

# MrCoM: A Meta-Regularized World-Model Generalizing Across Multi-Scenarios

Xuantang Xiong<sup>1,4,5\*</sup>, Ni Mu<sup>2\*</sup>, Runpeng Xie<sup>1</sup>, Senhao Yang<sup>1</sup>, Yaqing Wang<sup>1</sup>, Lexiang Wang<sup>1</sup>, Yao Luan<sup>2</sup>, Siyuan Li<sup>3</sup>, Shuang Xu<sup>1</sup>, Yiqin Yang<sup>1†</sup>, Bo Xu<sup>1†</sup>

<sup>1</sup>The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China

<sup>2</sup>Department of Automation, Tsinghua University,

<sup>3</sup>Faculty of Computing, Harbin Institute of Technology

<sup>4</sup>Tencent Robotics X & Futian Laboratory, Shenzhen

<sup>5</sup>Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences

## Abstract

Model-based reinforcement learning (MBRL) is a crucial approach to enhance the generalization capabilities and improve the sample efficiency of RL algorithms. However, current MBRL methods focus primarily on building world models for single tasks and rarely address generalization across different scenarios. Building on the insight that dynamics within the same simulation engine share inherent properties, we attempt to construct a unified world model capable of generalizing across different scenarios, named Meta-Regularized Contextual World-Model (MrCoM). This method first decomposes the latent state space into various components based on the dynamic characteristics, thereby enhancing the accuracy of world-model prediction. Further, MrCoM adopts meta-state regularization to extract unified representation of scenario-relevant information, and meta-value regularization to align world-model optimization with policy learning across diverse scenario objectives. We theoretically analyze the generalization error upper bound of MrCoM in multi-scenario settings. We systematically evaluate our algorithm’s generalization ability across diverse scenarios, demonstrating significantly better performance than previous state-of-the-art methods.

## 1 Introduction

Reinforcement learning, a key algorithm for solving decision-making problems, has achieved significant success in domains such as games (Silver et al. 2018; Berner et al. 2019), robotic control (Akkaya et al. 2019), autonomous driving (Kiran et al. 2021), and even large language model tuning (Sun et al. 2023). One crucial requirement for broader RL applications is to improve the sample efficiency, and recent advances in model-based approaches have emerged as a promising solution by learning a world-model.

Most model-based RL methods mainly focus on single-task settings, where a dedicated world model is trained for each individual task. Recent studies have demonstrated the robustness of world model hyperparameters across diverse

scenarios, that is, maintaining identical training configurations to train world models under varying conditions. However, they seldom investigate cross-scenario generalization of the world models themselves. This critical gap prevents the deployment of a unified world model across multiple environments, which could substantially reduce training costs. Therefore, this naturally leads to the following question:

*How to learn a unified world-model capable of generalizing across multi-scenarios?*

In this work, we aim to provide an effective solution to construct the unified world-model. First, we decompose the latent state space into various components based on the dynamic characteristics, thereby enhancing the accuracy of world-model prediction. Further, we propose the meta-state regularization mechanism to extract unified representation of scenario-relevant information, and meta-value regularization to align world-model optimization with policy learning across diverse scenario objectives. We name our method as Meta-Regularized Contextual World-Model (MrCoM). We conduct the theoretical analysis for the model’s generalization capability and derive the upper bound of the generalization error of our method.

For the experiments, we select the Mujoco-based MDC (Tassa et al. 2018; Todorov, Erez, and Tassa 2012), a simulation engine for robotic actions, as the benchmark. We construct the meta-scenarios set by changing the scenario dynamics and objective (e.g., robot’s limb size and length). Through extensive experimental results, we demonstrate the advantage of our method over current state-of-the-art approaches.

In summary, our contributions are threefold:

1. We propose MrCoM to learn a unified world-model based on the meta-state and meta-value regularization mechanism.
2. We conduct the theoretical analysis for MrCoM, deriving the generalization error upper bound.
3. We conduct comprehensive experiments demonstrating that, compared to other baselines, the world model learned by MrCoM can generalize across diverse scenarios.

\*These authors contributed equally.

†Corresponding Authors.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

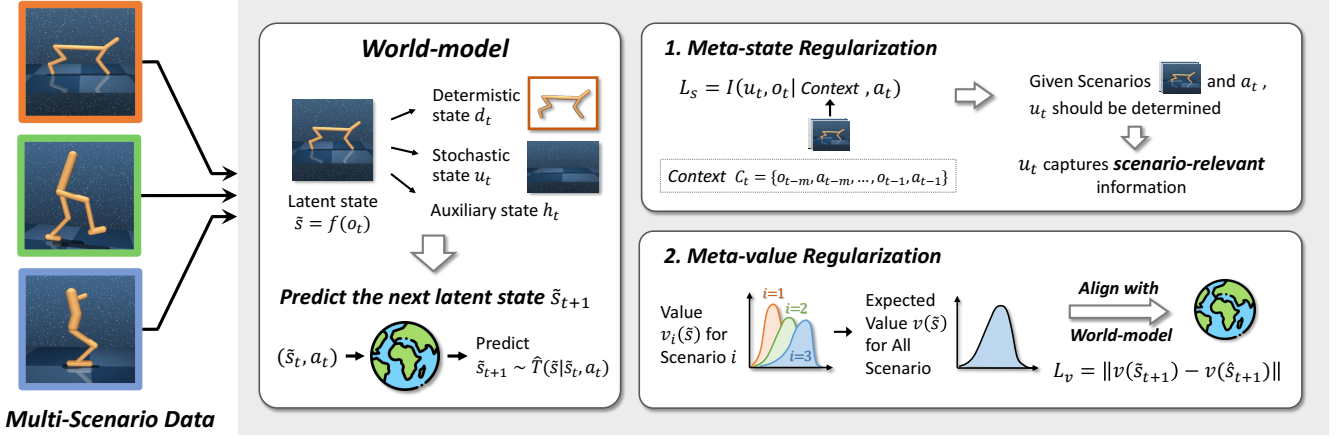


Figure 1: Framework of MrCoM, which merges multi-scenario data into a unified world-model. The meta-state regularization extracts scenario-relevant information, and meta-value regularization aligns world-model optimization with policy learning.

