXiv preprint arXiv:2403.00504*, 2024.

[43] Aidan Scannell, Mohammadreza Nakhaei, Kalle Kujanpää, Yi Zhao, Kevin Sebastian Luck, Arno Solin, and Joni Pajarinen. Discrete codebook world models for continuous control. *arXiv preprint arXiv:2503.00653*, 2025.

[44] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022.

[45] Weirui Ye, Shaohuai Liu, Thanard Kurutach, Pieter Abbeel, and Yang Gao. Mastering atari games with limited data. *Advances in neural information processing systems*, 34:25476–25488, 2021.

[46] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

[47] Michel Ma, Tianwei Ni, Clement Gehring, Pierluca D'Oro, and Pierre-Luc Bacon. Do transformer world models give better policy gradients? *arXiv preprint arXiv:2402.05290*, 2024.

[48] Jan Robine, Marc Höftmann, and Stefan Harmeling. Simple, good, fast: Self-supervised world models free of baggage. In *The Thirteenth International Conference on Learning Representations*, 2025.

[49] Maxime Burchi and Radu Timofte. Learning transformer-based world models with contrastive predictive coding. *arXiv preprint arXiv:2503.04416*, 2025.

[50] Jonathan Feldstein, Paulius Dilkas, Vaishak Belle, and Efthymia Tsamoura. Mapping the neuro-symbolic ai landscape by architectures: A handbook on augmenting deep learning through symbolic reasoning. *arXiv preprint arXiv:2410.22077*, 2024.

[51] Chaojun Ni, Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Wenkang Qin, Guan Huang, Chen Liu, Yuyin Chen, Yida Wang, Xueyang Zhang, et al. Recondreamer: Crafting world models for driving scene reconstruction via online restoration. *arXiv preprint arXiv:2411.19548*, 2024.

[52] Sicheng Zuo, Wenzhao Zheng, Yuanhui Huang, Jie Zhou, and Jiwen Lu. Gaussianworld: Gaussian world model for streaming 3d occupancy prediction. *arXiv preprint arXiv:2412.10373*, 2024.

[53] Chenyang Cao, Yucheng Xin, Silang Wu, Longxiang He, Zichen Yan, Junbo Tan, and Xueqian Wang. Fosp: Fine-tuning offline safe policy through world models. *arXiv preprint arXiv:2407.04942*, 2024.

[54] Yang Yue, Yulin Wang, Haojun Jiang, Pan Liu, Shiji Song, and Gao Huang. Echoworld: Learning motion-aware world models for echocardiography probe guidance. *arXiv preprint arXiv:2504.13065*, 2025.

[55] Yichao Liang, Nishanth Kumar, Hao Tang, Adrian Weller, Joshua B Tenenbaum, Tom Silver, João F Henriques, and Kevin Ellis. Visualpredicator: Learning abstract world models with neuro-symbolic predicates for robot planning. *arXiv preprint arXiv:2410.23156*, 2024.

[56] Leonardo Barcellona, Andrii Zadaianchuk, Davide Allegro, Samuele Papa, Stefano Ghidoni, and Efstratios Gavves. Dream to manipulate: Compositional world models empowering robot imitation learning with imagination. *arXiv preprint arXiv:2412.14957*, 2024.

[57] Ignat Georgiev, Varun Giridhar, Nicklas Hansen, and Animesh Garg. Pwm: Policy learning with multi-task world models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[58] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[59] Kenneth R Muske and James B Rawlings. Model predictive control with linear models. *AIChE Journal*, 39(2):262–287, 1993.

[60] Sanket Kamthe and Marc Deisenroth. Data-efficient reinforcement learning with probabilistic model predictive control. In *International conference on artificial intelligence and statistics*, pages 1701–1710. PMLR, 2018.

[61] Zhe Wu, David Rincon, Quanquan Gu, and Panagiotis D Christofides. Statistical machine learning in model predictive control of nonlinear processes. *Mathematics*, 9(16):1912, 2021.

[62] Vincent Micheli, Eloi Alonso, and François Fleuret. Efficient world models with context-aware tokenization. *arXiv preprint arXiv:2406.19320*, 2024.

[63] Siyuan Zhou, Yilun Du, Jiaben Chen, Yandong Li, Dit-Yan Yeung, and Chuang Gan. Robodreamer: Learning compositional world models for robot imagination. *arXiv preprint arXiv:2404.12377*, 2024.

[64] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. Daydreamer: World models for physical robot learning. In *Conference on robot learning*, pages 2226–2240. PMLR, 2023.

[65] Jessy Lin, Yuqing Du, Olivia Watkins, Danijar Hafner, Pieter Abbeel, Dan Klein, and Anca Dragan. Learning to model the world with language. *arXiv preprint arXiv:2308.01399*, 2023.

[66] Yunhai Feng, Nicklas Hansen, Ziyan Xiong, Chandramouli Rajagopalan, and Xiaolong Wang. Finetuning offline world models in the real world. *arXiv preprint arXiv:2310.16029*, 2023.

[67] Ignat Georgiev, Varun Giridhar, Nicklas Hansen, and Animesh Garg. Pwm: Policy learning with large world models. *arXiv preprint arXiv:2407.02466*, 2024.

[68] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

[69] Lunjun Zhang, Yuwen Xiong, Ze Yang, Sergio Casas, Rui Hu, and Raquel Urtasun. Learning unsupervised world models for autonomous driving via discrete diffusion. *arXiv preprint arXiv:2311.01017*, 2023.

[70] Marc Rigter, Jun Yamada, and Ingmar Posner. World models via policy-guided trajectory diffusion. *arXiv preprint arXiv:2312.08533*, 2023.

[71] Zihan Ding, Amy Zhang, Yuandong Tian, and Qinqing Zheng. Diffusion world model. *arXiv preprint arXiv:2402.03570*, 2024.

[72] Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.

[73] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024.

[74] Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*, 2023.

[75] Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. ivideogpt: Interactive videogpts are scalable world models. *arXiv preprint arXiv:2405.15223*, 2024.

[76] Jiannan Xiang, Guangyi Liu, Yi Gu, Qiyue Gao, Yuting Ning, Yuheng Zha, Zeyu Feng, Tianhua Tao, Shibo Hao, Yemin Shi, et al. Pandora: Towards general world model with natural language actions and video states. *arXiv preprint arXiv:2406.09455*, 2024.

[77] Samy Badreddine, Artur d'Avila Garcez, Luciano Serafini, and Michael Spranger. Logic tensor networks. *Artificial Intelligence*, 303:103649, 2022.

[78] Prithviraj Sen, Breno WSR de Carvalho, Ryan Riegel, and Alexander Gray. Neuro-symbolic inductive logic programming with logical neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 8212–8219, 2022.

Contents of Appendix

A Impact Statement

Inspired by the human cognitive model, our work proposes a novel DMWM architecture that integrates intuitive RSSM-S1 with logic-driven LINN-S2 for the first time, which addresses long-horizon imagination for model-based RL and MPC. By enhancing logical consistency of the architecture, DMWM improves both efficiency and robustness in complex tasks. Because of these benefits, DMWM provides a practical and interpretable solution for real-world applications such as autonomous driving and robotic planning.

