CAUSALDREAM: CAUSAL WORLD MODELS BY DECONFOUNDING OF DYNAMICS IN EPISODICALLY NON-STATIONARY ENVIRONMENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models trained on observational data can conflate the effects of actions with those of unobserved physical factors, such as gravity or mass, that stay fixed within an episode but vary across episodes. When the latent state fails to retain these factors, they confound the learned dynamics and undermine policy robustness. Simply conditioning the model on an inferred context neither ensures that the dynamics use it nor exploits it to generate experience beyond the current policy. We propose CausalDream, which treats these factors as exogenous variables of a structural causal model. Its Causal World Model conditions latent transitions on exogenous variables that a label-supervised estimator infers from a short probe trajectory. A Causal Consistency Regularizer reconstructs these estimates from latent trajectories so that the dynamics retain them. Counterfactual imagination then intervenes on the first imagined action while holding the estimates fixed, broadening the experience available for policy learning. Across eight DeepMind Control tasks, CausalDream achieves the highest aggregate learning AUC among all compared methods, 12.9% above an architecture-matched baseline that conditions on the same estimates without the causal components. It also predicts the effects of action interventions more accurately on all four evaluated tasks, showing that its gains go beyond context conditioning alone.

1 INTRODUCTION

Model-based reinforcement learning (MBRL) learns a world model of environment dynamics and trains policies on imagined trajectories, reducing costly real-world interaction (Sutton, 1991; Hafner et al., 2025). World models have achieved strong results in robotic control (Wu et al., 2023; Zhou et al., 2024; Li et al., 2025b; Huang et al., 2025) and open-world settings (Li et al., 2025a), with growing attention to generalization (Sun et al., 2024a; Prasanna et al., 2024) and sample efficiency (Ma et al., 2024).

Many control problems are *episodically non-stationary*: physical factors such as gravity or payload mass remain fixed within an episode but vary across episodes (Lee et al., 2020; Doshi-Velez & Konidaris, 2016; Luo et al., 2024). These factors affect state transitions directly and affect action selection indirectly through the interaction history. When the compressed latent state of a world model does not retain these physical factors, they act as unobserved confounders of the learned dynamics (Li et al., 2020). Because the policy acts on context-relevant history omitted from the model state, the learned action–outcome association mixes the effect of the action with that of the physical factor, as with hidden factors that influence both actions and outcomes (Lu et al., 2018). The model can then learn spurious correlations, yielding biased predictions and brittle policies under dynamics shifts (Pearl, 2009; Pearl et al., 2016).

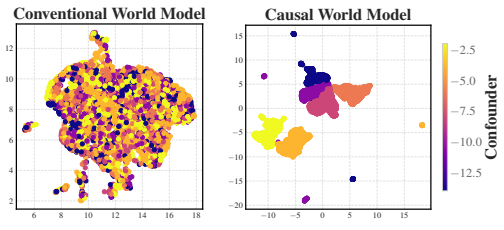


Figure 1: UMAP of latent states on *Walker Run*, colored by gravity (darker means lower). Latent states of DreamerV3 (left) are intermixed across gravity values, whereas those of our Causal World Model (right) are more clearly separated.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Figure 1 (left) illustrates this on *Walker Run*, where the latent states of DreamerV3 are intermixed across gravity values.

Contextual world models mitigate this problem by conditioning the dynamics on observed or inferred context (Lee et al., 2020; Prasanna et al., 2024; Röder et al., 2025). Conditioning alone, however, leaves two issues open. First, a powerful autoregressive sequence model can largely ignore a low-dimensional context input, so the latent dynamics need not retain it. Second, the policy still learns from trajectories imagined under its own actions, which concentrate on familiar behavior and can reinforce policy bias (Lu et al., 2020). Both issues call for treating the physical factors as causal variables that the model must retain and that can be held fixed while actions are varied.

We propose **CausalDream**, which formulates episodically non-stationary control as an episode-level Structural Causal Model (SCM) (Pearl, 2009; Pearl et al., 2016) with the physical factors as exogenous variables (Figure 2). A Twin Environment Mechanism (TEM) estimates these variables from a short probe trajectory collected before each episode, so the agent need not act blindly while the dynamics are unknown. The estimator uses simulator parameters only as privileged labels during pre-training and requires only observation–action trajectories at deployment, following the asymmetric paradigm of sim-to-real adaptation (Kumar et al., 2021). The Causal World Model (CWM) conditions its latent transitions on this estimate, and the Causal Consistency Regularizer (CCR) reconstructs the estimate from latent trajectories to encourage its retention. Counterfactual Imagination (CFI) then intervenes on the first imagined action while holding the initial model state and the estimate fixed, and follows the current policy thereafter, broadening the experience available for actor–critic learning. We use "counterfactual" in the model-based sense of prior counterfactual data augmentation (Lu et al., 2020; Sun et al., 2024b), and characterize conditions under which these rollouts agree with environment trajectories.

We evaluate CausalDream on nine control tasks from DeepMind Control and Gymnasium against conventional world models, contextual world models, and model-free methods. Across eight DeepMind Control tasks, CausalDream achieves the highest aggregate normalized learning AUC, 12.91% above ContextDream, an architecture-matched baseline that shares the estimator, conditioning pathway, and backbone but omits CCR and CFI. With ground-truth exogenous inputs given to both models, CausalDream also predicts the return effects of action interventions more accurately on all four evaluated tasks, and ablations show that removing TEM, CCR, or CFI each lowers performance. Our contributions are as follows:

- We cast world modeling under episodic dynamics shifts as an episode-level SCM, discussing when physical factors can confound a compressed latent model and specifying the intervention semantics for counterfactual imagination.
- We propose CausalDream, which combines TEM-based exogenous-variable estimation, a CWM regularized by CCR, and CFI for policy learning. We support it with analyses of support-level exogenous recovery, information retention under CCR, and rollout consistency on the covered domain.
- Through an architecture-matched baseline, paired action-intervention evaluations, and component ablations, we show that CausalDream's gains in sample efficiency, generalization, and action-effect prediction are not explained by conditioning on the estimated context alone.

2 Related Work

Conventional World Models. World models learn environment dynamics to support planning and policy learning through imagination (Ha & Schmidhuber, 2018; LeCun, 2022). Conventional models such as the Dreamer family (Hafner et al., 2020; 2021; 2025) do not explicitly distinguish episode-level dynamics factors from the evolving state representation, which can limit adaptation when these factors vary across episodes. To address this limitation, we construct a causal world model that conditions latent transitions on estimated exogenous factors and encourages their retention, supporting adaptation to dynamics shifts.

Causal World Models and Counterfactual RL. Causal reinforcement learning emphasizes modeling causal mechanisms rather than relying solely on statistical correlations, which is important for robustness and generalization under distribution shifts (Deng et al., 2023; He et al., 2022; Richens & Everitt, 2024). Prior causal world model studies show that conventional world models can fail when

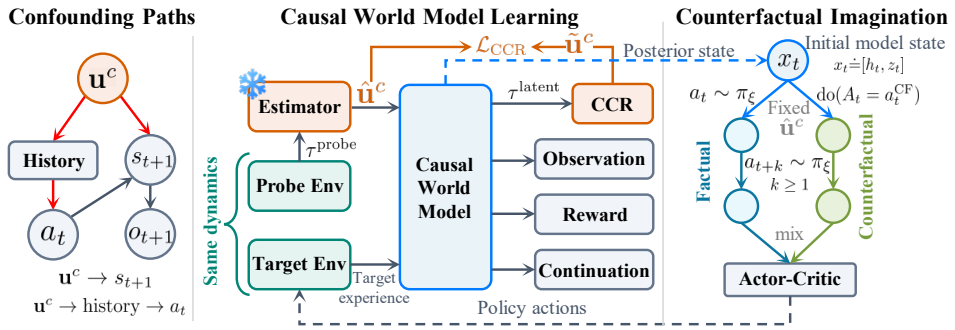


Figure 2: Overview of CausalDream. (*Left*) Confounding paths. (*Middle*) Exogenous-variable estimation via TEM and Causal World Model learning with the Causal Consistency Regularizer. (*Right*) Actor–Critic training with factual and counterfactual imagination under fixed $\hat{\mathbf{u}}^c$.

latent factors confound transition dynamics, motivating the need to explicitly account for exogenous dynamics factors (Li et al., 2020). Our work follows this line by instantiating a causal world model for episodically non-stationary control, where exogenous factors are estimated and incorporated into latent dynamics. In addition, prior counterfactual RL methods have used SCM-based reasoning for policy evaluation and counterfactual experience generation (Buesing et al., 2019; Lu et al., 2020; Pitis et al., 2020; Sun et al., 2024b). CausalDream complements these works by combining causal world modeling with counterfactual inference for imagined policy learning.

Contextual RL. Contextual RL typically models tasks as Contextual MDPs (Hallak et al., 2015) or Hidden Parameter MDPs (Doshi-Velez & Konidaris, 2016) to address variations in environmental dynamics, where the agent utilizes context variables to adapt its policy. Existing methods can be broadly categorized into two types: those utilizing explicitly observed privileged information as context (Prasanna et al., 2024), and those inferring the latent context from historical interactions (Lee et al., 2020; Röder et al., 2025). The episodically non-stationary environments considered in this paper can also be represented as Contextual MDPs or Hidden-Parameter MDPs by treating u^c as an episode-level context. We do not claim that the SCM formulation defines a strictly more general environment class. The SCM formulation specifies the intervention semantics used by CausalDream. Alternative-action rollouts retain the same initial model state and inferred episode-level dynamics factor. CausalDream combines this explicit conditioning with CCR and CFI, which performs action-intervention data augmentation for policy learning. More discussions of related works are in Appendix B.

3 Preliminaries

POMDP. We consider a partially observable Markov decision process with environment state $s_t \in \mathcal{S}$, observation $o_t \in \mathcal{O}$, action $a_t \in \mathcal{A}$, and reward $r_t = R(s_t, a_t)$. The agent observes o_t rather than s_t and learns a policy maximizing $\mathbb{E}_\pi[\sum_{t=1}^{T} \gamma^{t-1} r_t]$, where $\gamma \in [0, 1)$ (Sutton & Barto, 1998). We use $j_t \in \{0, 1\}$ to indicate episode continuation.

SCM. For a rigorous treatment of causality, we follow the framework of Pearl et al. (2016) and define the SCM using the tuple $\langle \mathcal{U}, \mathcal{V}, \mathcal{F} \rangle$, where $\mathcal{U}$ is a set of exogenous variables and $\mathcal{V}$ is a set of endogenous variables. In addition to describing statistical correlations, the SCM also characterizes causal mechanisms through structural equations expressed as $v_i = f_i(pa_i, u_i)$, for $i = 1, \ldots, n$, where $v_i \in \mathcal{V}, u_i \in \mathcal{U}, f_i \in \mathcal{F}$, and pa_i represents the parents of v_i. This formulation facilitates the distinction between spurious correlations and underlying causal mechanisms, thereby enabling counterfactual inference.

Causal POMDP. To investigate causality within the RL framework, similar to Li et al. (2020), we reformulate POMDP by SCM. In this formulation, the observation, state, action, and reward at each time step are categorized as endogenous variables, whereas environmental confounders are modeled as exogenous variables. For notational clarity, we represent the set of all confounders in a single episode as a vector $\mathbf{u}^c = (u_1^c, u_2^c, \ldots, u_N^c)^\top \in \mathbb{R}^N$, where N is the number of confounders. In this work, we focus on episode-level exogenous dynamics factors, where u_i^c remains fixed within an

episode but varies across episodes (Hallak et al., 2015; Doshi-Velez & Konidaris, 2016; Röder et al., 2025). This assumption captures a broad class of practical scenarios, such as payload mass, robot morphology, actuator configuration, or terrain condition, which typically remain stable during a short control episode but differ across tasks. For example, a legged delivery robot may carry packages of different unknown masses. The payload mass remains fixed during each delivery episode but varies across episodes. Accordingly, the transition dynamics is characterized as a deterministic function conditioned on these exogenous factors:

$$s_{t+1} = f(s_t, a_t, \mathbf{u}^c) \tag{1}$$

where f represents the causal mechanism of environments. The exogenous factors $\mathbf{u}^c$ influence subsequent states directly and action selection indirectly through the observation–action history, as illustrated in Figure 2 (left). If the behavior policy depends only on this history and independent policy randomness, conditioning on the complete history removes the resulting action–context dependence. A compressed model state, however, may omit dynamics information contained in that history. Residual confounding can arise when the behavior policy uses context-relevant information that the model state does not retain. This motivates explicitly conditioning the transition model on estimated exogenous factors and encouraging their retention in latent trajectories. Appendix F provides formal assumptions and theoretical analysis.