## 2 Problem Formulation

### 2.1 POMDP

A Partially Observable Markov Decision Process (POMDP) is an extension of a Markov Decision Process (MDP) used to model decision-making problems under uncertainty and incomplete information. It can be represented as a tuple  $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, r, T, \rho_0, \gamma, \mathcal{O}, \Omega\}$ . Here,  $\mathcal{S} \in \mathbb{R}^n$  denotes the state space,  $\mathcal{A} \in \mathbb{R}^m$  represents the action space,  $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$  is the reward function,  $T : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$  defines the dynamics of state transitions,  $\rho_0 \in \mathcal{S}$  represents the initial state distribution,  $\gamma \in [0, 1]$  is the discount factor,  $\mathcal{O} \in \mathbb{R}^k$  denotes the observation space, and  $\Omega : \mathcal{S} \rightarrow \mathcal{O}$  is observation function representing the mapping from states to observations.

The policy of an agent, denoted as  $\pi_\phi : \mathcal{O} \rightarrow \mathcal{A}$ , is parameterized by  $\phi$ . The discounted infinite horizon return is defined as  $G_\pi = \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)$ , where  $t$  represents the time step and  $a$  denotes the action taken by the agent. The objective is to find the optimal policy parameter  $\phi^*$  that maximizes the cumulative discounted reward:

$$\phi^* = \arg \max_{\phi} \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t r(s_t, \pi_\phi(o_t)) \right] \quad (1)$$

### 2.2 Meta-POMDP

In this work, we consider finding a policy that maximizes expected return in a Meta-POMDP, defined as  $\mathcal{M} = \{\mathcal{M}^i\}_{i=1}^N = \{\mathcal{S}^i, \mathcal{A}^i, r^i, T^i, \rho_0^i, \gamma, \mathcal{O}^i, \Omega^i\}_{i=1}^N$ , with a set of scenario indices  $1, \dots, N$ . Each scenario  $i$  is randomly sampled from the meta-scenario set  $\mathcal{T}^i \sim p(\mathcal{T})$ , where  $\mathcal{T}^i$  corresponds to a particular POMDP tuple  $\mathcal{M}^i$ . In this work, we assume that each scenario  $i$  presents a different state space  $\mathcal{S}^i$ , action space  $\mathcal{A}^i$ , reward function  $r^i$ , dynamics  $T^i$  and observation space  $\mathcal{O}^i$ . We formulate the policy optimization problem as finding a policy that maximizes expected return over all the scenarios:

$$\phi^* = \arg \max_{\phi} \mathbb{E}_{\mathcal{T}^i \sim p(\mathcal{T})} \left[ \sum_{t=0}^{\infty} \gamma^t r^i(s_t^i, \pi_\phi(o_t^i)) \right] \quad (2)$$

**Comparison with other setting:** Unlike prior works, we aim to learn a unified world-model effective across varying state spaces  $\mathcal{S}^i$ , action spaces  $\mathcal{A}^i$ , dynamics  $T^i$ , reward functions  $r^i$ , and observation functions  $\mathcal{O}^i$ . For example, CaDM (Lee et al. 2020) seeks to construct a world-model capable of generalizing across diverse dynamics  $T^i$ , which is represented as  $\{\mathcal{M}^i\}_{i=1}^N = \{\mathcal{S}, \mathcal{A}, r, T^i, \rho_0, \gamma, \mathcal{O}, \Omega\}_{i=1}^N$ . Dreamerv3 (Hafner et al. 2023) is designed to construct a world-model for individual tasks, where  $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, r, T, \rho_0, \gamma, \mathcal{O}, \Omega\}$ . MAMBA (Rimon et al. 2024) seeks to construct a world model effective across diverse scenarios with varying dynamics  $T^i$  and reward functions  $r^i$ , which is denoted as  $\{\mathcal{M}^i\}_{i=1}^N = \{\mathcal{S}, \mathcal{A}, r^i, T^i, \rho_0, \gamma, \mathcal{O}, \Omega\}_{i=1}^N$ .

### 2.3 Model-Based RL

Model-Based Reinforcement Learning (MBRL) constructs an approximate dynamics model  $\hat{T}_\theta(s'_t | s_t, a_t)$ , where the model parameters  $\theta$  are optimized based on observed data  $\mathcal{D}$  through maximizing the likelihood:

$$\theta^* = \arg \max_{\theta} \sum_{(s, a, s') \sim \mathcal{D}} \log \hat{T}_\theta(s'_t | s_t, a_t) \quad (3)$$

In the MBRL framework, policy learning can take advantage of real data and simulated data generated by the approximated dynamics model  $\hat{T}_\theta(s'_t | s_t, a_t)$ , which enhances sample efficiency and reduces dependency on real environment interactions.

## 3 Method

In this section, we propose a novel method to learn a unified world-model capable of generalizing across multi-scenarios,