Moreover, this work presents a generalized dual-mind framework for world models that can serve as a solid foundation for future research on general world models. We believe this approach marks a meaningful step toward artificial general intelligence (AGI) by bridging intuitive processes with logical reasoning. Hence, DWMW paves the way for more robust, logical and human-like decision-making systems.

B Background

B.1 Actor-Critic

The actor-critic model is a widely used algorithmic framework in Reinforcement Learning (RL), which combines the advantages of policy gradient methods and value function approximation. It addresses RL tasks through the collaborative learning of two primary components: the Actor (policy network) and the Critic (value network), given as follows:

$$\begin{aligned} &\text{Action model:} \quad a_\tau \sim q_\vartheta(a_\tau \mid s_\tau) \\ &\text{Value model:} \quad v_\psi(s_\tau) \approx \mathbb{E}_{q(\cdot|s_\tau)}\left(\sum_{t=\tau}^{H} \gamma^{t-\tau} r_\tau\right). \end{aligned} \tag{18}$$

In the world Model, the target of the actor-critic model is to maximize the reward over the imagined trajectories with the horizon size H. Specifically, the action model seeks to maximize an estimated value, while the value model strives to accurately predict the value estimate, which evolves as the action model updates. The training target of the Actor and the Value are given as follows [58, 5, 11].

$$\vartheta^* = \max_{\vartheta} \mathbb{E}_{q_\phi, q_\vartheta}\left[\sum_{\tau=t}^{t+H} V_\lambda(s_\tau)\right] \tag{19}$$

$$\psi^* = \min_{\psi} \mathbb{E}_{q_\phi, q_\vartheta}\left[\sum_{\tau=t}^{t+H} \frac{1}{2}\left(v_\psi(s_\tau) - V_\lambda(s_\tau)\right)^2\right] \tag{20}$$

$$V_\lambda(s_\tau) = (1-\lambda)\left(\sum_{n=1}^{H-1} \lambda^{n-1} V_n^N(s_\tau)\right) + \lambda^{H-1} V_H^N(s_\tau) \tag{21}$$

$$V_k^N(s_\tau) = \mathbb{E}_{q_\phi, q_\vartheta}\left[\sum_{n=\tau}^{h-1} \gamma^{n-\tau} r_n + \gamma^{h-\tau} v_\psi(s_h)\right], \tag{22}$$

where $h = \min(\tau + k, t + H)$, ϑ represents the parameters of the Actor ψ represents the parameters of the Critic, and λ is the discount factor.

B.2 Model Predictive Control

Model predictive control (MPC) is an optimization-based control method extensively applied in engineering and industrial control systems [13, 59, 60, 61]. By leveraging the system's dynamic model, MPC predicts future behavior through rolling optimization, generating optimal control inputs at each time step to achieve the desired system objectives. However, due to the fact that MPC strongly relies on the system model and the requirement for online optimization, it has difficulty in effective decision-making performance if the world model fails to provide stable and reliable

imagined trajectories [9, 11]. The gradient-based MPC framework [11] for world model is presented as

$$a^*_{t:t+H} = \max_{a^{(j)}_{t:t+H}} R^{(j)}, \quad R^{(j)} = \sum_{\tau=t+1}^{t+H+1} \mathbb{E}\big[q_\phi(r_\tau \mid s^{(j)}_\tau)\big] \tag{23}$$

$$\left\{a^{(j)}_{t:t+H}\right\}_{j=1}^{J} \sim \mathcal{N}(\mu_t, \mathrm{diag}(\sigma_t^2)) \tag{24}$$

$$s^{(j)}_{t:t+H+1} \sim q_\phi(s_t \mid o^{(j)}_{1:t}, a^{(j)}_{1:t-1}) \prod_{\tau=t+1}^{t+H+1} p_\phi(s_\tau \mid s^{(j)}_{\tau-1}, a^{(j)}_{\tau-1}) \tag{25}$$

$$a^{(j)}_{t:t+H} = a^{(j)}_{t:t+H} - \nabla R^{(j)}, \tag{26}$$

where $a^{(j)}_{t:t+H}$ and $s^{(j)}_{t:t+H}$ are sequences of states and actions of the candidate j from the timestep t to the timestep $t+H$.

C Derivation of Logical ELBO for Observation Loss

The ELBO with logic rules for the observation loss can be derived by

$$\begin{aligned}
\ln p_\varphi\left(o_{1:T} \mid a_{1:T}\right) &= \int p_\varphi(o_{1:T}, z_{1:T} \mid a_{1:T})\, dz_{1:T} \\
&= \mathbb{E}_{q_\varphi(z_{1:T}|o_{1:T},a_{1:T})}\left[\frac{p_\varphi(o_{1:T}, z_{1:T} \mid a_{1:T})}{q_\varphi(z_{1:T} \mid o_{1:T}, a_{1:T})}\right] \\
&= \ln \mathbb{E}_{q_\varphi(z_{1:T}|o_{1:T},a_{1:T})}\left[\prod_{t=1}^{T} \frac{p_\varphi(o_t \mid z_t) p_\varphi(z_t \mid z_{t-1}, a_{t-1}) \mathcal{C}(\phi(z_t, z_{t-1}, a_{t-1}))}{q_\varphi(z_t \mid o_{\le t}, a_{\le t})}\right] \\
&\ge \mathbb{E}_{q_\varphi(z_{1:T}|o_{1:T},a_{1:T})}\left[\sum_{t=1}^{T} \ln p_\varphi(o_t \mid z_t) + \ln p_\varphi(z_t \mid z_{t-1}, a_{t-1})\right. \\
&\qquad \left. + \ln \mathcal{C}(\phi(z_t, z_{t-1}, a_{t-1}) - \ln q_\varphi(z_t \mid o_{\le t}, a_{\le t}))\right] \\
&= \sum_{t=1}^{T} \Bigg(\underbrace{\mathbb{E}_{q_\varphi(z_t|o_{\le t},a_{<t})}\left[\ln p_\varphi\left(o_t \mid z_t\right)\right]}_{\text{Decoding}} + \underbrace{\mathbb{E}_{q_\varphi(z_t|o_{\le t},a_{<t})}\left[\ln \mathcal{C}(\phi(z_t, z_{t-1}, a_{t-1}))\right]}_{\text{Logic Inference}} \\
&\qquad \underbrace{- \mathbb{E}_{q_\varphi(z_{t-1}|o_{\le t-1},a_{<t-1})}\left[\mathrm{KL}\left[q_\varphi\left(z_t \mid o_{\le t}, a_{<t}\right) \| p_\varphi\left(z_t \mid z_{t-1}, a_{t-1}\right)\right]\right]}_{\text{Prediction}} \Bigg),
\end{aligned} \tag{27}$$

where $\mathcal{C}(\phi(z_t, z_{t-1}, a_{t-1})) = \mathrm{Sim}(\phi(z_t, z_{t-1}, a_{t-1}), \mathbf{T})$ is the logical consistency function that measures whether the logical rule $\phi : z_{t-1} \wedge a_{t-1} \to z_t$ is satisfied by imagination.

— Appendices continue on next page —

D Algorithms

D.1 DMWM With Actor-Critic

The training process of DMWM with actor-critic-based decision module is shown in Algorithm 1.