Episode-Level Counterfactual Inference. In this work, we use episode-level counterfactual inference in a model-based sense, following the convention of prior counterfactual data augmentation methods (Lu et al., 2020; Pitis et al., 2022; Sun et al., 2024b). Our episode-level counterfactual query asks how the learned latent trajectory would evolve under an alternative action, starting from the same model state and holding the inferred episode-level dynamics factor $\hat{\mathbf{u}}^c$ fixed. This corresponds to a fixed-exogenous action intervention under the learned structural model. Writing x_t for the latent model state and P_ϕ for the learned transition kernel, the imagined transition is $P_\phi(x_{t+1}^{\text{CF}} \mid x_t, \text{do}(A_t = a_t^{\text{CF}}), \hat{\mathbf{u}}^c)$, where x_{t+1}^{CF} denotes the successor model state under the alternative action $a_t^{\text{CF}} \in \mathcal{A}$, with the initial model state x_t and estimated exogenous factor $\hat{\mathbf{u}}^c$ held fixed. The operator $\text{do}(\cdot)$ specifies the action intervention.

DreamerV3 Architecture. DreamerV3 uses a Recurrent State-Space Model (RSSM) (Hafner et al., 2020) with deterministic state h_t and stochastic state z_t. The recurrent update is $h_t = f_\phi(h_{t-1}, z_{t-1}, a_{t-1})$. The posterior $q_\phi(z_t \mid h_t, o_t)$ incorporates observations, while the prior $p_\phi(z_t \mid h_t)$ supports imagination without observations. Prediction heads reconstruct observations and predict rewards and episode continuation from model state $x_t \doteq [h_t, z_t]$.

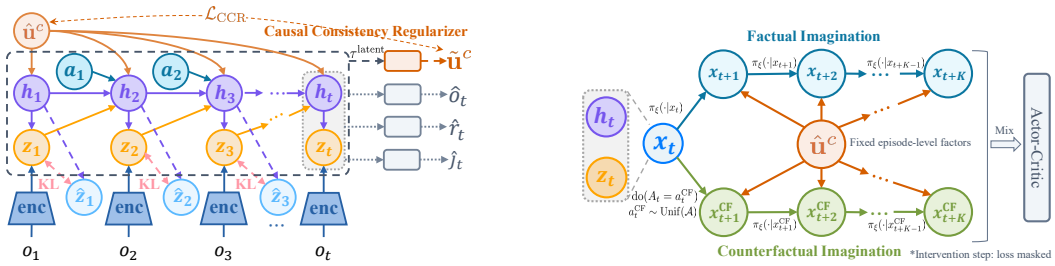
4 METHOD

CausalDream builds on DreamerV3 (Hafner et al., 2025). Figure 2 summarizes the framework, and Appendix A.1 provides the pseudocode.

4.1 ESTIMATION OF EXOGENOUS VARIABLES

Estimator Since the values of confounders are not available to the agent during policy learning but are required to build a causal world model, we first construct an estimator to approximate these exogenous variables. We estimate the physical exogenous variables rather than an unconstrained latent context. Simulator supervision anchors the representation to the factors that index the structural transition family. Holding $\hat{\mathbf{u}}^c$ fixed during counterfactual imagination therefore explicitly corresponds to holding the estimated episode-level physical condition fixed. To capture the latent factors from historical interactions, we employ an estimator $\mathcal{E}$ parameterized by θ. Following prior works (Nagabandi et al., 2019; Lee et al., 2020; Chen et al., 2024), the estimator consists of a visual encoder and a Gated Recurrent Unit (GRU), which takes the sequence of observation-action pairs $\tau = \{(o_t, a_t)\}_{t=1}^{L}$ as input to estimate the exogenous vector $\hat{\mathbf{u}}^c = \mathcal{E}_\theta(\tau)$.

We pre-train $\mathcal{E}_\theta$ on trajectories collected by a pre-defined policy, using simulator-provided physical parameters as privileged labels. For each factor, we minimize cross-entropy over its discrete support using Gaussian soft targets centered at the ground-truth value to reflect distances between parameter values (Mandal et al., 2024; de Vries & Thierens, 2025). The support value with the highest predicted probability gives the corresponding component of $\hat{\mathbf{u}}^c$. At deployment, the estimator requires only



(a) Causal World Model Learning

(b) Factual and counterfactual imagination

Figure 3: The overall architecture of the CWM and the policy learning in CausalDream. (a) The model explicitly incorporates $\hat{\mathbf{u}}^c$ into the recurrent dynamics. CCR reconstructs the episode-level exogenous estimate from the latent trajectory to encourage information retention. (b) While factual rollouts follow π_ξ, counterfactual trajectories are generated by applying $\mathrm{do}(A_t = a_t^{\mathrm{CF}})$ and subsequently following π_ξ, conditioned on fixed $\hat{\mathbf{u}}^c$. The model state $x_t \doteq [h_t, z_t]$.

observation–action trajectories, without ground-truth parameter inputs. The target construction and training details are provided in Appendix A.2.

Twin Environment Mechanism However, directly applying the estimator may introduce the cold-start problem in non-stationary environments because accurate estimation requires a trajectory of length L. Acting without prior knowledge forces the agent to interact blindly with the unknown dynamics during the initial phase. This initialization lag inevitably leads to suboptimal policy execution and poor early-stage performance, as the world model struggles to adapt to unobserved confounders in real-time. To address this, we introduce a Twin Environment Mechanism (TEM), which separates the identification of exogenous variables from the execution of the actual task. As shown in Figure 2 (middle), for each episode, we instantiate a probe environment $\mathrm{env}^{\mathrm{probe}}$ that shares identical parameters and mechanisms with the target environment $\mathrm{env}^{\mathrm{target}}$. Before the agent attempts the actual task in $\mathrm{env}^{\mathrm{target}}$, we use a pre-defined agent to execute a short exploration trajectory in $\mathrm{env}^{\mathrm{probe}}$. The estimator utilizes this trajectory to infer the exogenous variable. This inferred variable is then used to condition the world model and policy in $\mathrm{env}^{\mathrm{target}}$, allowing the agent to bypass the blind exploration phase and adapt to the specific dynamics immediately from the first time step. In simulation, TEM separates system identification from task execution through a parallel probe environment. A practical analogue for physical deployment is a short pre-task calibration phase on the same platform, provided that the exogenous factors remain unchanged between calibration and control.

4.2 Causal World Model Learning

Building upon the RSSM, we propose a Causal World Model, as shown in Figure 3a. Unlike conventional world models that rely solely on historical observations and actions, the CWM explicitly incorporates the estimated exogenous variable $\hat{\mathbf{u}}^c$ into the transition dynamics to achieve deconfounding of dynamics. To align with the structural equation in Equation 1, we reformulate the deterministic sequence model as follows:

$$h_t = f_\phi(h_{t-1}, z_{t-1}, a_{t-1}, \hat{\mathbf{u}}^c) \tag{2}$$

By conditioning on $\hat{\mathbf{u}}^c$, the CWM is structurally capable of distinguishing between different environmental dynamics.

Nevertheless, given the powerful autoregressive nature of the sequence model, simply conditioning on low-dimensional $\hat{\mathbf{u}}^c$ does not ensure the effective utilization of such causal information. This neglect may lead to causal inconsistency, where the learned dynamics diverge from the actual environment confounders. To mitigate this information loss, we introduce a Causal Consistency Regularizer (CCR) that encourages latent trajectories to retain the estimated exogenous factors. For the latent trajectory $\tau^{\mathrm{latent}} = \{(h_t, z_t, a_t)\}_{t=1}^{T}$, a recurrent decoder $g_\phi(\tau^{\mathrm{latent}}) \in \mathbb{R}^N$ reconstructs the estimated exogenous variable in symlog coordinates. We optimize CCR using

$$\mathcal{L}_{\mathrm{CCR}} = \sum_{k=1}^{N} \left[g_{\phi,k}(\tau^{\mathrm{latent}}) - \mathrm{symlog}(\hat{u}_k^c) \right]^2, \tag{3}$$

where $\hat{\mathbf{u}}^c$ is estimated by $\mathcal{E}_\theta$, and $\mathrm{symlog}(x) = \mathrm{sign}(x)\log(1+|x|)$ is applied componentwise. We apply symlog to compress target magnitudes while preserving the signs and information of the exogenous factors. For finite-support exogenous estimates, sufficiently small reconstruction error implies a lower bound on the information retained in the latent trajectory (Proposition 10). The CCR serves as a soft information-theoretic constraint, encouraging the latent dynamics to preserve information regarding the exogenous variable. CCR is optimized jointly with the dynamics and prediction objectives to combine exogenous-information retention with predictive learning.

Similar to DreamerV3, the parameters ϕ of the CWM are optimized by minimizing the joint loss function comprising the prediction loss $\mathcal{L}_{\text{pred}}$, the dynamics loss $\mathcal{L}_{\text{dyn}}$, the representation loss $\mathcal{L}_{\text{rep}}$ and the CCR loss $\mathcal{L}_{\text{CCR}}$:

$$\mathcal{L}_{\text{joint}} = \mathbb{E}_{q_\phi}\left[\frac{1}{T}\sum_{t=1}^{T}\big(\mathcal{L}_{\text{pred},t} + \mathcal{L}_{\text{dyn},t} + \mathcal{L}_{\text{rep},t}\big) + \mathcal{L}_{\text{CCR}}\right], \tag{4}$$

where $\mathcal{L}_{\text{pred}}$, $\mathcal{L}_{\text{dyn}}$, and $\mathcal{L}_{\text{rep}}$ follow the same definitions as in DreamerV3 (see details in Appendix A.3).

4.3 Counterfactual Imagination

Our approach largely adheres to the actor-critic framework as established in DreamerV3:

$$\text{Actor}: \quad \pi_\xi(a_t|h_t, z_t) \qquad \text{Critic}: \quad v_\psi(h_t, z_t) \tag{5}$$

Conventional world models train the agent using factual trajectories imagined by the current policy π_ξ, as shown in Figure 3b. However, this creates a feedback loop where the policy only visits familiar states and learns from this limited distribution, leading to policy bias and poor adaptability in non-stationary environments (Lu et al., 2020).

To diversify policy training, we introduce Counterfactual Imagination (CFI), which augments policy-generated imagined trajectories with alternative-action rollouts from the learned CWM, as shown in Figure 3b. Explicit exogenous conditioning allows the factual and alternative-action rollouts to share the same estimated episode-level factor.

Following the Abduction–Action–Prediction procedure of Pearl et al. (2016), adapted to our episode-level setting in Section 3, CFI uses the probe estimate $\hat{\mathbf{u}}^c = \mathcal{E}_\theta(\tau^{\text{probe}})$, intervenes on the initial imagined action, and predicts the resulting trajectory. From the same initial model state, we sample a_t^{CF} uniformly from the action space $\mathcal{A}$ (Lu et al., 2020; Sun et al., 2024b) while keeping $\hat{\mathbf{u}}^c$ and the learned transition mechanism unchanged:

$$h_{t+1}^{\text{CF}} = f_\phi(h_t, z_t, a_t^{\text{CF}}, \hat{\mathbf{u}}^c) \tag{6}$$

We then sample $z_{t+1}^{\text{CF}} \sim p_\phi(\cdot \mid h_{t+1}^{\text{CF}})$ to obtain $x_{t+1}^{\text{CF}} = [h_{t+1}^{\text{CF}}, z_{t+1}^{\text{CF}}]$, from which the current policy generates the remaining imagined trajectory. These interventions broaden the imagined experience available for policy learning. We mix factual and counterfactual trajectories in equal proportions. For both actor and critic updates, the intervened first step is masked, and only the policy-generated continuation contributes; factual trajectories contribute from their first step. The masked objectives are given in Appendix A.4. Theorem 11 establishes that these action-intervention rollouts agree with the abstract environment trajectories when the exogenous estimate and initial model state are correct, the model transition matches the abstract dynamics, and the rollouts remain within the covered domain.

5 Experiments

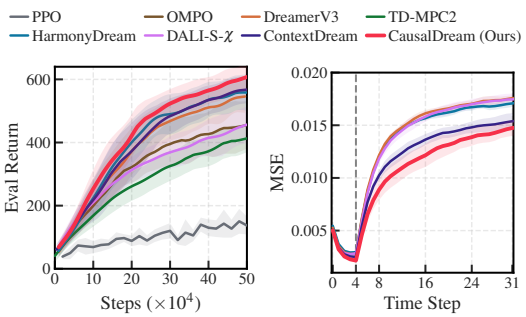
Environments. We evaluate our method on nine control tasks from the DeepMind Control Suite (Tassa et al., 2018) and Gymnasium (Towers et al., 2024). Following the episode-level formulation in Section 3, we vary physical factors, including *gravity, torso mass, wind, and actuator knee*, across episodes while keeping them fixed within each episode. All methods are trained and evaluated under the same task-specific physical-parameter distributions. We sample these factors from discrete uniform distributions and also consider combinations of factors. The discretization is relatively dense, with small parameter increments over broad ranges, providing a practical approximation to continuous dynamics variations. Detailed setups are provided in Appendix C.