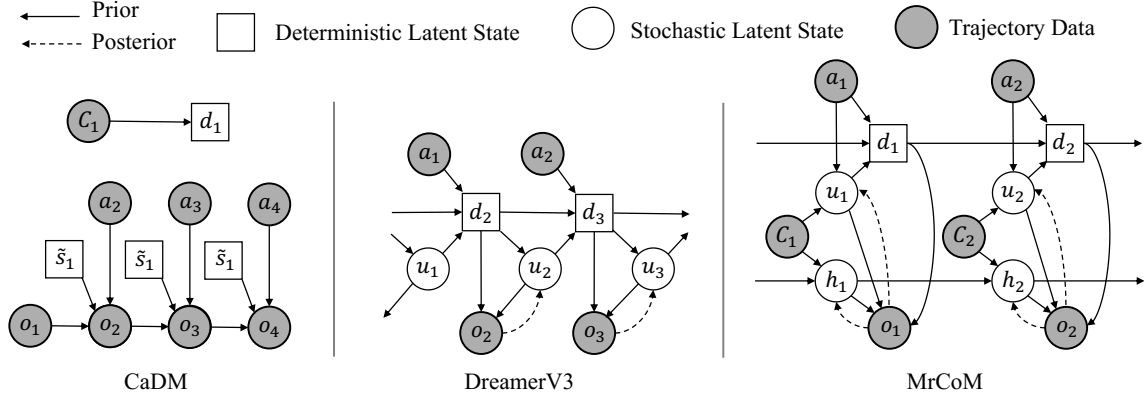


Figure 2: Model architecture of algorithms. Compared to other methods, MrCoM implements a refined partitioning of the latent state space. In this Figure,  $\tilde{s}$  denotes the latent state,  $d$  the deterministic latent state,  $u$  the stochastic latent state, and  $h$  the auxiliary state. In MrCoM, the latent state  $\tilde{s}_t$  is composed of the concatenation of  $u_t$ ,  $d_t$ , and  $h_t$ .

named **Meta-Regularized Contextual World-Model (MrCoM)**. We first present the model architecture, which decomposes the latent state space based on dynamic characteristics, enhancing prediction accuracy of the world model. Subsequently, we separately introduce meta-state regularization and meta-value regularization mechanisms. These two mechanisms extract scenario-relevant information and align model-policy optimization, achieving cross-scenario generalization of the world model. The framework of MrCoM is shown in Figure 1.

To optimize the world-model across multiple scenario, we inject contextual information into state prediction with the transformer architecture. During inference initialization, the contextual information will activate the model’s domain-specific predictive capability. The contextual information is defined as:

$$C_t = \{o_{t-m}, a_{t-m}, \dots, o_{t-1}, a_{t-1}\}, \quad (4)$$

where  $m$  denotes the context length. When  $t < m$ , scenario-related expert trajectories are used as contextual information, with the first-in-first-out mechanism to update the generated latest trajectories.

### 3.1 Model Architecture

Complex environments often exhibit stochastic disturbances, making precise modeling challenging. To tackle this issue, we first assume a mapping function  $f^*$  that projects environmental states into a unified latent space  $\tilde{\mathcal{S}}$ :

**Assumption 1 (Unified Mapping).** *For all scenarios, states and actions can be mapped by a function  $f^*$  to a unified latent state space  $\tilde{\mathcal{S}}$ . For simplicity, the unified mapping of the action space is not explicitly expressed here. Formally, the state unified mapping can be described as:*

$$\begin{aligned} \forall \mathcal{T}^i \sim p(\mathcal{T}), \quad \exists f^* : \mathcal{O}^i \rightarrow \tilde{\mathcal{S}} \\ \text{s.t., } f^*(o) = \tilde{s}, \quad \forall o \in \mathcal{O}^i, \text{ where } \tilde{s} \in \tilde{\mathcal{S}}. \end{aligned} \quad (5)$$

We assume the unified latent space  $\tilde{\mathcal{S}}$  can be decomposed into stochastic  $u_t$  and deterministic  $d_t$  components. We de-

rive the stochastic component  $u_t$  from contextual information  $C_t$ , which follows the Gaussian distribution to capture the uncertainty in state transitions:

$$p_\theta^{u_t} = p_\theta(u_t | C_t, a_t) \quad (6)$$

We compute the deterministic component  $d_t$  based on historical information  $d_{t-1}$  and the current stochastic input  $u_t$ , thereby preserving historical stochastic information:

$$p_\theta^{d_t} = p_\theta(d_t | d_{t-1}, u_t, a_t) \quad (7)$$

Since we aim to learn a unified world model applicable across multiple scenarios, we introduce an additional auxiliary state  $h_t$  conditioned on contextual information  $C_t$  to enhance prediction accuracy and minimize information loss. This auxiliary state  $h_t$  follows a Gaussian distribution:

$$p_\theta^{h_t} = p_\theta(h_t | C_t, h_{t-1}) \quad (8)$$

Based on the above analysis, the latent state  $\tilde{s}_t$  is composed of the concatenation of  $u_t$ ,  $d_t$ , and  $h_t$ . For the posterior distribution, we utilize the original observation  $o_t$  as the additional input and reconstruct the original observation  $o_t$  conditioned on the stochastic component  $u_t$ , deterministic component  $d_t$ , and auxiliary state  $h_t$ :

$$\begin{aligned} q_\theta^{u_t} &= q_\theta(u_t | C_t, a_t, o_t) \\ q_\theta^{h_t} &= q_\theta(h_t | C_t, h_{t-1}, o_t) \\ p_\theta^{o_t} &= p_\theta(o_t | u_t, d_t, h_t) \end{aligned} \quad (9)$$

Based on the above analysis, our world-model  $\hat{T}_\theta$  comprises the prior encoders  $p_\theta^{u_t}, p_\theta^{d_t}, p_\theta^{h_t}$ , the posterior encoders  $q_\theta^{u_t}, q_\theta^{h_t}$ , and the reconstruction decoder  $p_\theta^{o_t}$ . Based on the variational bound, we have the following loss function:

$$\begin{aligned} \mathcal{L}_{\text{var}} &= \mathcal{L}_{\text{KL}} + \mathcal{L}_{\text{recon}} \\ &= \mathbb{E}_{\mathcal{D}} \left[ \text{KL}(p_\theta^{u_t} \| q_\theta^{u_t}) + \text{KL}(p_\theta^{h_t} \| q_\theta^{h_t}) - \log p_\theta^{o_t} \right], \end{aligned} \quad (10)$$

where  $\mathcal{L}_{\text{KL}}$  denotes the KL divergence between the prior and posterior distributions of latent states, and  $\mathcal{L}_{\text{recon}}$  represents the reconstruction error of original observations. Please refer to Figure 2 for the model architecture.

### 3.2 Meta-State Regularization

Original observations  $o_t$  typically contain both scenario-relevant and scenario-irrelevant information. Scenario-relevant information refers to state variables determined by actions, whereas scenario-irrelevant information denotes action-invariant state elements. Crucially, scenario-relevant information correlates with scenario objectives and plays a vital role when the world model operates across various domains. Therefore, to learn a unified world-model applicable across multiple scenarios, we must extract scenario-relevant information from original observations.