Algorithm 1 DMWM With Actor-Critic

Hyper Parameters: seed episode S, training episodes N, batch size B, collect interval C, sequence length L, imagination horizon H, learning rate η.
Initialize dataset $\mathcal{D}$ with S seed episodes.
Initialize DMWM parameters ϕ, w.
Initialize Actor-Critic parameters ϑ, ψ.
for Training step $n \to N$ **do**
  for Collect interval $c \to C$ **do**
    // System 1 Training
    Sample B Sequences $\{(o_t, a_t, r_t)\}_{t=k}^{k+L} \sim \mathcal{D}$.
    Compute $h_t = f_\varphi\left(h_{t-1}, z_{t-1}, a_{t-1}\right)$.
    Predict $\hat{z}_t \sim p_\varphi\left(\hat{z}_t \mid h_t\right), \hat{o}_t \sim p_\varphi\left(\hat{o}_t \mid h_t, z_t\right)$.
    Update S1 $\psi \leftarrow \psi - \eta_\psi \nabla_\psi \mathcal{L}_{\text{S1}}(\psi)$.
    // System 2 Training
    Learn Logic Regularizations $\mathcal{L}_{\text{reg}}$.
    // S1's Guidance on S2 Based on Truth $\{(h_t, a_t, h_{t+1})\}$
    Compute $L_T^\lambda = \phi_1^\lambda \wedge \phi_2^\lambda \wedge \phi_3^\lambda \cdots \phi_{T-1}^\lambda$.
    Update S2 $w \leftarrow w - \eta_w \nabla_w \mathcal{L}_{\text{S1}}(w)$.
    // Actor-Critic Training
    Imagine $\{(z_\tau, a_\tau)\}_{\tau=t}^{t+H}$ from each z_t.
    Predict rewards $\mathbb{E}(q_\varphi(r_\tau \mid h_\tau, z_\tau))$ and values $v_\psi(s_\tau)$.
    Compute value estimates $V_\lambda(s_\tau)$.
    Update $\vartheta \leftarrow \vartheta + \eta_\vartheta \nabla_\vartheta \sum_{\tau=t}^{t+H} V_\lambda(s_\tau)$.
    Update $\psi \leftarrow \psi - \eta_\psi \nabla_\psi \sum_{\tau=t}^{t+H} \frac{1}{2}\|v_\psi(s_\tau) - V_\lambda(s_\tau)\|^2$.
    // S2's Guidance on S1 Based on Imagination $\{(z_\tau, a_\tau)\}$
    Compute Logic Consistency of $\{(z_\tau, a_\tau)\}$ With L_T^λ
    Update S1 $\psi \leftarrow \psi - \eta_\psi \nabla_\psi \mathcal{L}_{\text{S2}}(\psi)$.
  end for
  // Real Environment Interaction & Data Collection
  Start a environment $env.reset()$
  for Time step $t \to T$ **do**
    Compute $h_t = f_\varphi\left(h_{t-1}, z_{t-1}, a_{t-1}\right)$.
    Compute $z_t \sim q_\varphi\left(z_t \mid h_t, o_t\right)$
    Obtain $a_t \sim q_\vartheta(a_t \mid z_t)$ from decision-making model.
    Interact $r_t, o_{t+1} \leftarrow env.step(a_t)$.
  end for
  Add experience to dataset $\mathcal{D} \leftarrow \mathcal{D} \cup \{(o_t, a_t, r_t)\}_{t=1}^{T}$.
end for

D.2 DMWM With MPC

The training process of DMWM with Grad-MPC for decision-making [11] is shown in Algorithm 2.

Algorithm 2 DMWM With Grad-MPC

Hyper Parameters: iterations I, candidate Size J, learning rate η_R.
Initialize dataset $\mathcal{D}$ with S seed episodes.
Initialize DMWM parameters ϕ, w.
for Training step $n \to N$ **do**
  ...
  // Real Environment Interaction & Data Collection
  Start a environment $env.reset()$
  for Time step $t \to T$ **do**
    Compute $h_t = f_\varphi\left(h_{t-1}, z_{t-1}, a_{t-1}\right)$.
    Compute $z_t \sim q_\varphi\left(z_t \mid h_t, o_t\right)$
    // MPC Decision Making
    Sample Actions $\left\{a_{t:t+H}^{(j)}\right\}_{j=1}^{J} \sim \mathcal{N}(\mu_t, \mathrm{diag}(\sigma_t^2))$.
    for Iteration $i \to I$ **do**
      for Candidate sequence $j \to J$ **do**
        $s_{t:t+H+1}^{(j)} \sim q_\phi(s_t \mid o_{1:t}^{(j)}, a_{1:t-1}^{(j)}) \prod_{\tau=t+1}^{t+H+1} p_\phi(s_\tau \mid s_{\tau-1}^{(j)}, a_{\tau-1}^{(j)})$.
        $R^{(j)} = \sum_{\tau=t+1}^{t+H+1} \mathbb{E}\left[p(r_\tau \mid s_\tau^{(j)})\right]$.
        Update Action $a_{t:t+H}^{(j)} = a_{t:t+H}^{(j)} - \eta_R \nabla R^{(j)}$.
      end for
    end for
    $a_{t:t+H}^{*} = \max_{a_{t:t+H}^{(j)}} R^{(j)}$
    Interact $r_t, o_{t+1} \leftarrow env.step(a_t^*)$.
  end for
  Add experience to dataset $\mathcal{D} \leftarrow \mathcal{D} \cup \{(o_t, a_t, r_t)\}_{t=1}^{T}$.
  ...
end for

E Hyperparameters

E.1 Environment Setting

The action repeat setting for different DMControl tasks and robotic tasks is presented in TABLE 2.

Table 3: Action Repeat Setting and Action Dim

Env	Task	Action Dim	Action Repeat
DMC	Cartpole Swingup	1	8
	Pendulum Swingup	1	6
	Reacher Easy	2	4
	Finger Spin	2	2
	Cheetah Run	6	4
	Cup Catch	2	6
	Walker Walk	6	2
	Quadruped Walk	12	2
	Hopper Hop	4	2
MyoSuite	Key Turn Hard	39	1
	Object Hold Hard	39	1
	Pen Twirl	39	1
	Reach Hard	39	1
ManiSkill2	Lift Cube	4	2
	Pick Cube	4	2
	Stack Cube	4	2
	Turn Faucet	7	2

E.2 Model Setting

The hyperparameters of models are presented in TABLE 3.