Baselines. We compare CausalDream against a comprehensive set of representative strong baselines. The selection criteria are designed to verify our contributions from three distinct perspectives: (1) Conventional world models: we compare against DreamerV3 (Hafner et al., 2025) and TD-MPC2 (Hansen et al., 2024) to evaluate the benefits of our causal modeling and counterfactual imagination relative to conventional model-based approaches. (2) Adaptive or contextual world models: we include HarmonyDream (Ma et al., 2024) and DALI-S-χ (Röder et al., 2025) to assess our approach relative to methods based on adaptive learning or context inference. We construct ContextDream as an architecture-matched control that uses the same estimator, context-conditioning pathway, and world-model and policy backbone as CausalDream, but omits CCR and CFI. This comparison isolates the combined contribution of CCR and CFI within the same contextual architecture. (3) Model-free RL: we include OMPO (Luo et al., 2024), which addresses dynamics shifts, and PPO (Schulman et al., 2017) to compare sample efficiency with model-free approaches.

Implementation Details. To estimate exogenous variables, the probe agent collects a short trajectory of length $L = 8$ within TEM. All probe-environment interactions during policy learning are included in the reported interaction budget. Following the Abduction–Action–Prediction procedure, we fix the estimated exogenous variable $\hat{\mathbf{u}}^c$ and execute the intervention $\mathrm{do}(A_1 = a_1^{\mathrm{CF}})$ solely at the first step of the imagined rollout, subsequently following the current policy π_ξ for the remaining steps.

5.1 Main Results

Overall Performance. As shown in Figure 4a, CausalDream achieves the highest average return across the eight DMC tasks, with larger gains over the five world-model baselines on more challenging tasks. On basic tasks such as *Stand* and *Walk*, CausalDream achieves a 17.69% gain over the mean return of the five world-model baselines, while the gain increases to 30.06% on more demanding *Run* and *Hop* tasks (Appendix E.3). CausalDream outperforms DreamerV3 across all evaluated torso-mass and gravity settings (Appendix E.4), including the heaviest torso mass and strongest gravity. The gains also extend to compound dynamics shifts involving multiple concurrent physical factors (Appendix E.5). CausalDream achieves higher returns than all baselines on both evaluated settings, including joint gravity–wind variations in *Cheetah Run* and torso-mass–actuator variations in *Hopper Hop*.



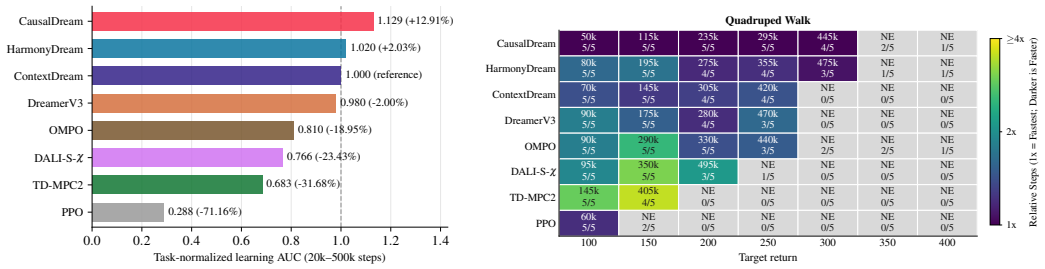
(a) Cross-task average (b) Prediction error

Figure 4: **Main results.** (a) Cross-task evaluation return averaged across all 8 DMC tasks; (b) Imagination prediction error on *Walker Run*.

Across these evaluations, CausalDream shows a consistent overall advantage over ContextDream, which uses the same estimator and estimated exogenous variables. This controlled comparison indicates that the gains cannot be explained by explicit exogenous-variable conditioning alone, but instead support the effectiveness of explicitly modeling confounded dynamics and leveraging counterfactual imagination.

Aggregate Sample Efficiency. We assess sample efficiency in aggregate learning performance and in target attainment. For target attainment, we report the median confirmation time across five seeds. NE indicates that fewer than three seeds confirm the target within the evaluation budget. As shown in Figure 5a, CausalDream achieves the highest aggregate normalized AUC of 1.129, with a 12.91% gain over ContextDream. The improvement is also consistent across individual tasks, ranging from 4.60% on *Cheetah Run* to 25.90% on *Hopper Hop* (Appendix E.2). Figure 5b shows that CausalDream requires the fewest interactions to reach every attainable return threshold on *Quadruped Walk*, and achieves both the highest attainment ratio and fastest confirmation at the demanding threshold of 300. The consistent advantage over ContextDream further indicates that this improvement cannot be explained by exogenous-variable conditioning alone.

Long-term Video Prediction. To ensure a controlled comparison, all methods are evaluated on the same trajectories collected by a third-party policy across the full gravity range, exposing them to



(a) Aggregate sample efficiency across 8 tasks.

(b) Threshold attainment on *Quadruped Walk*.

Figure 5: Sample efficiency analysis. (a) Aggregate normalized learning AUC across 8 DMC tasks. (b) Median target-confirmation steps across five seeds on *Quadruped Walk*, with confirmed seeds out of five shown below. NE indicates that fewer than three seeds confirm the target within the evaluation budget.

identical off-distribution states. As shown in Appendix E.6, CausalDream predicts future observations on *Walker Run* more accurately, while competing world models exhibit increasingly visible prediction errors as the rollout progresses. The quantitative results in Figure 4b show the same trend, with CausalDream achieving lower step-wise prediction error. Together, these results demonstrate that CausalDream maintains more accurate predictions over extended imagination horizons and is less susceptible to error accumulation.

Generalization. We evaluate OOD generalization of the learned CWM and policy, supplying CausalDream and ContextDream with ground-truth exogenous parameters to exclude estimation errors. In Table 1, CausalDream achieves the highest mean returns on *Walker Run* and *Cartpole Swingup*, with gains of 17.44% and 4.64% over ContextDream, respectively. With context access matched, these gains support the benefit of the combined CCR and CFI design beyond exogenous-variable conditioning alone (additional OOD results in Appendix E.8). We separately simulate estimator errors by perturbing the exogenous input within the training range. CausalDream maintains higher returns than ContextDream in most perturbed settings and shows smaller return losses on *Cartpole Swingup* under large errors ($|\epsilon| = 2$), indicating that its control benefits extend to imperfect exogenous inputs (Appendix E.9).

Table 1: OOD episode returns (higher is better). Bold/underline denote best/second-best.

Method	Walker Run	Cartpole Swingup
DreamerV3	206.19 $\pm$28.08	710.79 $\pm$130.50
TD-MPC2	211.50 $\pm$16.20	275.70 $\pm$51.76
HarmonyDream	121.90 $\pm$18.50	728.39 $\pm$61.21
DALI-S-χ	153.70 $\pm$20.50	<u>784.02</u> $\pm$45.69
OMPO	231.20 $\pm$48.80	273.14 $\pm$72.06
ContextDream	<u>244.90</u> $\pm$41.65	760.46 $\pm$103.34
CausalDream	**287.62** $\pm$35.59	**795.76** $\pm$56.24

5.2 Ablation Studies

Figure 6 examines the contributions of TEM, CCR, and CFI on *Walker Walk* and *LunarLander*. We remove TEM or CCR, replace the RNN-based CCR decoder with an MLP, and compare the default 50/50 factual–counterfactual mixture with factual-only and counterfactual-only policy training. Removing TEM or CCR reduces final mean returns on both tasks, supporting the benefits of episode-specific dynamics estimation and exogenous-information retention. The RNN-based CCR achieves higher final mean returns than the MLP alternative, while the balanced trajectory mixture outperforms either factual-only or counterfactual-only training. Additional experiments on *Walker Run* show that intervening only at the first imagined step performs best among the tested intervention counts. Additional results are provided in Section 5.3 and Appendix D.

5.3 Counterfactual Action-Effect Evaluation

The preceding experiments demonstrate the benefits of CausalDream in aggregate control performance and sample efficiency, with additional gains under dynamics shifts. To test the causal gains identified above more directly, we evaluate whether the learned model predicts how outcomes change under action interventions.

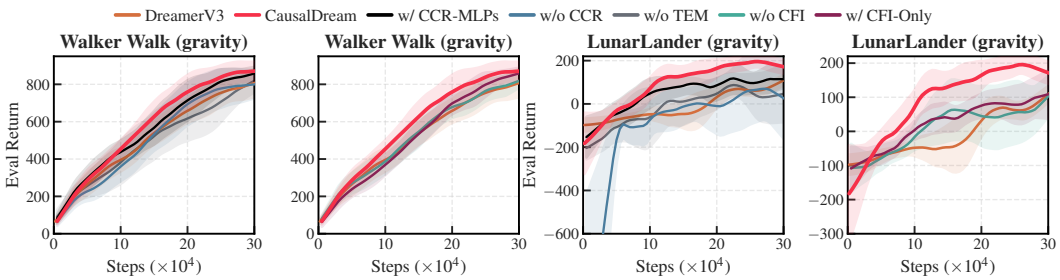


Figure 6: Ablations on *Walker Walk* (left pair) and *LunarLander* (right pair). Within each pair, the left panel evaluates TEM and CCR, and the right panel evaluates CFI.

Table 2: In-distribution (ID) return-effect MSE and candidate-selection regret ($H = 10$), including component ablations. Mean$_{\pm \mathrm{SD}}$ across seeds; bold/underline mark the best/second-best means across all five methods in each column. All values in 10^{-3}; lower is better.

	Hopper Hop		Walker Run		Cheetah Run		Cartpole Swingup	
Method	Return ↓	Regret ↓	Return ↓	Regret ↓	Return ↓	Regret ↓	Return ↓	Regret ↓
DreamerV3	$\underline{572.2}_{\pm 158}$	$286.6_{\pm 125}$	$342.4_{\pm 84}$	$340.4_{\pm 34}$	$81.5_{\pm 25}$	$93.5_{\pm 21}$	$46.9_{\pm 45}$	$30.8_{\pm 13}$
ContextDream	$683.7_{\pm 207}$	$283.6_{\pm 80}$	$\underline{275.0}_{\pm 53}$	$\underline{296.0}_{\pm 8.1}$	$82.5_{\pm 35}$	$\mathbf{87.5}_{\pm 31}$	$76.7_{\pm 72}$	$\underline{28.8}_{\pm 13}$
w/o CCR	$605.1_{\pm 130}$	$277.2_{\pm 45}$	$300.2_{\pm 75}$	$332.1_{\pm 57}$	$93.8_{\pm 30}$	$101.1_{\pm 30}$	$155.8_{\pm 270}$	$44.1_{\pm 44}$
w/o CFI	$611.4_{\pm 110}$	$\underline{273.0}_{\pm 92}$	$292.0_{\pm 59}$	$337.3_{\pm 47}$	$\mathbf{78.9}_{\pm 18}$	$\underline{93.0}_{\pm 27}$	$\underline{40.0}_{\pm 54}$	$\mathbf{27.8}_{\pm 24}$
CausalDream	$\mathbf{463.1}_{\pm 78}$	$\mathbf{220.0}_{\pm 107}$	$\mathbf{254.2}_{\pm 47}$	$\mathbf{291.3}_{\pm 21}$	$\underline{79.9}_{\pm 12}$	$98.9_{\pm 25}$	$\mathbf{30.2}_{\pm 8.9}$	$29.2_{\pm 7.8}$

From the same simulator state and episode condition, we change only the first action and replay the same subsequent actions across branches, then compare the predicted changes in return, reward, and observation between branches with simulator outcomes. Because these metrics score between-branch differences rather than per-branch predictions, they measure the modeled effect of $\mathrm{do}(A_1 = a_1^{\mathrm{CF}})$. CausalDream, its ablations, and ContextDream receive the same ground-truth exogenous parameters, so differences among them cannot arise from context information or estimation error. We also measure candidate-action selection through regret (protocol in Appendix E.10).