Inspired by Denoised MDPs, we find that the transition of scenario-relevant states can be largely determined given current actions  $a_t$  and contextual information  $C_t$ . Consequently, to filter out scenario-irrelevant information from original observations, we optimize the world-model by minimizing mutual information  $I(u_t, o_t | C_t, a_t)$  between stochastic components  $u_t$  and original observations  $o_t$ . Specifically, we can optimize the upper bound for the mutual information by introducing an additional variational approximation distribution  $q(u_t | C_t, a_t)$  (Poole et al. 2019):

$$\begin{aligned} I(u_t, o_t | C_t, a_t) &\equiv \mathbb{E}_{p(C_t, a_t)} \mathbb{E}_{p(u_t, o_t)} \left[ \log \frac{p(u_t | o_t, C_t, a_t)}{p(u_t | C_t, a_t)} \right] \\ &= \mathbb{E}_{p(C_t, a_t)} \mathbb{E}_{p(u_t, o_t)} \left[ \log \frac{p(u_t | o_t, C_t, a_t) q(u_t | C_t, a_t)}{q(u_t | C_t, a_t) p(u_t | C_t, a_t)} \right] \\ &= \mathbb{E}_{p(C_t, a_t)} \mathbb{E}_{p(u_t, o_t)} \left[ \log \frac{p(u_t | C_t, a_t, o_t)}{q(u_t | C_t, a_t)} \right] \\ &\quad - \text{KL}(p(u_t | C_t, a_t) \| q(u_t | C_t, a_t)) \\ &\leq \mathbb{E}_{p(C_t, a_t)} \mathbb{E}_{p(o_t)} [\text{KL}(p(u_t | C_t, a_t, o_t) \| q(u_t | C_t, a_t))] \end{aligned} \quad (11)$$

When the prior distribution approaches the posterior distribution, we have  $\text{KL}(p_\theta(u_t | C_t, a_t, o_t) \| q_\theta(u_t | C_t, a_t)) \approx \text{KL}(p_\theta(u_t | C_t, a_t) \| q_\theta(u_t | C_t, a_t, o_t))$ , and this divergence serves as a tight upper bound on mutual information. Therefore, based on the above analysis, we can minimize the following loss to extract scenario-relevant information into world-model:

$$\mathcal{L}_s = \mathbb{E}_{\mathcal{D}} [\text{KL}(p_\theta(u_t | C_t, a_t) \| q_\theta(u_t | C_t, a_t, o_t))] \quad (12)$$

### 3.3 Meta-Value Regularization

In the multi-scenario setting, the vast state space makes it challenging for the world-model to achieve accurate predictions universally. To address this issue, we let the world model focus primarily on regions involved in value function updates, which enhances the precision of value estimation and thereby improves policy performance.

To this end, we propose the meta-value regularization mechanism to align world-model optimization with policy learning. Specifically, we define the meta-value  $v_\psi(\tilde{s}_t)$ , which refers to the shared value extracted from latent states  $\tilde{s}_t$  across various scenario distributions. We optimize  $v_\psi(\tilde{s}_t)$  via the following loss function:

$$\mathcal{L}_{\text{value}} = \mathbb{E}_{\mathcal{T}^i \sim p(\mathcal{T})} \left[ \|v_{\psi_i}(\tilde{s}_t) - v_\psi(\tilde{s}_t)\|^2 \right], \quad (13)$$

where  $v_{\psi_i}(\tilde{s}_t)$  represents the value function for a specific scenario  $\mathcal{T}^i$  and  $\tilde{s}_t = \{u_t, d_t, h_t\}$ . We update  $v_{\psi_i}$  based on the following Bellman equation:

$$\mathcal{L}_{\text{value}_i} = \mathbb{E}_{\mathcal{T}^i \sim p(\mathcal{T})} \left[ \|v_{\psi_i}(\tilde{s}_t) - (r_t + \gamma v_{\psi_i}(\tilde{s}_{t+1}))\|^2 \right]. \quad (14)$$

Further, we update the world-model based on the learned meta-value  $v_\psi(\tilde{s}_t)$ :

$$\mathcal{L}_v = \mathbb{E}_{\mathcal{T}^i \sim p(\mathcal{T})} \left[ \|v_\psi(\tilde{s}_{t+1}) - \hat{T}_\theta(\hat{s}_{t+1} | \tilde{s}_t, a_t) v_\psi(\hat{s}_{t+1})\|^2 \right], \quad (15)$$

where  $\hat{s}_{t+1}$  represents the latent state predicted by the world-model  $\hat{T}_\theta$ .

### 3.4 Implementation Details

Based on the analysis in Section 3.1, 3.2, 3.3, the total loss function of MrCoM is defined as follows:

$$\mathcal{L}_{\text{MrCoM}} = \lambda_{\text{var}} \mathcal{L}_{\text{var}} + \lambda_s \mathcal{L}_s + \lambda_v \mathcal{L}_v. \quad (16)$$

where  $\lambda_{\text{var}}$ ,  $\lambda_s$ ,  $\lambda_v$  are hyperparameters that balance the weights of the loss terms.

The overall procedure of the algorithm comprises two phases: world-model training and scenario adaptation. In the training phase, we collect data across diverse scenarios using the behavioral policy  $\pi_{\phi_i}$ , followed by training the world model according to Equation 16. During scenario adaptation, given target scenario  $\mathcal{T}^{\text{target}}$ , we perform rollouts based on the trained world model  $\hat{T}_\theta$  to augment the dataset, which is then used to train a standard RL algorithm. The detailed algorithmic procedure and hyperparameters are provided in Appendix A and Appendix D.