Table 4: Hyperparameter Setting

Parameter	Symbol	Value
Dual-Mind World Model (General)		
Replay memory size	—	1e6
Batch size	B	50
Sequence length	L	64
Seed episode	S	5
Training episodes	N	1e3
Collect Interval	C	100
Max episode length	—	500
Exploration noise	—	0.3
Imagination horizon	H	30
Gradient clipping	—	100
RSSM-S1		
Activation function	—	Relu
Embedding size	—	1024
Hidden size	—	200
Belief size	—	200
State size	—	30
Overshooting distance	—	50
Overshooting KL-beta	—	0
Global KL-beta	—	0
overshooting reward scale	—	0
Free nats	—	3
Bit-depth	—	5
Weights	$\varpi_{\text{dyn}}, \varpi_{\text{rep}}$	1
Optimizer	—	Adam
Adam epsilon	—	1e-4
Learning rate	η_ψ	1e-3
LINN-S2		
Reasoning depth	α	30
Logic vector size	$\lvert v\rvert, \lvert m\rvert$	64
L2 weight	β_{ℓ_2}	1e-5
Regularization weight	β_{reg}	1
Logic MLP number	—	3
Optimizer	—	SGD
Learning rate	η_w	1e-2
Actor-Critic [5]		
Return lambda	λ	0.95
Planning horizon discount	—	0.99
Optimizer	—	Adam
Adam epsilon	—	1e-4
Learning rate	$\eta_\vartheta, \eta_\psi$	1e-4
Grad-MPC [11]		
Iterations	I	40
Candidate Size	J	1000
Learning Rate	η_R	0.1-0.01-0.005-0.0001
TD-MPC2 (refer to [13])		

F Further Related Work

F.1 World Model

World models are fundamental building blocks for model-based intelligent systems to make decisions, learn, and reason in complex environments. It enables prediction, efficient behavior planning through environment simulation, and strategy optimization using virtual simulations, thus reducing reliance on real-world trial and error [5, 6, 7, 62, 63]. Existing world modeling frameworks can be categorized as recurrent state space models (RSSMs) [2, 5, 6, 7, 10, 64, 65, 13, 66, 3, 15, 67], generative models [32, 33, 68, 69, 70, 34, 71, 72, 12, 73], and large language model (LLM) [35, 36, 74, 75, 37, 76].

The Dreamer series [5, 6, 7] lays an important foundation for general model-based reinforcement learning (MBRL) by continuously improving sample efficiency and task generalization ability through dynamic modeling and planning in the latent space. To adapt the RSSM-based world model for real-world environments, [2] optimized confidence and entropy regularization for the gradient to mitigate discrepancies between virtual simulations and real-world environments. [1] studied an object-centric world model, and introduced an object-state recurrent neural network (OS-RNN) to follow the object states. [12] extended multimodal RSSM to support joint text and visual inputs. [7, 13, 67] explored the multiple tasks with RSSM-based world models. [13] utilized SimNorm to sparse and normalize the potential states, which projected the latent representations to simplices of fixed dimensions, thus mitigating the gradient explosion and improving the training stability. [15] introduced masking-based latent reconstruction with a dual branch structure to handle the exogenous noise in the complex environment. [3] introduced the hierarchical imagination with parallel processing and proposed efficient time-balanced sampling.

The diffusion model-based world model offers high-quality and diverse generations with temporal smoothness and consistency, achieved through denoising and time inversion processes [32, 33, 68, 69, 70, 34, 12]. However, its reliance on multi-step denoising significantly slows downsampling, inference, and computation compared to RSSM, making it less suitable for real-time tasks and challenging to maintain long-term consistency [68, 69, 70, 12]. Additionally, unlike RSSM's latent space modeling, diffusion models cannot extract task-relevant features and accurately capture environment dynamics, particularly in complex environments with long-tailed distributions [32, 68, 70, 71, 12]. The LLM-based world model serves as a powerful tool for complex task planning and execution through its logical reasoning capability and dynamic controllability [35, 36, 74], and is able to realize task decomposition and cross-domain knowledge transfer through natural language instructions [76]. However, the model cannot perform in dynamic modeling, and the generated states and actions often cannot accurately reflect the dynamic changes in the environment. In addition, its high reliance on linguistic representation may lead to insufficient quality of modal alignment, posing the risk of information loss or misinterpretation, while the significant consumption of computational resources during training and inference limits the application of LLM-based world models in resource-constrained environments [35, 36, 74, 75].

Unlike all of the aforementioned works, we focus on the long-term imagination of world models and proposes to enable the logical consistency of state representations and predictions over an extended horizon for the first time. We retain the efficient sampling and representation capabilities of RSSM as System 1 and integrate a logic-integrated neural network (LINN) as System 2 to imbue the world model with logic reasoning capabilities. The proposed DMWM thus delivers reliable and efficient long-term imagination with logical consistency, addressing a critical research gap in existing approaches.

F.2 Logic Neural Reasoning

Logic neural reasoning enhances the logical consistency, long-term imagination, and generalization ability of world models by embedding logical rules and reasoning capabilities [38, 29]. By combining the interpretability of symbolic logic with the expressive power of neural networks, it provides robust and rapid reasoning for complex task planning [39, 40, 30, 41, 31, 77, 78]. However, explicitly modeling the logical structure of the world presents significant challenges due to the inherent ambiguity of logical rules, the prevalence of noisy data, and the dynamic and complex nature of logical interactions. [39] converted first-order logic rules into computational graphs for neural networks, enhancing their learning capabilities in low-data environments while providing limited support for complex temporal logic. [40] integrated Markov logic networks with knowledge graph

embeddings to address uncertainty in logical reasoning. In a logical neural network (LNN) [41], neurons are mapped neurons to logical formulas, enabling each neuron to function as a logical operator. However, LNNs rely on predefined and static logical rules, facing limitations in generalization when confronted with uncertainty or evolving logical variables. The work in [78] introduced inductive logic programming and automatically learned logic rules. However, the aforementioned approaches make it hard to capture the environment dynamics and extend the long-term imagination capability of the world model. In contrast, the concept of a LINN [30, 31] introduces a paradigm shift by dynamically constructing computational graphs and employing neural modules to learn logical operations. Thus, LINN can automatically infer implicit logical rules, circumventing the need for explicit specification. By integrating neural flexibility with logical rigor [30, 31], LINN achieves superior generalization capacity and enhanced robustness to noise, making it well-suited for reasoning tasks in dynamic and complex environments.

In this work, we enhance the long-term imagination capabilities of world models through logical reasoning, a critical yet unexplored area in world model research, with significant implications for creating advanced and general artificial intelligence (AGI).

— Appendices continue on next page —

G Complete Logic Regularization and Rules Table

The complete logic rules and the corresponding regularizers are given in TABLE 5 [30] for negation $\neg$, conjunction $\wedge$, disjunction $\vee$ and implication $\rightarrow$.