In Table 2, ContextDream is not more accurate than DreamerV3 on three of four tasks. These results show that context conditioning alone does not ensure accurate action effects. Compared with DreamerV3 and ContextDream, CausalDream achieves the lowest return-effect MSE on all four tasks, reducing it relative to ContextDream by 3.2% (*Cheetah Run*) to 60.6% (*Cartpole Swingup*). Removing CCR increases this error on all four tasks, and removing CFI on three (Table 2), with CCR contributing most consistently. Gains in candidate selection are task-dependent. CausalDream attains the lowest regret on *Hopper Hop* and *Walker Run*. On *Cheetah Run* and *Cartpole Swingup*, ContextDream and w/o CFI achieve lower regret. With context access matched, these results indicate that CausalDream's advantage over ContextDream is accompanied by more accurate modeling of action effects, rather than explained by access to the physical condition.

6 Conclusion and Future Work

In this paper, we propose CausalDream, a causal world model framework for robust policy learning in episodically non-stationary environments. The framework combines exogenous-variable estimation through the Twin Environment Mechanism (TEM), the Causal Consistency Regularizer (CCR), and Counterfactual Imagination (CFI) for policy learning. Across eight DMC tasks, CausalDream improves aggregate normalized learning AUC by 12.91% over the architecture-matched ContextDream baseline. Together with component ablations and more accurate action-effect prediction on four tasks, these results indicate that retaining exogenous information and exploiting it through action interventions, rather than conditioning alone, underlies the gains. Several directions remain for future work. First, extending the framework to continuously varying non-stationarity would broaden applicability beyond episode-constant dynamics. Second, incorporating continuous or flexible distributional representations could improve estimation fidelity over discrete approximations.

AI USE STATEMENT

We used LLMs to assist with manuscript polishing, mathematical exposition, and interpretation of experimental results. The authors take responsibility for the final content.

REFERENCES

Lars Buesing, Theophane Weber, Yori Zwols, Nicolas Heess, Sebastien Racaniere, Arthur Guez, and Jean-Baptiste Lespiau. Woulda, coulda, shoulda: Counterfactually-guided policy search. In *International Conference on Learning Representations*, 2019.

Ruifeng Chen, Xiong-Hui Chen, Yihao Sun, Siyuan Xiao, Minhui Li, and Yang Yu. Policy-conditioned environment models are more generalizable. In *Forty-first International Conference on Machine Learning*, 2024.

Sjoerd de Vries and Dirk Thierens. Learning with confidence: training better classifiers from soft labels. *Machine Learning*, 114(11):238, 2025.

Zhihong Deng, Jing Jiang, Guodong Long, and Chengqi Zhang. Causal reinforcement learning: A survey. *arXiv preprint arXiv:2307.01452*, 2023.

Finale Doshi-Velez and George Konidaris. Hidden parameter Markov decision processes: A semi-parametric regression approach for discovering latent task parametrizations. In *Proceedings of the International Joint Conference on Artificial Intelligence*, volume 2016, pp. 1432, 2016.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. *Advances in Neural Information Processing Systems*, 31, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *International Conference on Learning Representations*, 2020.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering Atari with discrete world models. *International Conference on Learning Representations*, 2021.

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

Assaf Hallak, Dotan Di Castro, and Shie Mannor. Contextual Markov decision processes. *arXiv preprint arXiv:1502.02259*, 2015.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *International Conference on Machine Learning*, 2022.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. *International Conference on Learning Representations*, 2024.

Tom He, Jasmina Gajcin, and Ivana Dusparic. Causal counterfactuals for improving the robustness of reinforcement learning. *arXiv preprint arXiv:2211.05551*, 2022.

Yuhang Huang, Jiazhao Zhang, Shilong Zou, Xinwang Liu, Ruizhen Hu, and Kai Xu. LaDi-WM: A latent diffusion-based world model for predictive manipulation. *arXiv preprint arXiv:2505.11528*, 2025.

Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. RMA: Rapid motor adaptation for legged robots. In *Robotics: Science and Systems*, 2021.

Yann LeCun. A path towards autonomous machine intelligence. *OpenReview*, 2022.

Kimin Lee, Younggyo Seo, Seunghyun Lee, Honglak Lee, and Jinwoo Shin. Context-aware dynamics model for generalization in model-based reinforcement learning. In *International Conference on Machine Learning*, pp. 5757–5766. PMLR, 2020.

Jiajian Li, Qi Wang, Yunbo Wang, Xin Jin, Yang Li, Wenjun Zeng, and Xiaokang Yang. Open-world reinforcement learning over long short-term imagination. *The Thirteenth International Conference on Learning Representations*, 2025a.

Minne Li, Mengyue Yang, Furui Liu, Xu Chen, Zhitang Chen, and Jun Wang. Causal world models by unsupervised deconfounding of physical dynamics. *arXiv preprint arXiv:2012.14228*, 2020.

Wenxuan Li, Hang Zhao, Zhiyuan Yu, Yu Du, Qin Zou, Ruizhen Hu, and Kai Xu. PIN-WM: Learning physics-informed world models for non-prehensile manipulation. *arXiv preprint arXiv:2504.16693*, 2025b.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, pp. 4114–4124. PMLR, 2019.

Chaochao Lu, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Deconfounding reinforcement learning in observational settings. *arXiv preprint arXiv:1812.10576*, 2018.

Chaochao Lu, Biwei Huang, Ke Wang, José Miguel Hernández-Lobato, Kun Zhang, and Bernhard Schölkopf. Sample-efficient reinforcement learning via counterfactual-based data augmentation. *arXiv preprint arXiv:2012.09092*, 2020.

Yu Luo, Tianying Ji, Fuchun Sun, Jianwei Zhang, Huazhe Xu, and Xianyuan Zhan. OMPO: A unified framework for RL under policy and dynamics shifts. *International Conference on Machine Learning*, 2024.

Haoyu Ma, Jialong Wu, Ningya Feng, Chenjun Xiao, Dong Li, Jianye Hao, Jianmin Wang, and Mingsheng Long. HarmonyDream: Task harmonization inside world models. *International Conference on Machine Learning*, 2024.

Saptarshi Mandal, Xiaojun Lin, and Rayadurgam Srikant. A theoretical analysis of soft-label vs hard-label training in neural networks. *arXiv preprint arXiv:2412.09579*, 2024.

Anusha Nagabandi, Ignasi Clavera, Simin Liu, Ronald S. Fearing, Pieter Abbeel, Sergey Levine, and Chelsea Finn. Learning to adapt in dynamic, real-world environments through meta-reinforcement learning. *International Conference on Learning Representations*, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, USA, 2nd edition, 2009. ISBN 052189560X.

Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.

Silviu Pitis, Elliot Creager, and Animesh Garg. Counterfactual data augmentation using locally factored dynamics. *Advances in Neural Information Processing Systems*, 33:3976–3990, 2020.

Silviu Pitis, Elliot Creager, Ajay Mandlekar, and Animesh Garg. MoCoDA: Model-based counterfactual data augmentation. *Advances in Neural Information Processing Systems*, 35:18143–18156, 2022.

Sai Prasanna, Karim Farid, Raghu Rajan, and André Biedenkapp. Dreaming of many worlds: Learning contextual world models aids zero-shot generalization. *Reinforcement Learning Journal*, 1, 2024.

Jonathan Richens and Tom Everitt. Robust agents learn causal world models. In *The Twelfth International Conference on Learning Representations*, 2024.

Frank Röder, Jan Benad, Manfred Eppe, and Pradeep Kr Banerjee. Dynamics-aligned latent imagination in contextual world models for zero-shot generalization. *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Ruixiang Sun, Hongyu Zang, Xin Li, and Riashat Islam. Learning latent dynamic robust representations for world models. *International Conference on Machine Learning*, 2024a.

Yuewen Sun, Erli Wang, Biwei Huang, Chaochao Lu, Lu Feng, Changyin Sun, and Kun Zhang. ACAMDA: Improving data efficiency in reinforcement learning through guided counterfactual data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 15193–15201, 2024b.

Richard S Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bulletin*, 2(4):160–163, 1991.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press Cambridge, 1998.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. DeepMind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulao, Andreas Kallinteris, Markus Krimmel, Arjun KG, et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024.

Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. DayDreamer: World models for physical robot learning. In *Conference on Robot Learning*, pp. 2226–2240. PMLR, 2023.

Siyuan Zhou, Yilun Du, Jiaben Chen, Yandong Li, Dit-Yan Yeung, and Chuang Gan. RoboDreamer: Learning compositional world models for robot imagination. *International Conference on Machine Learning*, 2024.

A THE IMPLEMENTATION DETAILS OF CAUSALDREAM AND BASELINES

A.1 THE PSEUDOCODE OF CAUSALDREAM

Algorithm 1 CausalDream

Initialize observation o_t, reward r_t, episode continuation flag j_t, action a_t, model state $x_t \doteq [h_t, z_t]$, actor π_ξ, critic v_ψ, exogenous estimator $\mathcal{E}_\theta$, replay buffer $\mathcal{D}$, sequence length L.
for each episode **do**
  // Exogenous Variables Estimation
  Initialize Twin Environment Mechanism $\text{env}^{\text{target}}$ and $\text{env}^{\text{probe}}$.
  Pre-defined agent interacts with $\text{env}^{\text{probe}}$ to collect trajectory $\tau^{\text{probe}} = \{(o_t, a_t)\}_{t=1}^{L}$.
  Estimate exogenous variables $\hat{\mathbf{u}}^c = \mathcal{E}_\theta(\tau^{\text{probe}})$.
  Initialize the target observation and RSSM state.
  if replay buffer $\mathcal{D}$ is empty **then**
    Prefill $\mathcal{D}$ with environment transitions before the first model update.
  end if
  for each $\text{env}^{\text{target}}$ timestep **do**
    for each update step **do**
      // Causal World Model Learning
      Draw batch sequence $\{(o_t, a_t, r_t, j_t, o_{t+1}, \hat{\mathbf{u}}^c)\}$ from replay buffer $\mathcal{D}$.
      Use causal world model to compute model state x_t, future predictions $\hat{x}_{t+1}$ and decode $\hat{o}_t, \hat{r}_t, \hat{j}_t$ and the CCR output $g_\phi(\tau^{\text{latent}})$.
      Update causal world model parameters ϕ by optimizing Equation 4.
      // Policy Learning with Counterfactual Imagination
      Imagine factual rollouts $\mathcal{D}^{\text{F}}$ using π_ξ.
      Sample the first counterfactual action a_1^{CF} uniformly from the action space $\mathcal{A}$.
      Imagine $\mathcal{D}^{\text{CF}}$ with $\text{do}(A_1 = a_1^{\text{CF}})$, fixed $\hat{\mathbf{u}}^c$, and π_ξ from step 2 onward.
      Update π_ξ and v_ψ using $\mathcal{D}^{\text{F}}$ and the masked continuation of $\mathcal{D}^{\text{CF}}$ according to Equation 12.
    end for
    // Environment Interaction
    Update (h_t, z_t) from o_t, the previous action, and the fixed estimate $\hat{\mathbf{u}}^c$.
    Sample $a_t \sim \pi_\xi(\cdot \mid h_t, z_t)$ and step $\text{env}^{\text{target}}$ once to obtain (o_{t+1}, r_t, j_t).
    Store $(o_t, a_t, r_t, j_t, o_{t+1}, \hat{\mathbf{u}}^c)$ in $\mathcal{D}$ and advance to the next timestep.
  end for
end for

A.2 THE IMPLEMENTATION OF EXOGENOUS VARIABLES ESTIMATOR

Simulation-only privileged supervision. We distinguish the information available during estimator pre-training from that required at deployment. During pre-training, CausalDream is trained in a physics simulator where the episode-level dynamics parameters $\mathbf{u}^c$ (e.g., payload mass or actuator strength) are known to the simulator; we use these simulator-provided parameters as privileged supervision to learn an estimator $\mathcal{E}_\theta : \tau \mapsto \hat{\mathbf{u}}^c$. At deployment, the estimator infers the exogenous factors from the probe trajectory without requiring ground-truth parameter inputs. Such training–deployment asymmetry is widely adopted in sim-to-real adaptation, where simulation-accessible physical quantities supervise an adaptation module that subsequently operates in the real world without any privileged information (Kumar et al., 2021). Concretely, the practical analogue of the probe environment in a physical system is a short, safe pre-task probing or calibration phase executed on the same platform before task execution, under the episode-level assumption that the dynamics factors remain unchanged between probing and control.

The procedure for pre-training the estimator is shown in Algorithm 2. To train the estimator $\mathcal{E}_\theta$, we randomize the physical dynamics parameters of the simulator and use a pre-defined agent to interact with it, recording the resulting observation–action trajectories together with the sampled parameter values. These parameters are produced automatically by the simulator's domain randomization and therefore require no manual annotation or real-world measurement. This produces a large

dataset $\{(o_t, a_t, \mathbf{u}^c)\}$. Given that the exogenous variables follow a discrete uniform distribution, the estimation problem is approached as a multi-class classification task. In contrast to the direct regression of exogenous variables, a categorical distribution is more effective at capturing the multi-modal uncertainty.