## 4 Theoretical Analysis

This section establishes the theoretical foundation of MrCoM's framework and provides insights into the proposed objective. Firstly, we introduce the following assumption:

**Assumption 2** (Dynamics Homogeneity). *For states from different scenarios that map to the same state  $\tilde{s}_t \in \tilde{\mathcal{S}}$ , their state transition probabilities under the same action are identical. This is represented as:*

$$\begin{aligned} \text{if } f^*(o_t^i) = f^*(o_t^j) = \tilde{s}_t, \quad \forall o_t^i \in \mathcal{O}^i, \forall o_t^j \in \mathcal{O}^j, \\ \text{then } T^i(s_t^i, a_t) = T^j(s_t^j, a_t), \quad \forall a_t \in \mathcal{A}. \end{aligned} \quad (17)$$

Under the Assumption 1 and 2, it can be inferred that there exists a shared latent transition function  $\tilde{T}$  across scenarios, defined on the latent space  $\tilde{\mathcal{S}}$ :

$$\begin{aligned} \tilde{T}(\tilde{s}' | \tilde{s}, a) &= T^i(s' | s, a) \\ \forall \mathcal{T}^i \sim p(\mathcal{T}), s \in \mathcal{S}^i, a \in \mathcal{A}^i, \exists \tilde{T} : \tilde{\mathcal{S}} \times \mathcal{A} &\rightarrow \tilde{\mathcal{S}}, \end{aligned} \quad (18)$$

where  $T^i$  is the true transition function on the scenario  $i$ . Then, we define the dynamic error as follows:

$$\max_i \mathbb{E}_{\mathcal{T}^i \sim p(\mathcal{T})} D_{\text{TV}} \left( \tilde{T}(\tilde{s}' | \tilde{s}, a) \| T^i(s' | s, a) \right) \leq \epsilon_{\mathcal{T}} \quad (19)$$

Next, we define the state representation error as follows:

$$\max_i \mathbb{E}_{\mathcal{T}^i \sim p(\mathcal{T})} |f(o) - \tilde{s}| \leq \epsilon_S, \quad (20)$$

where  $f$  is the representation function and  $\tilde{s}$  is the true latent state. Based on the above definitions, we first consider the dynamics model error bound under the state representation:

**Lemma 3.** *Given the state representation error  $\epsilon_S$  and dynamics model error  $\epsilon_T$ , the upper bound of the dynamics error under the state representation is:*

$$D_{TV} \left( \tilde{T}(f(o') | f(o), a) \| T^i(s' | s, a) \right) \leq \epsilon_T + C_T \cdot \epsilon_S. \quad (21)$$

Here,  $C_T = \max_s \nabla_s \sum_a T(s' | s, a)$  denotes the maximum derivative of the dynamics function concerning  $s$ , representing the sensitivity of state changes to the state transition function.

*Proof.* Please refer to Appendix B.1 for the detailed proof.  $\square$

Then, we derive the policy difference bound under state representation as follows:

**Lemma 4.** *Given policy difference in the state space  $\max_s D_{TV}(\pi_1(a | s) \| \pi_2(a | s)) \leq \epsilon_\pi$ , the upper bound of the policy difference under the state representation error  $\epsilon_S$  is:*

$$D_{TV}(\pi_1(a | f(o)) \| \pi_2(a | s)) \leq \epsilon_\pi + \frac{1}{2} C_\pi \cdot \epsilon_S \quad (22)$$

Here,  $C_\pi = \max_s \nabla_s \sum_a \pi_1(a | s)$  represents the maximum sensitivity of the policy function to state variations.

*Proof.* Please refer to Appendix B.2 for the detailed proof.  $\square$

Next, we consider the upper bound of the performance difference under two distinct dynamics and policies as follows:

**Lemma 5.** *Given dynamics model error  $\epsilon_T$ , the policy difference  $\epsilon_\pi$  and the maximum reward  $R$ , the performance difference has the following upper bound:*

$$|G^1(\pi_1) - G^2(\pi_2)| \leq \frac{2R\gamma(\epsilon_\pi + \epsilon_T)}{(1-\gamma)^2} + \frac{2R\epsilon_\pi}{1-\gamma} \quad (23)$$

Here,  $G^1(\pi_1)$  denotes the performance of  $\pi_1$  under dynamics  $T^1$ , and  $G^2(\pi_2)$  denotes the performance of  $\pi_2$  under dynamics  $T^2$ .

*Proof.* Please refer to Appendix B.3 for the detailed proof.  $\square$

Let  $\tilde{G}^i(\pi)$  denote the performance of  $\pi(f(o))$  under the dynamics  $\tilde{T}^i(f(o), a)$ . Let  $\tilde{G}_\theta(\pi)$  denote the performance under the learned world-model  $\tilde{T}_\theta$ . By combining Lemma 3 and Lemma 4 with Lemma 5, we can derive the generalization error bound in the Meta-POMDP setting:

**Theorem 6.** *Given dynamics model error  $\epsilon_T$ , policy difference  $\epsilon_\pi$ , and the state representation error  $\epsilon_S$ , the upper bound of the generalization error on the Meta-POMDP is:*

$$\left| \tilde{G}^i(\pi) - \tilde{G}_\theta(\pi) \right| \leq \frac{R\gamma[4\epsilon_\pi + 2\epsilon_T + (C_\pi + 2C_T)\epsilon_S]}{(1-\gamma)^2} + \frac{2R(2\epsilon_\pi + C_\pi\epsilon_S)}{1-\gamma}. \quad (24)$$

*Proof.* Please refer to Appendix B.4 for the detailed proof.  $\square$

From the above theory, the upper bound comprises three primary error sources: dynamics model error  $\epsilon_T$ , state representation error  $\epsilon_S$ , and policy difference  $\epsilon_\pi$ . We find that these three errors contribute linearly to the generalization error bound of the model. In MrCoM, we minimize the dynamics error through latent state factorization, reduce the representation error via meta-state regularization, and mitigate the policy error by meta-value regularization. By integrating these three components, we effectively reduce the generalization error of MrCoM on Meta-POMDPs.

## 5 Experiments

We design our experiments to answer the following questions: (Q1) How does MrCoM perform in the multi-scenario setting? (Q2) How does MrCoM perform when the dynamics function changes? (Q3) How does MrCoM perform when the observation space changes? (Q4) What is the contribution of each proposed technique in MrCoM?