Table 5: Complete Logical Regularizers and Rules for System 2

	Logical Rule	Equation	Logic Regularizer r_i
$\neg$	**Negation**	$w \rightarrow \mathbf{T} = \mathbf{T}$	$r_1 = \sum_{w \in W \cup \{\mathbf{T}\}} \text{Sim}(\text{NOT}(w), w)$
	Negations	$\neg(\neg w) = w$	$r_2 = \sum_{w \in W} 1 - \text{Sim}(\text{NOT}(\text{NOT}(w)), w)$
$\wedge$	**Identity**	$w \wedge \mathbf{T} = w$	$r_3 = \sum_{w \in W} 1 - \text{Sim}(\text{AND}(w, \mathbf{T}), w)$
	Annihilator	$w \wedge \mathbf{T} = w$	$r_4 = \sum_{w \in W} 1 - \text{Sim}(\text{AND}(w, \mathbf{F}), \mathbf{F})$
	Idempotence	$w \wedge \mathbf{F} = \mathbf{F}$	$r_5 = \sum_{w \in W} 1 - \text{Sim}(\text{AND}(w, w), w)$
	Complement	$w \wedge w = w$	$r_6 = \sum_{w \in W} 1 - \text{Sim}(\text{AND}(w, \text{NOT}(w)), \mathbf{F})$
$\vee$	**Identity**	$w \vee \mathbf{F} = w$	$r_7 = \sum_{w \in W} 1 - \text{Sim}(\text{OR}(w, \mathbf{F}), w)$
	Annihilator	$w \vee \mathbf{T} = \mathbf{T}$	$r_8 = \sum_{w \in W} 1 - \text{Sim}(\text{OR}(w, \mathbf{T}), \mathbf{T})$
	Idempotence	$w \vee w = w$	$r_9 = \sum_{w \in W} 1 - \text{Sim}(\text{OR}(w, w), w)$
	Complement	$w \vee \neg w = \mathbf{T}$	$r_{10} = \sum_{w \in W} 1 - \text{Sim}(\text{OR}(w, \text{NOT}(w)), \mathbf{T})$
$\rightarrow$	**Identity**	$w \rightarrow \mathbf{T} = \mathbf{T}$	$r_{11} = \sum_{w \in W} 1 - \text{Sim}(\text{OR}(\text{NOT}(w), \mathbf{T}), \mathbf{T})$
	Annihilator	$w \rightarrow \mathbf{F} = \neg w$	$r_{12} = \sum_{w \in W} 1 - \text{Sim}(\text{OR}(\text{NOT}(w), \mathbf{F}), \text{NOT}(w))$
	Idempotence	$w \rightarrow w = \mathbf{T}$	$r_{13} = \sum_{w \in W} 1 - \text{Sim}(\text{OR}(\text{NOT}(w), w), \mathbf{T})$
	Complement	$w \rightarrow \neg w \equiv \neg w$	$r_{14} = \sum_{w \in W} 1 - \text{Sim}(\text{OR}(\text{NOT}(w), \text{NOT}(w)), \text{NOT}(w))$

H Additional Experiments

H.1 Logical Consistency

Table 6: Logical consistency comparison between our proposed DMWM and various RSSM baselines across 20 DMC tasks. We report the mean and variance of logical consistency from imagination over 100 test episodes with the horizon size of $H = 10, 30, 50, 100$.

Env	H	Dreamer	Hieros	HRSSM	DMWM (**Ours**)
Acrobot Swingup	10	0.713 ± 0.031	0.730 ± 0.012	0.722 ± 0.017	0.733 ± 0.007
	30	0.667 ± 0.063	0.704 ± 0.032	0.712 ± 0.044	$\mathbf{0.731 \pm 0.017}$
	50	0.568 ± 0.129	0.692 ± 0.058	0.687 ± 0.083	0.715 ± 0.032
	100	0.485 ± 0.167	0.672 ± 0.112	0.651 ± 0.132	0.699 ± 0.078
Cartpole Balance	10	0.721 ± 0.023	0.730 ± 0.009	0.729 ± 0.011	0.730 ± 0.008
	30	0.683 ± 0.057	0.711 ± 0.032	0.713 ± 0.041	$\mathbf{0.727 \pm 0.023}$
	50	0.574 ± 0.112	0.687 ± 0.084	0.695 ± 0.072	0.717 ± 0.045
	100	0.491 ± 0.171	0.663 ± 0.142	0.655 ± 0.121	0.701 ± 0.092
Cartpole Balance Sparse	10	0.705 ± 0.132	0.719 ± 0.032	0.722 ± 0.034	0.722 ± 0.026
	30	0.602 ± 0.167	0.682 ± 0.101	$0.693 \pm \mathbf{0.081}$	0.695 ± 0.093
	50	0.432 ± 0.203	0.632 ± 0.178	0.643 ± 0.162	0.652 ± 0.135
	100	0.321 ± 0.286	0.571 ± 0.232	0.589 ± 0.213	0.603 ± 0.182
Cartpole Swingup	10	0.719 ± 0.031	0.729 ± 0.011	0.723 ± 0.023	0.730 ± 0.011
	30	0.678 ± 0.062	0.698 ± 0.043	0.703 ± 0.082	0.723 ± 0.032
	50	0.563 ± 0.091	0.672 ± 0.124	0.662 ± 0.145	0.702 ± 0.078
	100	0.474 ± 0.158	0.621 ± 0.191	0.602 ± 0.191	0.672 ± 0.152
Cartpole Swingup Sparse	10	0.703 ± 0.142	0.717 ± 0.036	0.715 ± 0.052	0.720 ± 0.030
	30	0.613 ± 0.187	0.669 ± 0.121	0.671 ± 0.098	0.699 ± 0.087
	50	0.443 ± 0.213	0.621 ± 0.185	0.621 ± 0.173	0.669 ± 0.121
	100	0.403 ± 0.252	0.532 ± 0.241	0.559 ± 0.231	0.627 ± 0.162
Cheetah Run	10	0.709 ± 0.085	0.723 ± 0.031	0.721 ± 0.023	0.730 ± 0.016
	30	0.643 ± 0.131	0.689 ± 0.113	0.695 ± 0.087	0.725 ± 0.049
	50	0.527 ± 0.165	0.651 ± 0.147	0.667 ± 0.131	0.703 ± 0.092
	100	0.428 ± 0.221	0.606 ± 0.186	0.627 ± 0.183	0.676 ± 0.142
Cup Catch	10	0.711 ± 0.049	0.728 ± 0.010	0.726 ± 0.014	0.732 ± 0.008
	30	0.652 ± 0.087	0.701 ± 0.072	0.714 ± 0.061	$\mathbf{0.728 \pm 0.021}$
	50	0.534 ± 0.113	0.681 ± 0.098	0.683 ± 0.112	0.712 ± 0.053
	100	0.465 ± 0.181	0.647 ± 0.182	0.633 ± 0.172	0.692 ± 0.112