Taking the physical property of gravity g as an illustrative example, $\mathbf{u}^c = \{g\}^\top$, the range $[g_{\text{min}}, g_{\text{max}}]$ is first discretized into K bins with a step size of Δ. And a discrete support set is defined as $\mathcal{B} = \{b_1, b_2, \ldots, b_K\}$, where b_k denotes the center value of the k-th bin. Due to the ordinal nature of physical properties, the standard one-hot distribution ignores the inherent distance relationships among different classes. To this end, we build a discretized Gaussian distribution P centered on the ground truth value g_{gt} as the target distribution. This method is commonly referred to as soft label (Mandal et al., 2024; de Vries & Thierens, 2025). The probability mass p_k associated with the k-th category is defined as follows:

$$p_k = \frac{1}{Z} \exp\left(-\frac{(b_k - g_{\text{gt}})^2}{2\sigma^2}\right) \tag{7}$$

where σ denotes the smoothing parameter governing the width of the distribution, while Z represents the normalization constant ensuring that the sum of probabilities across all categories equals 1, i.e., $\sum_{k=1}^{K} p_k = 1$. Consequently, categories in closer proximity to the ground truth value are assigned higher probabilities, thereby effectively encouraging the estimator to learn the proximity relationships inherent in physical values.

The estimator $\mathcal{E}_\theta$ comprises a visual encoder CNN and a Gated Recurrent Unit (GRU). The visual encoder takes $\tau = \{(o_t, a_t)\}_{t=1}^{L}$ as input, and the GRU output is processed by Softmax to get the predicted distribution Q_θ. Thus, the training objective of the estimator is to minimize the KL divergence between the predicted distribution Q_θ and the target distribution P:

$$\mathcal{L}(\theta) = \text{KL}(P \parallel Q_\theta) = \sum_{k=1}^{K} p_k \log \frac{p_k}{q_k} \tag{8}$$

Since the target distribution P remains fixed for any given sample, the above optimization is equivalent to minimizing the Cross-Entropy Loss with soft labels:

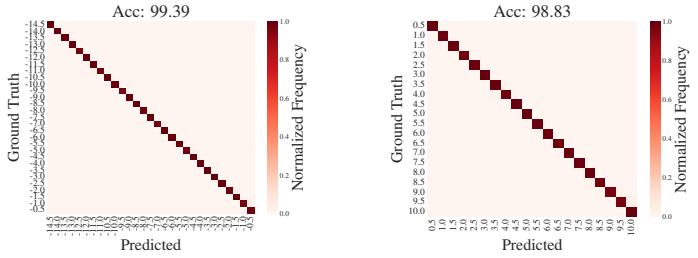
$$\mathcal{L}(\theta) = -\sum_{k=1}^{K} \left[\frac{1}{Z} \exp\left(-\frac{(b_k - g_{\text{gt}})^2}{2\sigma^2}\right) \cdot \log q_k\right] \tag{9}$$

By effectively leveraging distributional information, this loss function facilitates the estimator to learn the proximity relationships inherent in physical values, even in the presence of prediction inaccuracies. For multiple exogenous factors, we construct a separate categorical target for each component and sum their soft-label cross-entropy losses.

Data Collection and Training Setup In this paper, the step size Δ is set to 0.5 or 1.0 according to the task (Section C.2), and the estimator input length is $L = 8$. The pre-defined probe agent executes a task-specific fixed action sequence for both pre-training data collection and inference. Each input contains eight observation–action pairs: the reset observation paired with a zero action, followed by seven observations paired with the fixed actions that produced them. Thus, $L = 8$ counts input pairs, including the reset pair. Figure 7 reports estimation results for Walker Run and Hopper Hop. We set $\sigma = 0.5$ for the Gaussian soft labels. Both endpoints of each parameter range are included, so $K = 1 + (c_{\max} - c_{\min})/\Delta$. Accordingly, $K = 29$ for gravity in Walker, Quadruped, Cheetah, and Cartpole, $K = 23$ for LunarLander gravity, and $K = 20$ for Hopper torso mass. The additional wind and knee-actuator variables in the two-confounder experiments have $K = 5$ and $K = 11$, respectively. To construct the dataset for pre-training the estimator, we deployed the pre-defined probe agent in the probe environment to collect interaction trajectories. The data collection process involved sampling exogenous variables $\mathbf{u}^c$ from a discrete uniform distribution covering the entire training range. In total, we collected a dataset of 3600 trajectories, which was randomly partitioned into a training set of 3000 trajectories and a test set of 600 trajectories. The simulator interactions used to collect this offline estimator pre-training dataset are excluded from the interaction budgets reported in the learning curves; all online probe-environment interactions during policy learning are included. The estimator was trained for 100 epochs on the training set using the Adam optimizer. The results on the test set are reported in Figure 7.

Algorithm 2 Pre-training of Exogenous Variables Estimator (Gravity)

1: ***// Data Collection***
2: Use pre-defined agent to collect data $\{(o_t, a_t, \mathbf{u}^c)\}$ from the environment.
3: ***// Training the Estimator***
4: **for** each training iteration **do**
5: Draw batch sequence $\tau = \{(o_t, a_t)\}_{t=1}^{L}$ with $\mathbf{u}^c$ from the collected data.
6: Discretize $[g_{\min}, g_{\max}]$ into K bins with step Δ to get support set $\mathcal{B} = \{b_1, b_2, \ldots, b_K\}$.
7: Construct target distribution P with Equation 7.
8: Compute predicted distribution Q_θ from estimator $\mathcal{E}_\theta(\tau)$.
9: Update estimator parameters θ by minimizing Equation 9.
10: **end for**
11: ***// Inference***
12: Given trajectory as input and output the class with the highest probability of Q_θ as the estimated exogenous variable $\hat{\mathbf{u}}^c$.



(a) Results on *Walker Run*. (b) Results on *Hopper Hop*.

Figure 7: Confusion matrices for confounder estimation results.

A.3 Additional Details of Causal World Model

In this paper, we develop a Causal World Model building upon the RSSM architecture. Specifically, we explicitly incorporate the estimated exogenous variable into the sequence model and design a Causal Consistency Regularizer. Consequently, components of our proposed Causal World Model are as follows:

$$\begin{aligned}
&\text{Sequence model:} && h_t = f_\phi(h_{t-1}, z_{t-1}, a_{t-1}, \hat{\mathbf{u}}^c) \\
&\text{Posterior:} && z_t \sim q_\phi(z_t \mid h_t, o_t) \\
&\text{Prior:} && \hat{z}_t \sim p_\phi(\hat{z}_t \mid h_t) \\
&\text{Other predictors:} && \hat{y}_t \sim p_\phi(\hat{y}_t \mid h_t, z_t) \\
&\text{Causal Consistency Regularizer:} && g_\phi(\tau^{\text{latent}}) \in \mathbb{R}^N
\end{aligned} \tag{10}$$

with $y_t \in \{o_t, r_t, j_t\}$. The per-step terms in Equation 4 include the loss scales used in our DreamerV3 implementation:

$$\begin{aligned}
\mathcal{L}_{\text{pred}} &\doteq -\log p_\phi(o_t \mid h_t, z_t) - \log p_\phi(r_t \mid h_t, z_t) - \log p_\phi(j_t \mid h_t, z_t), \\
\mathcal{L}_{\text{dyn}} &\doteq 0.5 \max(1, \mathrm{KL}[\mathrm{sg}(q_\phi(z_t \mid h_t, o_t)) \parallel p_\phi(z_t \mid h_t)]), \\
\mathcal{L}_{\text{rep}} &\doteq 0.1 \max(1, \mathrm{KL}[q_\phi(z_t \mid h_t, o_t) \parallel \mathrm{sg}(p_\phi(z_t \mid h_t))]),
\end{aligned} \tag{11}$$

where sg denotes stop-gradient. The three terms are averaged across the $T = 64$ steps of each training sequence before adding one trajectory-level CCR loss.

CCR decoder and implementation. Each training sequence is a contiguous 64-step sample from one episode. The CCR decoder concatenates the RSSM feature $[h_t, z_t]$ with a_t, projects each input to 256 dimensions, and processes the sequence with a single-layer GRU of 256 hidden units. It averages all 64 GRU outputs and applies a 256-to-128 linear layer, SiLU activation, a 128-dimensional residual block with dropout 0.1, and a final linear layer to produce one value per exogenous factor in $\mathbb{R}^N$.

The decoder output $g_\phi(\tau^{\text{latent}})$ reconstructs the exogenous estimate in symlog coordinates using the CCR loss in Equation 3. This trajectory-level loss is added once, with unit scale, to the averaged per-step losses in Equation 4.

A.4 Masked Actor–Critic Objectives for Counterfactual Rollouts

Let i index an imagined rollout and $b_i \in \{\mathrm{F}, \mathrm{CF}\}$ denote its branch. Each rollout contains latent states $x_{i,1:H}$ and $H-1$ return targets. Factual actions follow π_ξ throughout. In the counterfactual branch, $a_{i,1}$ is sampled uniformly from the action space $\mathcal{A}$, while $a_{i,t} \sim \pi_\xi(\cdot \mid x_{i,t})$ for $t \geq 2$; the estimated exogenous variable remains fixed. We exclude the intervened first step from both updates using

$$m_{i,t} = \begin{cases} 0, & b_i = \mathrm{CF} \text{ and } t = 1, \\ 1, & \text{otherwise}, \end{cases} \qquad Z = \sum_i \sum_{t=1}^{H-1} m_{i,t}. \tag{12}$$

The world model still uses the intervened action to reach $x_{i,2}$. With predicted successor reward $\hat{r}_{i,t+1}$ and continuation $\hat{j}_{i,t+1}$, the imagined return is

$$R^\lambda_{i,t} = \hat{r}_{i,t+1} + \gamma \hat{j}_{i,t+1} \left[(1-\lambda) v_\psi(x_{i,t+1}) + \lambda R^\lambda_{i,t+1} \right], \qquad R^\lambda_{i,H} = v_\psi(x_{i,H}). \tag{13}$$

Thus $R^\lambda_{i,2}$ excludes the intervened first-step reward. Let $w_{i,t}$ be the continuation weight, $S \geq 1$ the return-scale normalizer, and $p_\psi(\cdot \mid x)$ the distributional critic with scalar readout $v_\psi(x)$. We denote the slow critic target by $\bar{v}$. The masked losses are

$$\mathcal{L}_{\text{actor}} = -\frac{1}{Z} \sum_i \sum_{t=1}^{H-1} m_{i,t} \left[w_{i,t} G_{i,t} + \eta \mathcal{H}[\pi_\xi(\cdot \mid x_{i,t})] \right], \tag{14}$$

$$\mathcal{L}_{\text{critic}} = -\frac{1}{Z} \sum_i \sum_{t=1}^{H-1} m_{i,t} w_{i,t} \left[\log p_\psi\big(\mathrm{sg}(R^\lambda_{i,t}) \mid \mathrm{sg}(x_{i,t})\big) + \log p_\psi(\mathrm{sg}(\bar{v}(x_{i,t})) \mid \mathrm{sg}(x_{i,t})) \right], \tag{15}$$

where sg stops gradients. For continuous actions, the dynamics-gradient term is $G_{i,t} = (R^\lambda_{i,t} - v_\psi(x_{i,t}))/S$. For discrete actions, the score-function term is $G_{i,t} = \mathrm{sg}(R^\lambda_{i,t} - v_\psi(x_{i,t})) \log \pi_\xi(a_{i,t} \mid x_{i,t})$. The mask removes both the return and entropy terms for the first counterfactual step, as well as its critic target; all subsequent counterfactual actions are generated by the policy.

A.5 Hyperparameters

In this paper, other baseline methods are configured with their default hyperparameters, while the hyperparameters of CausalDream are largely aligned with those of DreamerV3. The key hyperparameters of CausalDream and other methods are listed in Table 3 and 4.

The implementation of all methods:

- **CausalDream**: ***We will release the code upon acceptance.***
- **DreamerV3**: `https://github.com/NM512/dreamerv3-torch`
- **DALI-S-χ**: `https://github.com/frankroeder/DALI`
- **HarmonyDream**: `https://github.com/thuml/HarmonyDream`
- **TD-MPC2**: `https://github.com/nicklashansen/tdmpc2`
- **OMPO**: `https://github.com/Roythuly/OMPO`
- **PPO**: `https://github.com/DLR-RM/stable-baselines3`

A.6 Computational Resources

We implement CausalDream using PyTorch (Paszke et al., 2019) and conduct training on an NVIDIA A100 GPU (40GB). In terms of training duration, for each run on DMC and Gymnasium tasks spanning 500k environment steps, CausalDream takes about 32 hours. Regarding GPU memory usage, these experiments consume approximately 7GB of VRAM. Therefore, they can be readily conducted on standard GPUs with 12GB of VRAM.