### 5.1 Setup

**Domains.** We evaluate our method and baselines on the DMControl scenarios (Tassa et al. 2018; Todorov, Erez, and Tassa 2012). Specifically, we select three scenarios on DMControl: hopper, walker, and cheetah. To change dynamics  $T$ , we modify the environment parameters by uniformly sampling the torso length and size within an interval of  $\alpha\%$  around their default values. To change reward  $r$ , we adjust the agent’s target speed as follows. Let  $v_{\max}$  represent the fastest speed of the agent under default dynamics settings. The scenario objective  $v_i$  is uniformly sampled from the interval  $[0, \beta\% \cdot v_{\max}]$ .

**Multi-scenario.** In our experiments, we train world-model capable of generalizing across diverse scenarios. Specifically, we merge data from Hopper, Walker and Cheetah to train the world-model. Then, we select one scenario (e.g., Hopper) for scenario adaptation. We conduct both in-distribution and out-of-distribution evaluations, with detailed specifications provided in the section 5.2.

**Baselines.** To validate the effectiveness of our method, we choose the following state-of-the-art algorithms as baselines: DreamerV3 (Hafner et al. 2023) features a context-augmented RSSM architecture, demonstrating strong cross-domain adaptability within a unified framework. MAMBA (Rimon et al. 2024) extends DreamerV3 to

	(a) Train $\alpha = 5, \beta = 20$ , Evaluate $\alpha = 5, \beta = 20$				(b) Train $\alpha = 5, \beta = 20$ , Evaluate $\alpha = 10, \beta = 50$			
	MAMBA	DreamerV3	CaDM	Ours	MAMBA	DreamerV3	CaDM	Ours
Hopper	43.1±12.7	52.3±12.6	41.4±18.5	<b>57.7±14.2</b>	35.1±18.2	43.3±11.2	38.8±13.9	<b>52.0±9.9</b>
Walker	47.1±6.8	56.7±12.0	55.2±15.8	<b>60.8±7.4</b>	29.4±12.5	37.5±11.1	49.7±8.2	<b>53.2±8.7</b>
Cheetah	52.3±12.5	<b>53.7±17.4</b>	47.1±13.2	48.6±12.7	<b>49.5±14.0</b>	43.2±13.1	42.7±13.1	45.2±13.9
	(c) Train $\alpha = 10, \beta = 50$ , Evaluate $\alpha = 10, \beta = 50$				(d) Train $\alpha = 10, \beta = 50$ , Evaluate $\alpha = 20, \beta = 100$			
	MAMBA	DreamerV3	CaDM	Ours	MAMBA	DreamerV3	CaDM	Ours
Hopper	38.3±15.2	42.1±11.6	42.1±13.1	<b>52.1±12.2</b>	21.3±15.6	36.6±25.2	21.2±11.3	<b>48.1±13.8</b>
Walker	29.3±18.6	38.2±8.9	29.6±14.9	<b>40.1±13.1</b>	16.5±21.2	30.0±11.9	26.1±16.2	<b>47.5±10.3</b>
Cheetah	28.7±13.1	40.1±16.2	37.1±10.8	<b>45.8±9.6</b>	27.4±11.6	41.7±21.3	41.2±18.6	<b>44.7±11.2</b>

Table 1: Experimental results on the *multi-scenarios* setting with various dynamics  $T$  and reward function  $r$ . The results on the left (a) and (c) are for the in-distribution setting, while the results on the right (b) and (d) are for the out-of-distribution setting. We adopt the normalized return metric with five random seeds.

	(a) Train $\alpha = 5$ , Evaluate $\alpha = 5$				(b) Train $\alpha = 5$ , Evaluate $\alpha = 10$			
	MAMBA	DreamerV3	CaDM	Ours	MAMBA	DreamerV3	CaDM	Ours
Hopper	46.2±14.3	62.1±9.1	64.2±15.4	<b>67.2±13.1</b>	31.2±16.5	49.7±12.5	<b>60.2±11.2</b>	58.1±12.1
Walker	52.4±11.2	59.1±13.1	52.6±9.7	<b>62.3±10.2</b>	42.0±18.8	35.6±13.0	48.2±6.9	<b>59.0±11.0</b>
Cheetah	53.0±10.8	48.2±11.3	51.2±11.3	<b>56.8±9.2</b>	51.1±12.3	45.1±16.2	46.3±12.6	<b>49.9±15.6</b>
	(c) Train $\alpha = 10$ , Evaluate $\alpha = 10$				(d) Train $\alpha = 10$ , Evaluate $\alpha = 20$			
	MAMBA	DreamerV3	CaDM	Ours	MAMBA	DreamerV3	CaDM	Ours
Hopper	51.5±12.8	52.6±9.1	61.6±13.5	<b>62.0±11.6</b>	39.7±12.6	<b>49.8±15.2</b>	43.8±10.8	47.5±13.2
Walker	47.3±14.0	57.8±14.3	50.7±16.0	<b>61.3±8.7</b>	38.1±13.7	44.8±10.9	40.7±12.7	<b>49.7±15.2</b>
Cheetah	<b>61.2±11.6</b>	52.9±16.4	48.3±13.8	60.7±13.1	45.2±14.0	<b>50.1±12.9</b>	47.7±13.8	48.0±16.8

Table 2: Experimental results on the *multi-scenarios* setting with various dynamics function  $T$ . The results on the left (a) and (c) are for the in-distribution setting, while the results on the right (b) and (d) are for the out-of-distribution setting. We adopt the normalized return metric with five random seeds.

meta-RL through trajectory sampling strategies and an adaptive horizon scheduling mechanism. CaDM (Lee et al. 2020) utilizes contextual trajectory data to construct bidirectional dynamics models, enabling effective generalization across diverse dynamical systems.