Finger Spin	10	0.712 ± 0.075	0.727 ± 0.011	0.728 ± 0.017	0.733 ± 0.009
	30	0.647 ± 0.102	0.705 ± 0.052	0.712 ± 0.057	$\mathbf{0.729 \pm 0.017}$
	50	0.517 ± 0.131	0.687 ± 0.091	0.679 ± 0.136	0.710 ± 0.062
	100	0.435 ± 0.211	0.636 ± 0.165	0.645 ± 0.185	0.684 ± 0.127
Finger Turn Easy	10	0.708 ± 0.098	0.726 ± 0.013	0.724 ± 0.018	0.732 ± 0.015
	30	0.634 ± 0.113	0.702 ± 0.051	0.705 ± 0.067	0.728 ± 0.023
	50	0.512 ± 0.157	0.681 ± 0.117	0.682 ± 0.132	0.712 ± 0.067
	100	0.412 ± 0.191	0.644 ± 0.194	0.617 ± 0.197	0.683 ± 0.148
Finger Turn Hard	10	0.704 ± 0.118	0.723 ± 0.014	0.724 ± 0.021	0.731 ± 0.011
	30	0.627 ± 0.131	0.698 ± 0.061	0.703 ± 0.073	0.725 ± 0.029
	50	0.487 ± 0.162	0.673 ± 0.112	0.673 ± 0.152	0.705 ± 0.073
	100	0.385 ± 0.231	0.623 ± 0.201	0.601 ± 0.205	0.675 ± 0.142
Hopper Hop	10	0.703 ± 0.092	0.729 ± 0.023	0.726 ± 0.023	0.731 ± 0.017
	30	0.633 ± 0.127	0.704 ± 0.087	0.701 ± 0.092	0.722 ± 0.038
	50	0.506 ± 0.191	0.664 ± 0.132	0.673 ± 0.132	0.698 ± 0.092
	100	0.407 ± 0.237	0.612 ± 0.237	0.623 ± 0.201	0.689 ± 0.143
Hopper Stand	10	0.709 ± 0.056	0.728 ± 0.021	0.725 ± 0.019	0.732 ± 0.013
	30	0.645 ± 0.112	0.699 ± 0.057	0.704 ± 0.081	0.724 ± 0.039
	50	0.523 ± 0.168	0.671 ± 0.121	0.689 ± 0.137	0.703 ± 0.080
	100	0.421 ± 0.197	0.632 ± 0.211	0.642 ± 0.189	0.692 ± 0.139
Pendulum Swingup	10	0.715 ± 0.054	0.728 ± 0.017	0.719 ± 0.023	0.732 ± 0.015
	30	0.611 ± 0.137	0.709 ± 0.054	0.699 ± 0.079	$\mathbf{0.730 \pm 0.037}$
	50	0.526 ± 0.161	0.684 ± 0.112	0.664 ± 0.146	0.721 ± 0.088
	100	0.359 ± 0.224	0.641 ± 0.198	0.612 ± 0.204	0.686 ± 0.131
Quadruped Run	10	0.718 ± 0.042	0.727 ± 0.023	0.725 ± 0.019	0.731 ± 0.013
	30	0.642 ± 0.107	0.701 ± 0.072	0.702 ± 0.074	0.726 ± 0.042
	50	0.556 ± 0.143	0.679 ± 0.135	0.683 ± 0.137	0.705 ± 0.078
	100	0.398 ± 0.217	0.644 ± 0.193	0.629 ± 0.184	0.682 ± 0.129
Quadruped Walk	10	0.719 ± 0.045	0.728 ± 0.027	0.724 ± 0.017	0.732 ± 0.011
	30	0.656 ± 0.092	0.701 ± 0.067	0.703 ± 0.072	0.723 ± 0.039
	50	0.534 ± 0.129	0.682 ± 0.114	0.685 ± 0.132	0.701 ± 0.082
	100	0.413 ± 0.212	0.654 ± 0.183	0.634 ± 0.179	0.687 ± 0.137
Reacher Easy	10	0.711 ± 0.065	0.729 ± 0.012	0.724 ± 0.013	0.732 ± 0.008
	30	0.634 ± 0.102	0.706 ± 0.061	0.705 ± 0.067	$\mathbf{0.731 \pm 0.031}$
	50	0.464 ± 0.167	0.675 ± 0.104	0.678 ± 0.121	0.720 ± 0.075
	100	0.373 ± 0.221	0.631 ± 0.175	0.614 ± 0.163	0.705 ± 0.125
Reacher Hard	10	0.712 ± 0.072	0.729 ± 0.013	0.724 ± 0.014	0.732 ± 0.010
	30	0.608 ± 0.121	0.702 ± 0.062	0.703 ± 0.072	$\mathbf{0.730 \pm 0.042}$
	50	0.501 ± 0.189	0.663 ± 0.114	0.674 ± 0.132	0.717 ± 0.082
	100	0.349 ± 0.233	0.602 ± 0.183	0.607 ± 0.172	0.701 ± 0.165
Walker Run	10	0.702 ± 0.137	0.720 ± 0.023	0.713 ± 0.027	0.729 ± 0.013
	30	0.554 ± 0.158	0.676 ± 0.067	0.682 ± 0.092	0.711 ± 0.072
	50	0.424 ± 0.213	0.613 ± 0.132	0.658 ± 0.188	0.672 ± 0.128
	100	0.323 ± 0.261	0.532 ± 0.191	0.572 ± 0.243	0.645 ± 0.175
Walker Stand	10	0.706 ± 0.087	0.728 ± 0.012	0.721 ± 0.012	0.734 ± 0.008
	30	0.598 ± 0.128	0.690 ± 0.051	0.692 ± 0.113	0.722 ± 0.067
	50	0.494 ± 0.173	0.648 ± 0.108	0.658 ± 0.162	0.701 ± 0.138
	100	0.369 ± 0.241	0.568 ± 0.172	0.572 ± 0.221	0.662 ± 0.185
Walker Walk	10	0.701 ± 0.118	$0.727 \pm \mathbf{0.015}$	$0.723 \pm \mathbf{0.022}$	0.731 ± 0.028
	30	0.612 ± 0.140	0.696 ± 0.063	0.701 ± 0.073	$\mathbf{0.730 \pm 0.034}$
	50	0.474 ± 0.181	0.664 ± 0.123	0.667 ± 0.112	0.718 ± 0.079
	100	0.371 ± 0.251	0.598 ± 0.194	0.601 ± 0.171	0.675 ± 0.142