Table 3: Hyperparameters of CausalDream and other world models.

Name	CausalDream	TD-MPC2	DreamerV3	HarmonyDream	DALI-S-χ
Observation	RGB	RGB	RGB	RGB	RGB
Batch size	16	256	16	16	16
Batch length	64	3	64	64	64
Optimizer	Adam	Adam	Adam	Adam	Adam
Learning rate	1×10^{-4}	3×10^{-4}	1×10^{-4}	1×10^{-4}	1×10^{-4}
Parameters	18.7M	4.9M	17.9M	17.9M	17.9M
CCR scale	1.0	–	–	–	–
Action repeat	2	2	2	2	2
Time limit	1000	1000	1000	1000	1000
Eval episode	10	10	10	10	10

Table 4: Hyperparameters of Model-Free RL methods.

Hyperparameter	OMPO	PPO
Observation	State	State
Batch size	256	64
Batch length	-	-
Optimizer	Adam	Adam
Learning rate	3×10^{-4}	3×10^{-4}
Total parameters	359,180	9,219
Action repeat	2	2
Time limit	1000	1000
Eval episode	10	10

B Additional Related Works

Conventional World Models In the Dreamer (Hafner et al., 2025), the world model primarily focuses on observation modeling and reward modeling, with a pronounced emphasis on the former during training. This overemphasis on observation modeling can lead the world model to incorrectly learn correlations between task-irrelevant observational details (e.g., background textures) and actions or rewards. To mitigate this problem, HarmonyDream (Ma et al., 2024) introduces a harmonizer module that adaptively adjusts the loss coefficients for observation modeling and reward modeling, with experimental results confirming its efficacy. However, loss balancing alone does not explicitly identify episode-level physical factors such as gravity and mass. When these factors vary, adaptation still depends on whether the learned latent representation captures their effects. TD-MPC (Hansen et al., 2022), as another popular world model architecture, integrates the strengths of model-based and model-free approaches. It employs a task-oriented latent dynamics model for short-horizon trajectory optimization to mitigate high planning costs and leverages temporal difference learning to jointly train a terminal value function for estimating long-term rewards. Building on this, TD-MPC2 (Hansen et al., 2024) enhances robustness, scalability, and multi-task adaptability through various improvements. However, its task-oriented world model does not explicitly estimate the episode-level physical factors underlying dynamics shifts.

Our work focuses on investigating the impact of confounders on state dynamic transitions and constructing a causal world model for deconfounding dynamics. Compared to other world model approaches, our method conditions latent dynamics on estimated exogenous factors, encourages their retention, and uses fixed-exogenous counterfactual imagination for data augmentation, aiming to improve policy robustness and sample efficiency.

Causal World Models and Counterfactual RL Causal Reinforcement Learning includes a class of reinforcement learning methodologies that integrate supplementary assumptions or prior knowledge to analyze and comprehend the underlying causal mechanisms governing actions and their consequences, thereby empowering the agent to formulate more judicious and efficacious decisions

(Deng et al., 2023). Unlike traditional reinforcement learning methods that primarily rely on statistical correlations between observations and rewards, causal RL emphasizes the identification and utilization of causal relationships among variables in the environment, which is crucial for enhancing the robustness, generalization, and interpretability of policy (Deng et al., 2023; He et al., 2022). Sample efficiency remains a central challenge in reinforcement learning research. Achieving high performance with minimal interaction data is desirable for reducing computational costs and training duration. Specifically, recent studies (Buesing et al., 2019; Lu et al., 2020; Pitis et al., 2022; Sun et al., 2024b) have explored the use of counterfactual data augmentation which is a causality-driven technique for generating synthetic counterfactual experiences, thereby decreasing the need for real-world interactions and boosting data efficiency.

In this paper, we introduce a causal world model that deconfounds dynamics by conditioning on estimated exogenous factors, thus supporting fixed-exogenous counterfactual imagination. Subsequently, motivated by these developments, we incorporate counterfactual inference into our policy learning phase to improve policy.

Contextual RL As a significant branch of reinforcement learning, Contextual RL centers on explicitly incorporating context which serves as prior information into the sequential decision-making framework (Hallak et al., 2015). While conventional RL problems are typically formulated as Markov Decision Processes (MDPs), real-world environments are rarely stationary. Instead, they often exhibit non-stationary dynamics influenced by various contextual factors. Contextual RL was specifically proposed to address this problem. Built on the RSSM, context RSSM (cRSSM) (Prasanna et al., 2024) is a contextual world model aimed at enabling zero-shot generalization in unseen dynamics. It explicitly incorporates context, such as gravity and friction, into the RSSM's key components, including the sequence model, prior, posterior, and predictors. Additionally, the policy network also takes the context as input. This design enforces contextual constraints across the model, guiding both dynamics and policy learning.

Both ContextDream and CausalDream inject the estimated exogenous variables into the recurrent sequence model. The resulting deterministic state h_t carries context information to the downstream RSSM components and policy. ContextDream therefore controls for contextual conditioning within our architecture, while cRSSM uses a different design with direct context inputs to multiple components.

DALI-S-χ (Röder et al., 2025) is a recent contextual world model, also grounded in DreamerV3. However, unlike cRSSM, which directly utilizes explicit confounders like gravity as contexts, DALI-S-χ introduces a self-supervised context encoder to infer latent context from interaction history. Further, it employs a forward dynamics loss and a cross-modal regularization loss to ensure dynamics effectively encodes contextual information.

Latent context inference versus factor-aligned exogenous estimation. Self-supervised contextual world models provide an attractive way to infer dynamics-relevant information without explicit context labels. DALI, for example, learns a latent context representation from interaction histories using a forward-dynamics objective. However, predictive context identification and factor-level identification serve different purposes. A self-supervised representation can encode sufficient information about environmental dynamics without assigning individual coordinates to specific physical factors. Unsupervised disentanglement is not identifiable without additional structural assumptions or inductive biases (Locatello et al., 2019). CausalDream instead exploits simulator-accessible supervision to anchor the estimated variables to the physical factors defining the SCM. This explicit alignment is not required for contextual adaptation in general, but provides a well-defined exogenous coordinate system for the fixed-exogenous action counterfactuals considered in our framework.

The episodically non-stationary environments considered in this work can also be represented as Contextual MDPs or Hidden-Parameter MDPs when the episode-level dynamics factor is treated as a context variable. We do not claim that the SCM formulation defines a strictly more general environment class than these frameworks. Instead, the causal formulation provides the theoretical foundation for counterfactual reasoning in model-based imagination. While contextual MDPs can also keep the context fixed during rollouts, the SCM framework is required for formally defining interventions and counterfactuals in the sense of Pearl's causal hierarchy. CausalDream operationalizes this theoretical

framework through exogenous-variable estimation, information-preserving dynamics learning, and fixed-exogenous counterfactual imagination.

C Environment Details

C.1 Task Visualization

In this paper, a set of 9 control tasks, comprising 8 continuous control tasks from the DMC benchmark and 1 discrete control task from the Gymnasium benchmark, are selected for evaluation. Characterized by diverse objectives and varying levels of complexity, these tasks are designed to comprehensively assess the performance of CausalDream and other baseline methods. The visualization of these tasks is presented in Figure 8, where each task is represented by a distinct image. The tasks include *Walker Walk*, *Walker Run*, *Hopper Stand*, *Hopper Hop*, *Quadruped Walk*, *Quadruped Run*, *Cheetah Run*, *Cartpole Swingup*, and *LunarLander*. The task specifications are summarized in Table 5.

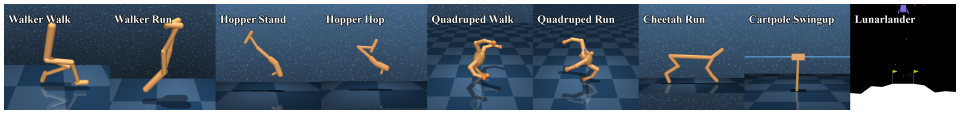


Figure 8: The visualization of tasks selected in this paper. Each task is represented by a distinct image, showcasing the unique objectives and environments of the tasks.

Table 5: The description of tasks selected in this paper. The state and action dimensions are specified, along with the metric used for evaluation and whether the task is continuous or discrete.

Task	**State dim**	**Action dim./count**	**Metric**	**Continuous/Discrete**
Cartpole Swingup	5	1	Reward	Continuous
Cheetah Run	17	6	Reward	Continuous
Hopper Stand	15	4	Reward	Continuous
Hopper Hop	15	4	Reward	Continuous
Quadruped Walk	78	12	Reward	Continuous
Quadruped Run	78	12	Reward	Continuous
Walker Walk	24	6	Reward	Continuous
Walker Run	24	6	Reward	Continuous
LunarLander	8	4 actions	Reward	Discrete

C.2 Non-stationary Environments Settings

In this paper, all methods are trained and evaluated in non-stationary environments, as shown in Figure 9. Specifically, the environmental confounders are modeled as exogenous variables sampled from a discrete uniform distribution. At the beginning of each episode, a confounder value is sampled from a specified range $[c_{\min}, c_{\max}]$ with a fixed discretization step size Δ. This value remains constant throughout the episode but varies stochastically across episodes. The detailed parameter configurations for each task are outlined in Table 6.

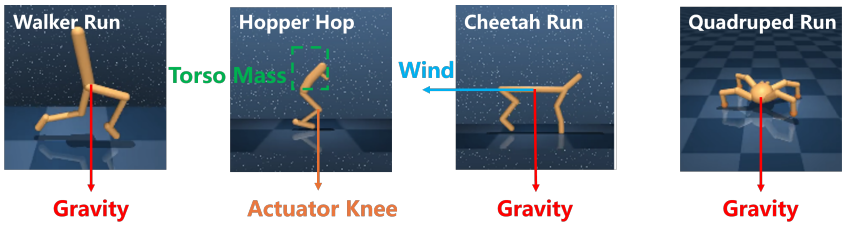


Figure 9: The visualization of confounders in partial tasks. The confounders are represented by distinct images, showcasing their influence on the dynamics of the tasks.

Table 6: The environment settings of experiments with single confounders.

Task	Confounder	Train	Evaluation	Step Size	Episode length
Walker Walk	Gravity	[-14.5, -0.5]	[-14.5, -0.5]	0.5	1000
Walker Run	Gravity	[-14.5, -0.5]	[-14.5, -0.5]	0.5	1000
Quadruped Walk	Gravity	[-14.5, -0.5]	[-14.5, -0.5]	0.5	1000
Quadruped Run	Gravity	[-14.5, -0.5]	[-14.5, -0.5]	0.5	1000
Cheetah Run	Gravity	[-14.5, -0.5]	[-14.5, -0.5]	0.5	1000
Cartpole Swingup	Gravity	[-14.5, -0.5]	[-14.5, -0.5]	0.5	1000
LunarLander	Gravity	[-11.5, -0.5]	[-11.5, -0.5]	0.5	1000
Hopper Stand	Torso Mass	[0.5, 10.0]	[0.5, 10.0]	0.5	1000
Hopper Hop	Torso Mass	[0.5, 10.0]	[0.5, 10.0]	0.5	1000

Experiments with Multiple Confounders To assess the capability of CausalDream in handling complex dynamics shifts, we design scenarios governed by multiple simultaneous exogenous variables. In these experiments, the environment is controlled by two distinct confounders denoted as $\mathbf{u}^c = [u_1^c, u_2^c]^\top$. Each confounder u^c follows an independent discrete uniform distribution defined by its specific range and step size. Consequently, at the initialization of each episode, both confounders are sampled simultaneously, resulting in a compound perturbation to the transition dynamics. This setting presents a significantly harder challenge than single-confounder tasks, as the agent must account for the combined effects of multiple latent causes to achieve robust control. The detailed configurations for these experiments are listed in Table 7.

Table 7: Environment settings for experiments with multiple confounders. The viscosity and density in *Cheetah Run* are set to 0.2 and 1.2, respectively.