## 5.2 Main Results

**Answer for question 1.** In this setting, we train and evaluate algorithms in multi-scenarios by merging data from Hopper, Walker and Cheetah. In addition, we divide the test scenarios into in-distribution and out-of-distribution. For the in-distribution setting, the distributions of the training scenario set match those of the test scenarios (e.g., Train ( $\alpha = 5, \beta = 20$ ), Evaluate ( $\alpha = 5, \beta = 20$ )). For the out-of-distribution setting, the test scenario’s distributions exceed those of the training set (e.g., Train ( $\alpha = 5, \beta = 20$ ), Evaluate ( $\alpha = 10, \beta = 50$ )).

By comparing the experimental results (a) and (c) in Table 1, we find that both MrCoM and CaDM are more adaptable to dynamic changes in meta-scenarios. MrCoM demonstrates the more significant performance advantage and generalization capability over other algorithms as the range of

meta-scenarios expands. In addition, the comparison between in-distribution and out-of-distribution experimental results shows that all algorithms have a performance decline when the test distribution exceeds the training distribution. We find MAMBA shows the most significant drop, which is attributed to the fact that the meta-scenarios do not conform to its assumption of decomposable scenario distributions. In contrast, our algorithm shows a smaller decline and achieves optimal results in 11 out of 12 cases.

**Answer for question 2.** We fix the training scenario objective by setting the target speed to  $50\% \cdot v_{\max}$ . Then, we change the dynamics function by randomly sampling parameters  $\alpha$ . Like the **Answer for question 1**, we train algorithms on the multi-scenarios and divide the experiments into in-distribution and out-of-distribution settings. The experimental results in Table 2 show out-of-distribution conditions result in a performance decline for all methods, but MrCoM shows a smaller decrease, indicating stronger adaptability to out-of-distribution scenarios. Out of 6 results, our method achieved the best performance in 5 cases, demonstrating that MrCoM provides optimal generalization performance concerning dynamics.

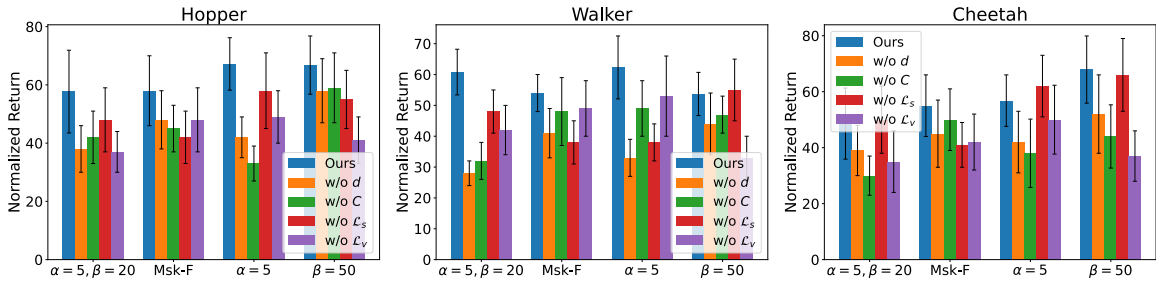


Figure 3: Module ablation studies on various scenarios with five random seeds.

**Answer for question 3.** In this setting, we evaluate algorithms under various observation functions  $\mathcal{O}^i$  by conducting the following modifications: *Add-G*: Adds Gaussian noise to each dimension of the original observation. *Add-D*: Introduces additional noise information, equivalent to 10% of the original dimensions, into the observation. *Msk-R*: Randomly masks 10% of the observation values in the original observation. *Msk-F*: Masks a fixed 10% of the observation values in the original observation across specific dimensions. The experimental results in Table 3 show that our algorithm achieved the best results in 8 out of 12 cases across various observation functions. Particularly in the *Add-G* setting, the MrCoM method demonstrates significant advantages.

### 5.3 Ablation Study

This experiment primarily examines the impact of various modules of our method on the final performance and validates their implementation’s effectiveness on the overall model. Our experiments are divided into the following groups: (1) *w/o d*: Without constructing the deterministic state  $d$ , using only the stochastic states  $u, h$ . (2) *w/o C*: Without using context and prompt scenario information during state prediction. (3) *w/o  $\mathcal{L}_s$* : Without the meta-state regularization loss. (4) *w/o  $\mathcal{L}_v$* : Without constructing the meta-value and using meta-value regularization during learning the dynamics model.

The experimental settings also incorporate the aforementioned different configurations to compare each component’s impact comprehensively. The experimental results in Figure 3 show that *w/o C* has the most significant impact in the multi-scenario setting. The absence of  $d$  considerably affects temporal information, thus significantly impacting dynamics changes. Overall, it can be seen that the complete MrCoM consistently achieves optimal performance across different settings.

## 6 Related Works

**Model-based RL** offers an effective methods to enhance sampling efficiency (Nagabandi et al. 2018). Dyna (Sutton 1991) uses data generated from model-based predictions for data augmentation in agent trajectories. Muzero (Schrittwieser et al. 2020), on the other hand, employs Monte Carlo Tree Search (Coulom 2006) based on its model, enabling

the identification of superior strategies with fewer interactions with the environment. However, previous model-based methods often focus on generalization concerning changes in dynamics (Lee et al. 2020) or adapting to diverse reward functions (Rimon et al. 2024; Kidambi et al. 2020), without addressing the issue of unified modeling across different scenarios. Additionally, there is an increasing focus on resolving the challenge of learning models that align to acquire an effective policy (Lambert et al. 2020). This can be primarily categorized into two approaches: value-aware (Abachi 2020; Wang et al. 2023) and policy-aware (Grimm et al. 2021; Voelcker et al. 2022; Voloshin, Jiang, and Yue 2021). However, these methods often make world models more task-specific, thereby losing the ability to generalize across scenarios.

**Meta RL** enables agents to learn from related tasks and then adapt to new tasks with minimal interaction data. Most meta-RL algorithms adhere to a model-free paradigm, subdivided into optimization-based and context-based methodologies. MAML (Finn, Abbeel, and Levine 2017) and PrOMP (Rothfuss et al. 2018), exemplifying optimization-based meta-RL, update policies based on gradients derived from training tasks. Context-based meta-RL, represented by RL2 (Duan et al. 2016), suggests conditioning policies on hidden states extracted from task trajectories. PEARL (Rakelly et al. 2019) distinguishes task inference from action selection, facilitating explicit task representations. Unlike that, in this paper, we hope to learn a world model that can be generalized across multiple scenarios.