H.2 Trial Efficiency

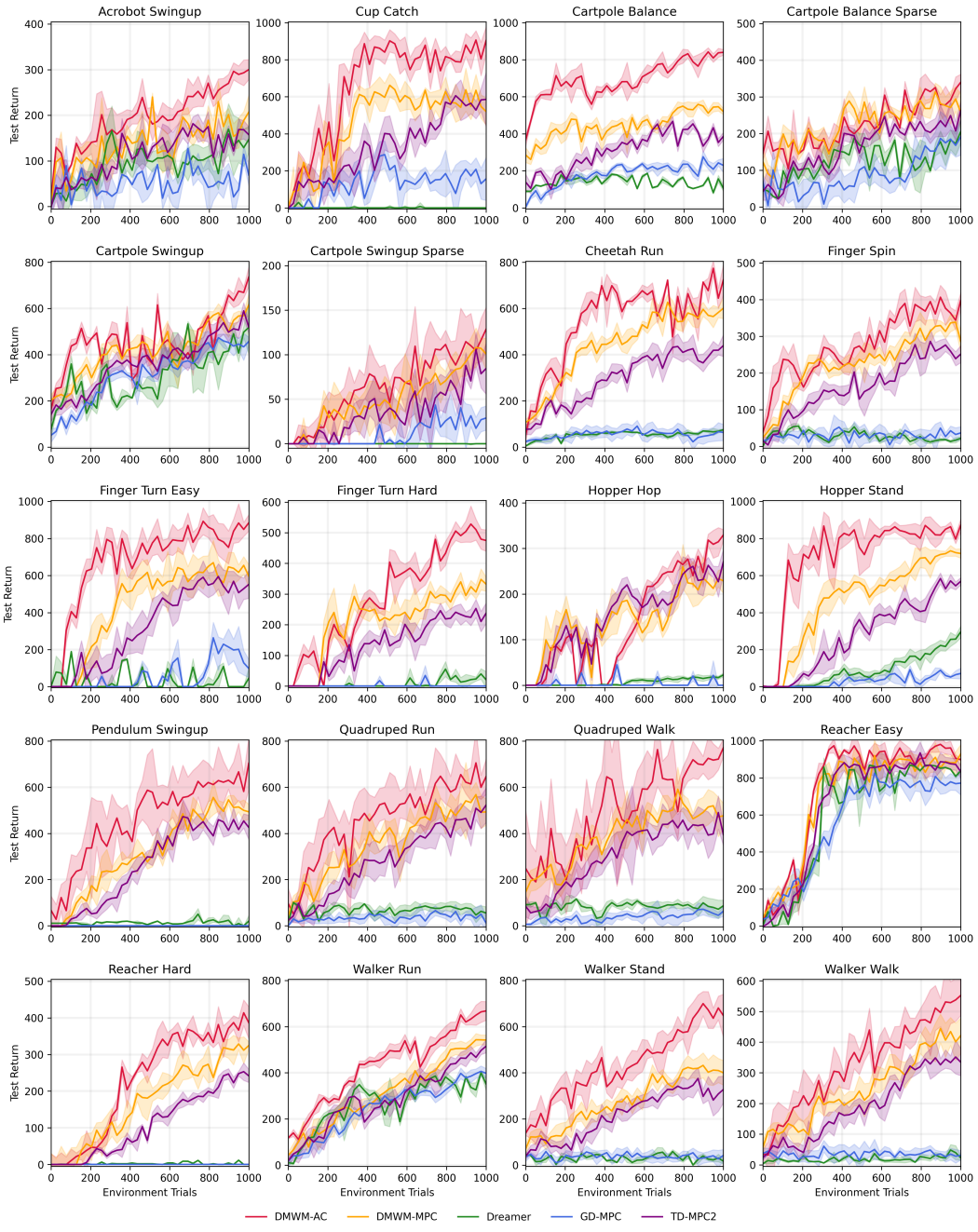

Figure 10: Performance comparison of test results on 20 DMC tasks under limited environment steps, where the standard error is shaded in the distraction setting. The horizontal axis indicates the number of environment data that is used to train the models. The vertical axis represents the average test return over 100 test episodes.

H.3 Data Efficiency

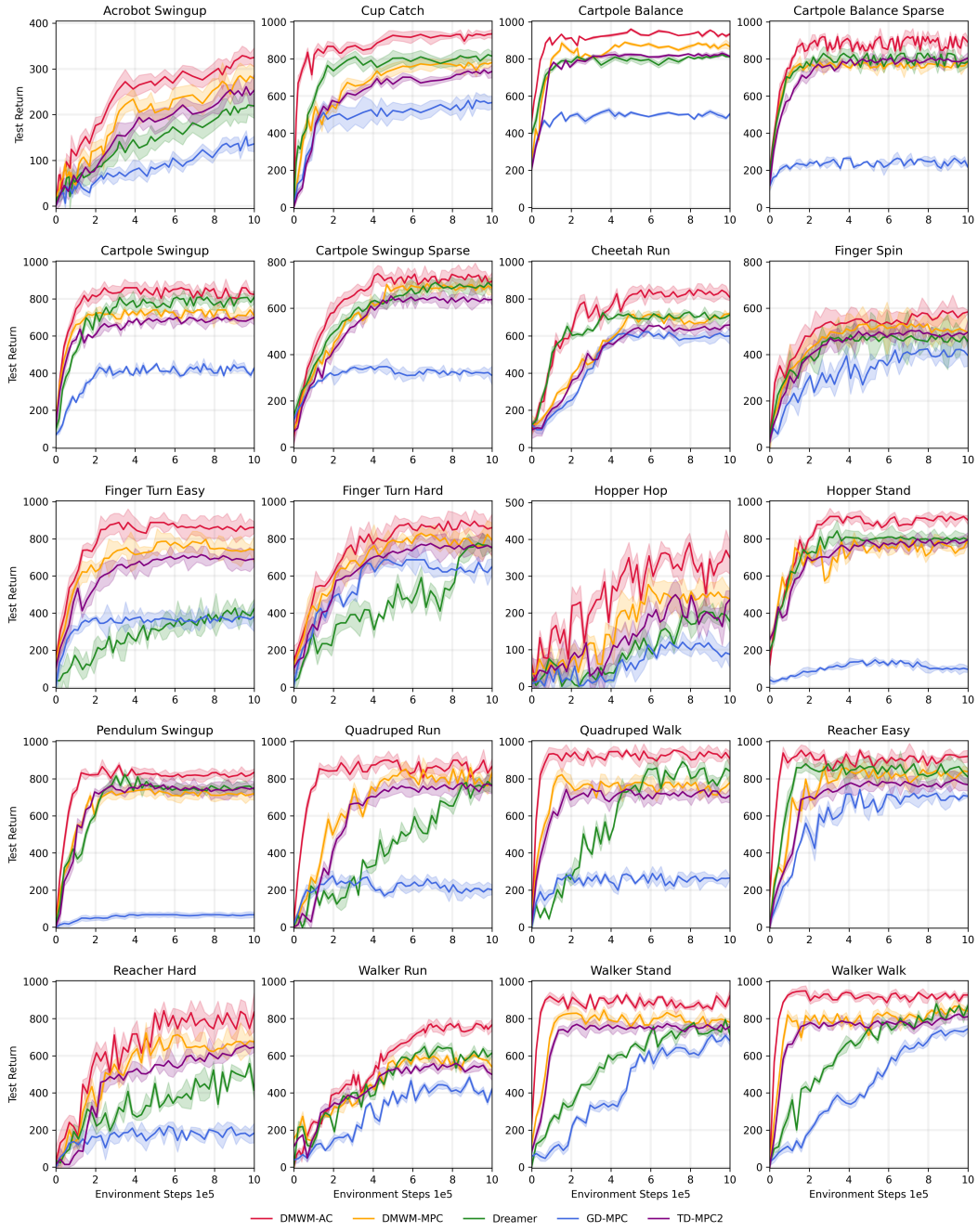

Figure 11: Performance comparison of test results on 20 DMC tasks under limited environment interactions, where the standard error is shaded in the distraction setting. The horizontal axis indicates the number of times that the models explore the environments. The vertical axis represents the average test return over 100 test episodes.