Task	Confounder	Train	Evaluation	Step Size	Episode length
Cheetah Run	Gravity	[-14.5, -0.5]	[-14.5, -0.5]	0.5	1000
	Wind	[-5.0, -1.0]	[-5.0, -1.0]	1.0	1000
Hopper Hop	Torso Mass	[0.5, 10.0]	[0.5, 10.0]	0.5	1000
	Actuator Knee	[25.0, 35.0]	[25.0, 35.0]	1.0	1000

Generalization to Unseen Dynamics Beyond standard non-stationary environments, we further evaluate the robustness of CausalDream by testing its generalization capabilities on Out-of-Distribution dynamics. In these experiments, the range of confounders used for evaluation extends significantly beyond the range used during training. Specifically, the agent is trained on a limited set of confounder values (Training Domain) but must adapt to unseen values (Evaluation Domain) during testing. This setup simulates extreme scenarios where environmental conditions drift into unknown territories. The configuration for these generalization tasks is detailed in Table 8.

Table 8: The environment settings of generalization experiment.

Task	Confounder	Train	Evaluation	Step Size	Episode length
Hopper Hop	Torso Mass	[0.5, 10.0]	[10.5, 12.0]	0.5	1000
Walker Run	Gravity	[-14.5, -0.5]	[-20.0, -15.0]	0.5	1000
Cheetah Run	Gravity	[-14.5, -0.5]	[-20.0, -15.0]	0.5	1000
Cartpole Swingup	Gravity	[-14.5, -0.5]	[-20.0, -15.0]	0.5	1000

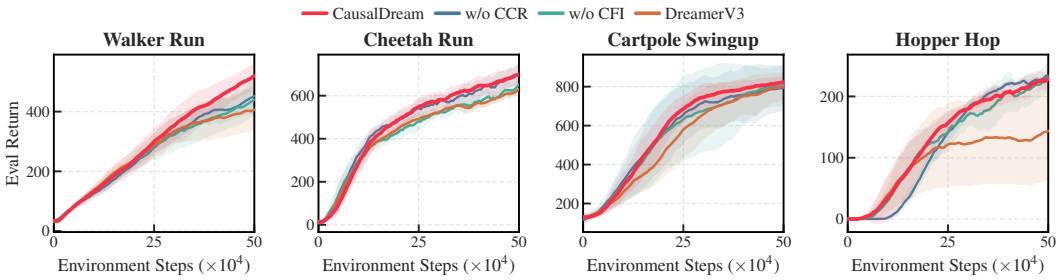


Figure 10: Ablation studies on the *Walker Run*, *Cheetah Run*, *Cartpole Swingup*, and *Hopper Hop* tasks under single confounder setting.

D Ablation Studies

We perform ablation studies to examine the contributions of the Twin Environment Mechanism (TEM), Causal Consistency Regularizer (CCR), and Counterfactual Imagination (CFI). In addition to the component-wise comparisons in Figure 6, Figure 10 evaluates CCR and CFI across four continuous-control benchmarks, while Table 9 further examines their effects on intervention-response prediction.

Effectiveness of Twin Environment Mechanism In the *w/o TEM* variant, the estimator receives the same eight-pair probe sequence collected in the target environment, including the reset observation paired with a zero action. Rewards from the probing interactions contribute to the episode return. After estimating $\hat{\mathbf{u}}^c$ once, normal RSSM-based policy control begins and the estimate remains fixed for the rest of the episode. The first and third panels of Figure 6 show that CausalDream achieves higher final mean returns on both tasks. TEM enables policy control from the start of the target episode, avoiding the low-reward probing phase within that episode.

Analysis of Causal Consistency Regularizer CCR is introduced to encourage latent trajectories to preserve information about the estimated exogenous variable. We compare the RNN-based CCR used in CausalDream with an MLP-based alternative and the variant without CCR. The first and third panels of Figure 6 show that the RNN-based CCR achieves higher final mean returns than both the MLP alternative and the variant without CCR on *Walker Walk* and *LunarLander*. This suggests that conditioning the dynamics model on the estimated exogenous variable alone is insufficient; explicitly constraining the latent trajectory to retain this information provides an additional benefit.

The additional results in Figure 10 reinforce this observation: removing CCR consistently reduces evaluation performance across the four continuous-control benchmarks. The intervention-response results provide complementary evidence. As shown in Table 9, removing CCR increases both Return and Reward effect MSE on all four tasks and also increases candidate-selection regret. Thus, CCR improves not only downstream control performance but also the model's ability to preserve dynamics information relevant to action interventions.

Analysis of Counterfactual Imagination We next investigate the effectiveness of CFI in enhancing policy learning. Experimental results presented in Figure 6 and Figure 10 demonstrate that CausalDream consistently outperforms its counterpart without CFI across most of the training process. By generating counterfactual trajectories through intervention on actions, CausalDream effectively mitigates policy bias and augments the data distribution, thereby facilitating the development of a more robust and stable policy. The variant without CFI still outperforms the conventional baseline

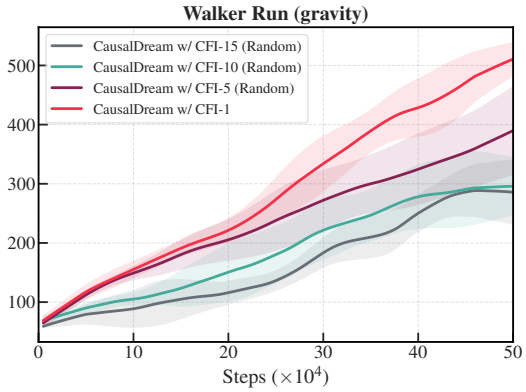


Figure 11: Effect of intervention frequency on *Walker Run*. CFI-1 intervenes only at the first imagined step; CFI-k ($k = 5, 10, 15$) intervenes at k randomly selected steps.

in several settings, indicating that the Causal World Model itself already contributes to improved control, while CFI provides an additional policy-learning benefit.

We also preliminarily investigate the impact of the mixing of factual and counterfactual trajectories during policy learning. The variant w/o CFI corresponds to using factual trajectories only, and the w/ CFI-Only variant corresponds to relying solely on counterfactual trajectories, whereas CausalDream employs a balanced mixture with factual and counterfactual trajectories each constituting 50%.

The second and fourth panels of Figure 6 show that the balanced factual–counterfactual mixture achieves higher final mean returns than either factual-only or counterfactual-only training on both *Walker Walk* and *LunarLander*. This empirical finding validates our hypothesis that the efficacy of counterfactual data augmentation lies in its ability to complement rather than replace factual experiences.

Analysis of the Number of Interventions To analyze the effect of intervention frequency, we conduct an ablation study on the *Walker Run* task by varying the number of intervened time steps during imagined rollouts. Specifically, we compare the default setting that intervenes only at the first step with variants that randomly select 5, 10, and 15 time steps for intervention. As shown in Figure 11, the performance gradually decreases as the number of interventions increases, with CFI-1 achieving the best result. This suggests that counterfactual intervention is more effective as a moderate augmentation to factual trajectories rather than as a frequent replacement of policy-generated actions. Excessive random interventions may shift imagined trajectories away from the current policy distribution and introduce additional prediction errors and training noise. Therefore, we adopt the design of performing a single intervention at the first step of each imagined rollout.

Ablation on Counterfactual Action Effects To connect the component ablations with intervention-based prediction, we evaluate *w/o CCR* and *w/o CFI* using the benchmark in Section 5.3. Table 9 shows that removing CCR increases Return and Reward effect MSE and mean candidate-selection regret on all four tasks. Removing CFI increases Reward effect MSE on all four tasks and Return effect MSE on three tasks, with a small decrease on *Cheetah Run*. CFI also reduces mean regret on *Walker Run* and *Hopper Hop*, while *w/o CFI* achieves slightly lower mean regret on *Cheetah Run* and *Cartpole Swingup*. These results support CCR's contribution to action-effect prediction and show that CFI's benefits for candidate selection vary across tasks.

Overall Interpretation. The ablations show complementary roles for the three components. TEM identifies the episode-level dynamics condition, CCR encourages this information to remain represented throughout latent dynamics, and CFI exploits the resulting world model to augment policy learning with alternative action outcomes. Removing any of these components weakens the complete framework, while the counterfactual action-effect evaluation further shows that CCR is particularly important for intervention-consistent dynamics prediction. Together, these results support the design

Table 9: Full action-effect metrics for the component ablations ($H = 10$); Return and Regret are also shown in Table 2. All variants receive ground-truth exogenous parameters. Cells show mean$_{\pm \mathrm{SD}}$ across seeds; bold marks the best means among the three listed variants. Return/Reward/Image denote effect MSE. All values (mean and SD) are in units of 10^{-3}.

	Cheetah Run				**Walker Run**			
Method	Return ↓	Reward ↓	Image ↓	Regret ↓	Return ↓	Reward ↓	Image ↓	Regret ↓
w/o CCR	$93.8_{\pm 30}$	$4.41_{\pm 1.0}$	$0.74_{\pm 0.17}$	$101.1_{\pm 30}$	$300.2_{\pm 75}$	$24.03_{\pm 4.4}$	$\mathbf{13.95}_{\pm 0.59}$	$332.1_{\pm 57}$
w/o CFI	$\mathbf{78.9}_{\pm 18}$	$4.18_{\pm 1.1}$	$0.73_{\pm 0.14}$	$\mathbf{93.0}_{\pm 27}$	$292.0_{\pm 59}$	$23.80_{\pm 3.9}$	$13.96_{\pm 0.65}$	$337.3_{\pm 47}$
CausalDream	$79.9_{\pm 12}$	$\mathbf{4.02}_{\pm 0.69}$	$\mathbf{0.63}_{\pm 0.13}$	$98.9_{\pm 25}$	$\mathbf{254.2}_{\pm 47}$	$\mathbf{22.60}_{\pm 3.7}$	$14.22_{\pm 0.48}$	$\mathbf{291.3}_{\pm 21}$
	Hopper Hop				**Cartpole Swingup**			
Method	Return ↓	Reward ↓	Image ↓	Regret ↓	Return ↓	Reward ↓	Image ↓	Regret ↓
w/o CCR	$605.1_{\pm 130}$	$51.76_{\pm 8.5}$	$1.80_{\pm 0.27}$	$277.2_{\pm 45}$	$155.8_{\pm 270}$	$5.49_{\pm 9.5}$	$0.32_{\pm 0.16}$	$44.1_{\pm 44}$
w/o CFI	$611.4_{\pm 110}$	$53.94_{\pm 8.6}$	$1.81_{\pm 0.25}$	$273.0_{\pm 92}$	$40.0_{\pm 54}$	$1.13_{\pm 1.3}$	$0.28_{\pm 0.093}$	$\mathbf{27.8}_{\pm 24}$
CausalDream	$\mathbf{463.1}_{\pm 78}$	$\mathbf{46.91}_{\pm 11}$	$\mathbf{1.62}_{\pm 0.15}$	$\mathbf{220.0}_{\pm 107}$	$\mathbf{30.2}_{\pm 8.9}$	$\mathbf{0.78}_{\pm 0.24}$	$\mathbf{0.24}_{\pm 0.048}$	$29.2_{\pm 7.8}$

of CausalDream as a combination of exogenous-variable estimation, causally constrained dynamics learning, and counterfactual policy optimization.

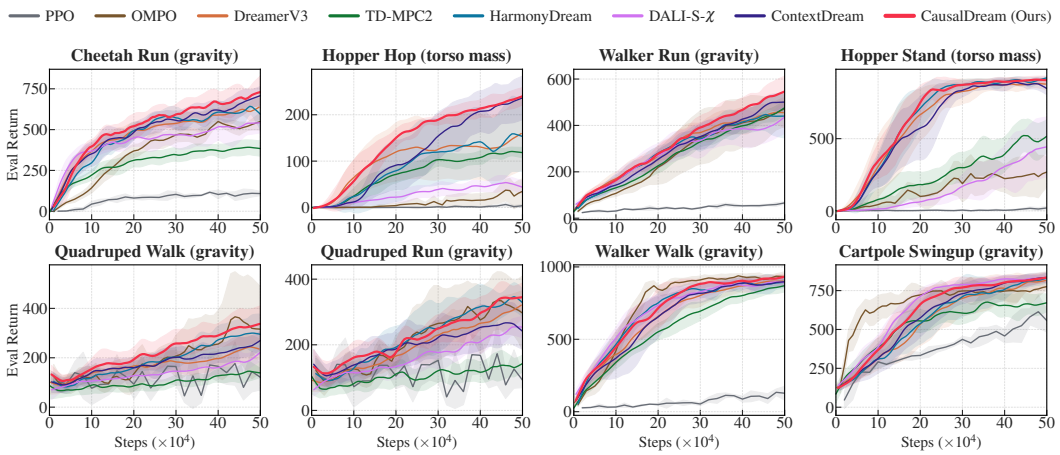


Figure 12: Evaluation learning curves of CausalDream and baseline algorithms across all eight DMC continuous control tasks under single-confounder variations. Each curve depicts the mean evaluation return over five random seeds, with shaded regions representing ± 1 standard deviation. The evaluated confounding factor for each environment is indicated in parentheses.