## 7 Conclusion

In this paper, we aim to improve the generalization ability of the world-model across multi-scenarios. To address this issue, we propose MrCoM, a unified world-model capable of generalizing across different scenarios. Specifically, MrCoM adopts meta-state regularization to extract unified representation of scenario-relevant information, and meta-value regularization to align world-model optimization with policy learning across diverse scenario objectives. We theoretically analyze the upper bound of the world-model’s generalization error. We conduct extensive experiments using the MuJoCo-based physics engine. The experimental results show that the learned world-model achieves strong generalization ability across multi-scenarios.

## Acknowledgments

This work is supported by the National Key R&D Program of China (No.2022ZD0116405).

## References

- Abachi, R. 2020. *Policy-aware model learning for policy gradient methods*. University of Toronto (Canada).
- Akkaya, I.; Andrychowicz, M.; Chociej, M.; Litwin, M.; McGrew, B.; Petron, A.; Paino, A.; Plappert, M.; Powell, G.; Ribas, R.; et al. 2019. Solving rubik’s cube with a robot hand. *arXiv preprint arXiv:1910.07113*.
- Berner, C.; Brockman, G.; Chan, B.; Cheung, V.; Dbiak, P.; Dennison, C.; Farhi, D.; Fischer, Q.; Hashme, S.; Hesse, C.; et al. 2019. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*.
- Coulom, R. 2006. Efficient selectivity and backup operators in Monte-Carlo tree search. In *International conference on computers and games*, 72–83. Springer.
- Duan, Y.; Schulman, J.; Chen, X.; Bartlett, P. L.; Sutskever, I.; and Abbeel, P. 2016.  $RL^2$ : Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*.
- Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, 1126–1135. PMLR.
- Grimm, C.; Barreto, A.; Farquhar, G.; Silver, D.; and Singh, S. 2021. Proper value equivalence. *Advances in Neural Information Processing Systems*, 34: 7773–7786.
- Hafner, D.; Pasukonis, J.; Ba, J.; and Lillicrap, T. 2023. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*.
- Janner, M.; Fu, J.; Zhang, M.; and Levine, S. 2019. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32.
- Kidambi, R.; Rajeswaran, A.; Netrapalli, P.; and Joachims, T. 2020. Morel: Model-based offline reinforcement learning. *Advances in neural information processing systems*, 33: 21810–21823.
- Kiran, B. R.; Sobh, I.; Talpaert, V.; Mannion, P.; Al Salhab, A. A.; Yogamani, S.; and Pérez, P. 2021. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(6): 4909–4926.
- Lambert, N.; Amos, B.; Yadan, O.; and Calandra, R. 2020. Objective mismatch in model-based reinforcement learning. *arXiv preprint arXiv:2002.04523*.
- Lee, K.; Seo, Y.; Lee, S.; Lee, H.; and Shin, J. 2020. Context-aware dynamics model for generalization in model-based reinforcement learning. In *International Conference on Machine Learning*, 5757–5766. PMLR.
- Nagabandi, A.; Clavera, I.; Liu, S.; Fearing, R. S.; Abbeel, P.; Levine, S.; and Finn, C. 2018. Learning to adapt in dynamic, real-world environments through meta-reinforcement learning. *arXiv preprint arXiv:1803.11347*.
- Poole, B.; Ozair, S.; Van Den Oord, A.; Alemi, A.; and Tucker, G. 2019. On variational bounds of mutual information. In *International Conference on Machine Learning*, 5171–5180. PMLR.
- Rakelly, K.; Zhou, A.; Finn, C.; Levine, S.; and Quillen, D. 2019. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *International conference on machine learning*, 5331–5340. PMLR.
- Rimon, Z.; Jurgenson, T.; Krupnik, O.; Adler, G.; and Tamar, A. 2024. Mamba: an effective world model approach for meta-reinforcement learning. *arXiv preprint arXiv:2403.09859*.
- Rothfuss, J.; Lee, D.; Clavera, I.; Asfour, T.; and Abbeel, P. 2018. Promp: Proximal meta-policy search. *arXiv preprint arXiv:1810.06784*.
- Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; et al. 2020. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839): 604–609.
- Silver, D.; Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Lai, M.; Guez, A.; Lanctot, M.; Sifre, L.; Kumaran, D.; Graepel, T.; et al. 2018. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419): 1140–1144.
- Sun, Z.; Shen, S.; Cao, S.; Liu, H.; Li, C.; Shen, Y.; Gan, C.; Gui, L.-Y.; Wang, Y.-X.; Yang, Y.; et al. 2023. Aligning large multimodal models with factually augmented rlhf. *arXiv preprint arXiv:2309.14525*.
- Sutton, R. S. 1991. Dyna, an integrated architecture for learning, planning, and reacting. *ACM Sigart Bulletin*, 2(4): 160–163.
- Tassa, Y.; Doron, Y.; Muldal, A.; Erez, T.; Li, Y.; Casas, D. d. L.; Budden, D.; Abdolmaleki, A.; Merel, J.; Lefrancq, A.; et al. 2018. Deepmind control suite. *arXiv preprint arXiv:1801.00690*.
- Todorov, E.; Erez, T.; and Tassa, Y. 2012. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, 5026–5033. IEEE.
- Voelcker, C.; Liao, V.; Garg, A.; and Farahmand, A.-m. 2022. Value gradient weighted model-based reinforcement learning. *arXiv preprint arXiv:2204.01464*.
- Voloshin, C.; Jiang, N.; and Yue, Y. 2021. Minimax model learning. In *International Conference on Artificial Intelligence and Statistics*, 1612–1620. PMLR.
- Wang, X.; Wongkamjan, W.; Jia, R.; and Huang, F. 2023. Live in the moment: Learning dynamics model adapted to evolving policy. In *International Conference on Machine Learning*, 36470–36493. PMLR.