H.4 Long-term Imaginations Over Extended Horizon Size

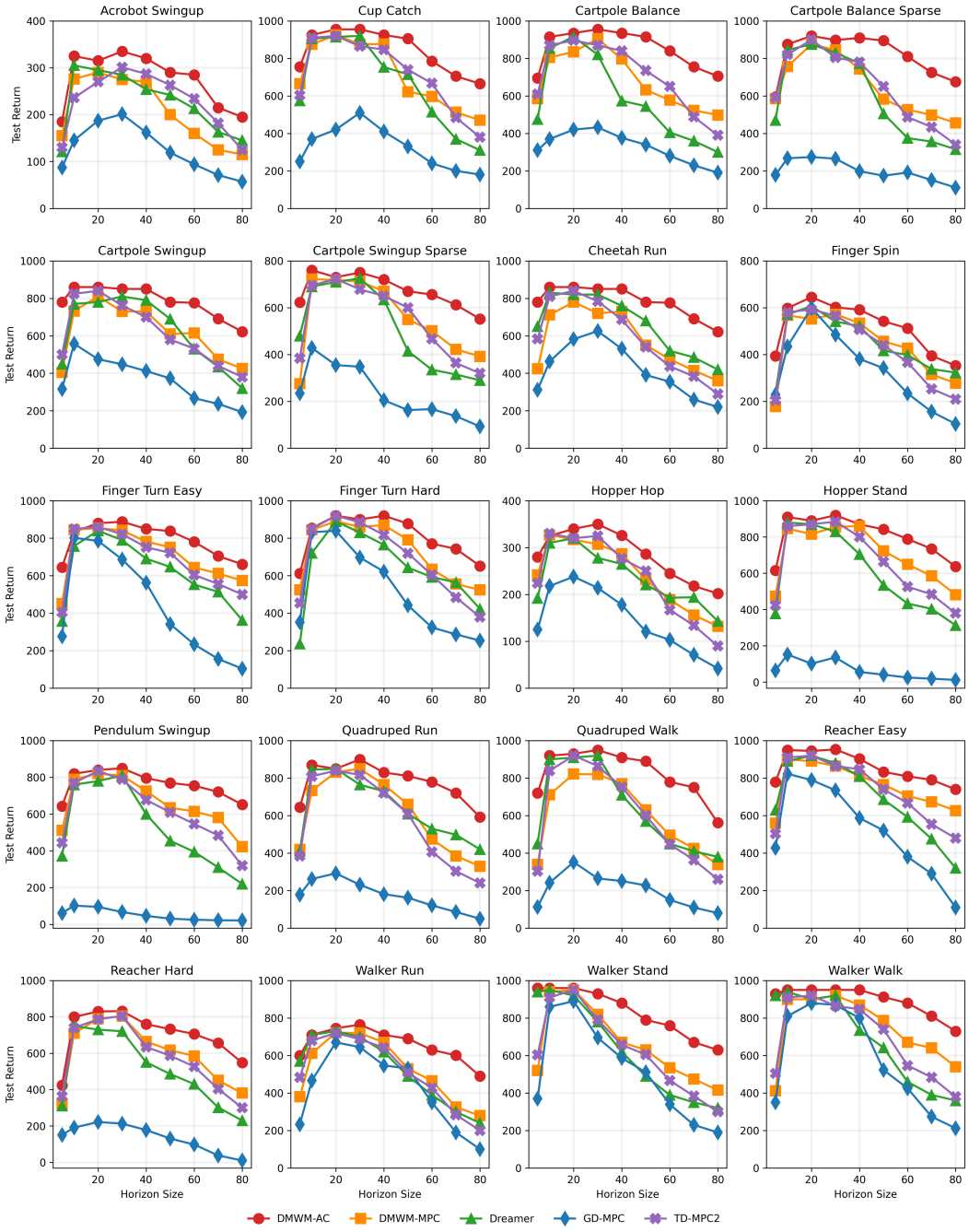

Figure 12: Performance comparison of test results on 20 DMC tasks over different horizon size of imagination. The horizontal axis indicates the horizon size of each imagination. The vertical axis represents the average test return over 100 test episodes.

H.5 Impact of Logic Inference Depth Over Extended Horizon Size

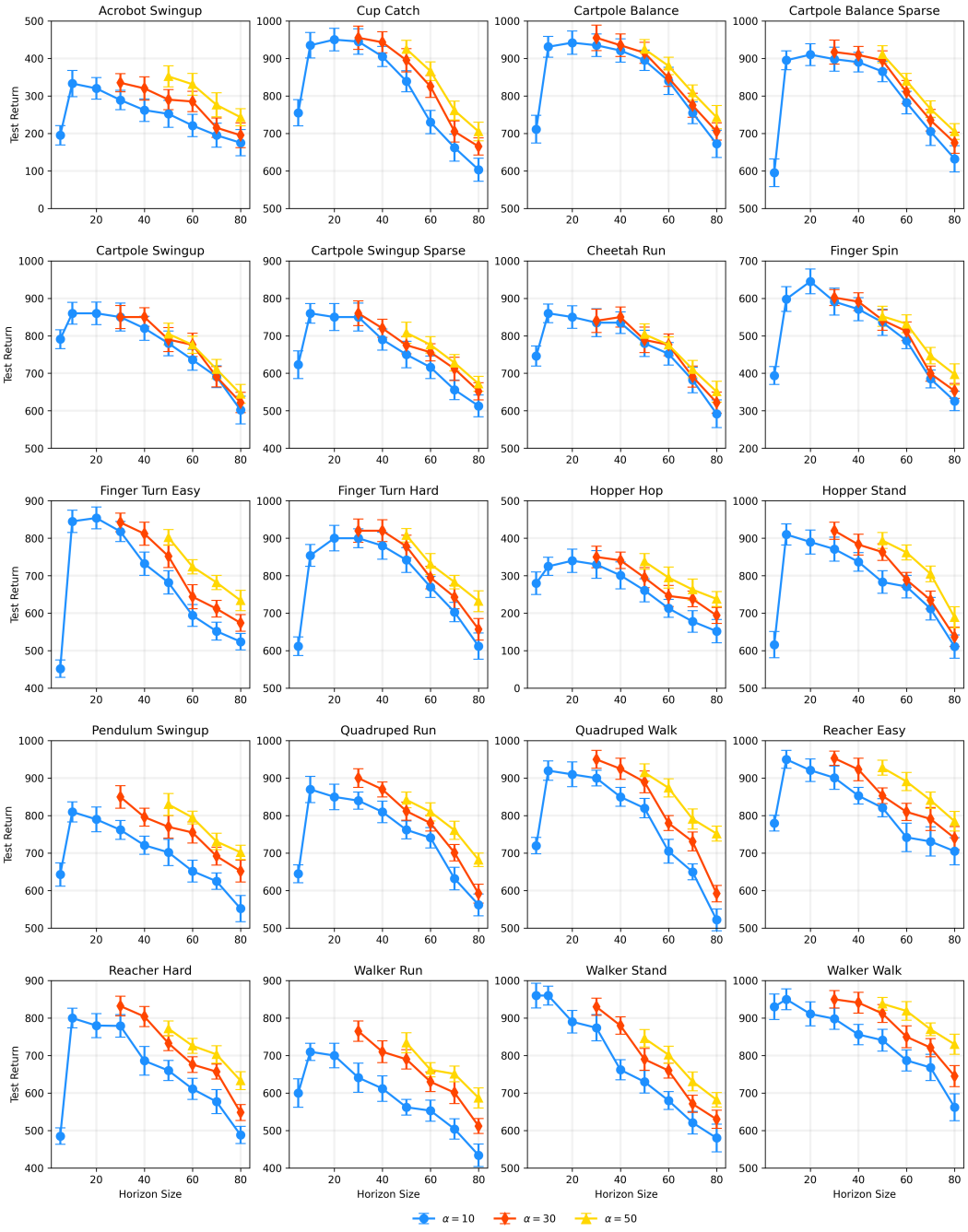

Figure 13: Performance comparison of test results on 20 DMC tasks with different logic inference depth α over extended horizon size of imagination. The horizontal axis indicates the horizon size of each imagination, and the vertical axis represents the average test return over 100 test episodes.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We explain our contributions and scope in detail in the introduction (Section 1).

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We derive the logical ELBO in Appendix C.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please refer to Section 3, Appendix D, and Appendix E to reproduce our results. We also release all the code and data.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: answerYes

Justification: We provide the code and data in the supplemental material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We described the training and test details in Section 3, Appendix D, and Appendix E.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report error bars for Figure 4-14, and in Table 2 and 6.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Section 3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Appendix A.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The creators or original owners of assets used in the paper are properly credited and the license and terms of use are explicitly mentioned and are properly respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in our research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.