E Extended Experiment Results

E.1 Evaluation Learning Curves on Individual DMC Tasks

Figure 12 presents the individual evaluation learning curves for CausalDream and all baseline methods across all eight continuous control tasks in the DMC benchmark under single-confounder variations. In each environment, transition dynamics shift episodically according to the physical confounder indicated in parentheses. CausalDream achieves higher learning AUC than ContextDream on all eight tasks, with particularly large gains on *Hopper Hop* and *Quadruped Walk* (Section E.2).

E.2 Aggregate Sample Efficiency: Metric and Coverage

Interaction steps count individual environment transitions, including each repeated execution of an action. All methods, including PPO, use an action repeat of 2 and the same 500k-step budget under this convention. Evaluation is performed every 5,000 environment steps for all methods.

For task q, method m, and seed s, let $R_{q,m,s}(t)$ be the unsmoothed evaluation return at interaction step t. We compute the time-averaged area under its learning curve,

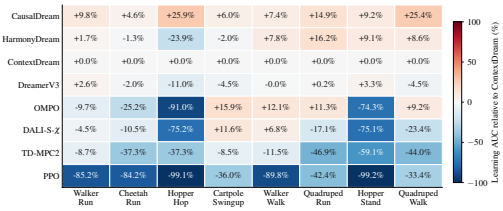
$$A_{q,m,s} = \frac{1}{b-a}\int_a^b R_{q,m,s}(t)\,dt, \qquad S_m = \frac{1}{8}\sum_{q=1}^{8} \frac{\overline{A}_{q,m}}{\overline{A}_{q,\mathrm{ContextDream}}}, \tag{16}$$

where $a = 20{,}000$, $b = 500{,}000$, and $\overline{A}_{q,m}$ is the equally weighted seed mean. Integration uses the trapezoidal rule on each seed's native evaluation grid, with linear interpolation between adjacent observations at the boundaries and no extrapolation. Normalization occurs after seed averaging, and each task has weight $1/8$. All comparisons use the same 20k–500k integration window shown in Figure 5a.

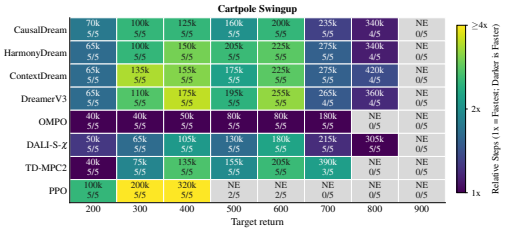
Per-task relative AUC and threshold attainment. Figure 13a visualizes the relative learning AUC of all evaluated algorithms across each of the eight individual DMC environments with respect to ContextDream. Furthermore, Figures 13(b–h) present the target return threshold attainment results across the remaining seven DMC tasks, complementing the *Quadruped Walk* evaluation shown in Figure 5b.

A seed confirms a target when its unsmoothed evaluation return is at least the threshold for three consecutive recorded evaluations; the confirmation time is the step of the third evaluation. For each method and task, we report the median confirmation time across five seeds, treating unconfirmed runs

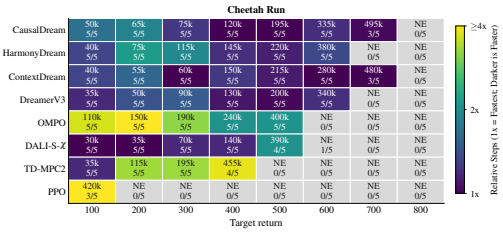
as ∞. Confirmation steps are reported without rounding. NE indicates that fewer than three seeds confirm the target within the evaluation budget. The lower entry reports confirmed seeds out of five. Cell colors compare the reported steps with the fastest method at the same threshold (darker indicates faster).



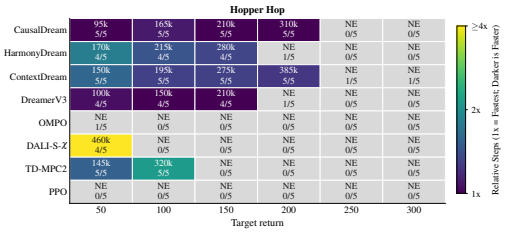
(a) Per-task relative AUC (%).



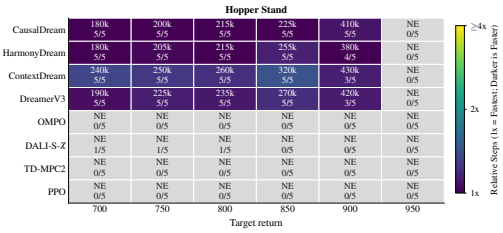
(b) Cartpole Swingup.



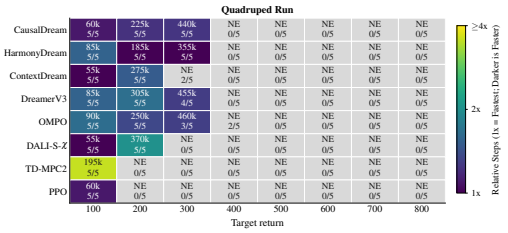
(c) Cheetah Run.



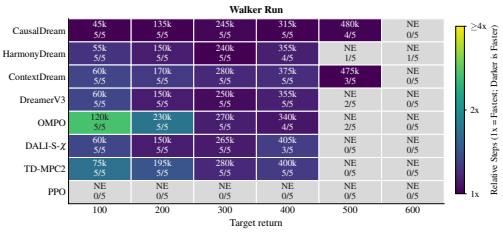
(d) Hopper Hop.



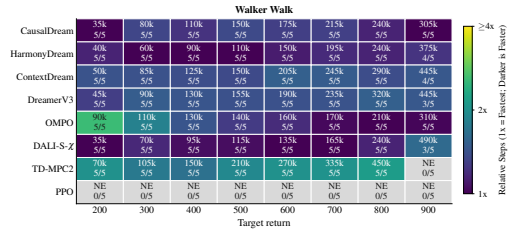
(e) Hopper Stand.



(f) Quadruped Run.



(g) Walker Run.



(h) Walker Walk.

Figure 13: Extended sample efficiency and threshold attainment across DMC tasks under non-stationary dynamics. (a) Heatmap of learning AUC relative to ContextDream (%) across all eight tasks. (b–h) Target return threshold attainment on the remaining seven DMC tasks. Upper entries give median target-confirmation steps across five seeds; lower entries report confirmed seeds out of five. NE indicates that fewer than three seeds confirm the target within the evaluation budget. Colors compare the displayed steps with the fastest method at the same threshold (darker indicates faster).

E.3 Performance Dissection across Task Complexity

To examine how task complexity affects performance under non-stationary dynamics, we group the DMC benchmarks into basic tasks (*Walker Walk*, *Hopper Stand*, and *Quadruped Walk*) and more demanding locomotion tasks (*Walker Run*, *Hopper Hop*, *Quadruped Run*, and *Cheetah Run*). *Cartpole*

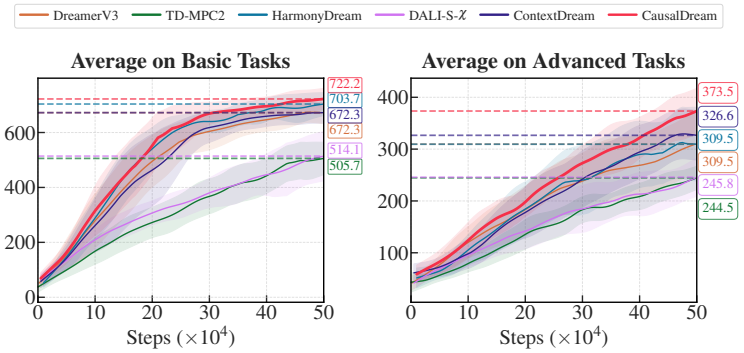


Figure 14: Aggregated performance comparison across task difficulty on DMC tasks under non-stationary dynamics. Left: Average learning curves on basic tasks (*Walker Walk, Hopper Stand, Quadruped Walk*). Right: Average learning curves on advanced tasks (*Walker Run, Hopper Hop, Quadruped Run, Cheetah Run*). Curves report the mean evaluation return over five random seeds with shaded areas indicating one standard deviation. Dashed horizontal lines and right-hand boxes display the final evaluation returns achieved at 500k environment steps.

Swingup is not included in either group, as it is neither a *Stand*/*Walk* nor a *Run*/*Hop* locomotion task. Figure 14 compares the aggregated learning curves of the two groups.

For each task group, we first average the final returns across tasks for each method, then take the arithmetic mean of the five baseline values from DreamerV3, TD-MPC2, HarmonyDream, DALI-S-$\mathcal{X}$, and ContextDream. The relative gain is $100(R_{\text{CausalDream}}/R_{\text{baseline}} - 1)\%$. Using the final returns displayed in Figure 14, the baseline means are 613.62 for basic tasks and 287.18 for advanced tasks; the corresponding CausalDream returns of 722.2 and 373.5 yield gains of 17.69% and 30.06%, respectively.

On the basic tasks, most strong world-model baselines remain competitive, with DreamerV3, HarmonyDream, and ContextDream achieving similar final performance. CausalDream nevertheless obtains the highest average return of 722.2, corresponding to a 17.69% improvement over the mean return of the five world-model baselines. This suggests that when control demands are relatively moderate, existing models can already tolerate part of the dynamics variation, leaving a smaller margin for improvement.

The difference becomes substantially larger on the more challenging *Run* and *Hop* tasks. These tasks require faster motion, more frequent contacts, and more precise multi-step control, making policy performance more sensitive to errors in the learned dynamics. Under these conditions, CausalDream reaches an average final return of 373.5, compared with 326.6 for ContextDream and 309.5 for DreamerV3, corresponding to a 30.06% improvement over the mean return of the five world-model baselines. The widening gap from basic to advanced tasks indicates that the benefit of CausalDream becomes more pronounced as accurate dynamics modeling becomes increasingly important.

ContextDream already conditions its dynamics model on the same estimated exogenous information, but the performance gap remains clear on the advanced tasks. This result suggests that explicit exogenous-variable conditioning alone does not fully address complex dynamics variation, and supports the additional benefit of the causal consistency and counterfactual components used in CausalDream.

E.4 Improvement across Confounder Spectrum

Figure 15 compares the performance gains of CausalDream across different torso-mass and gravity settings. CausalDream outperforms DreamerV3 at every evaluated setting and HarmonyDream, TD-MPC2, and DALI-S-$\mathcal{X}$ across nearly the entire evaluated dynamics range. Compared with ContextDream, performance varies with torso mass on *Hopper Hop*, with ContextDream achieving higher returns at several larger-mass settings. On *Walker Run*, CausalDream outperforms ContextDream at 28 of the 29 evaluated gravity values. Overall, these results show that the gains of CausalDream

persist across diverse dynamics conditions rather than being driven by a few favorable configurations.

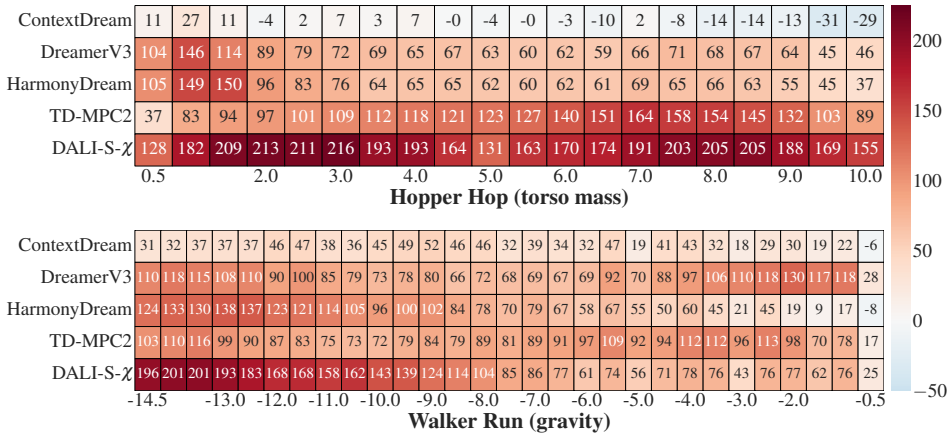


Figure 15: Heatmap of Performance Gain across the confounder spectrum. The results are averaged over 10 runs and 5 seeds at 500K steps.

E.5 Evaluation under Multiple Confounders

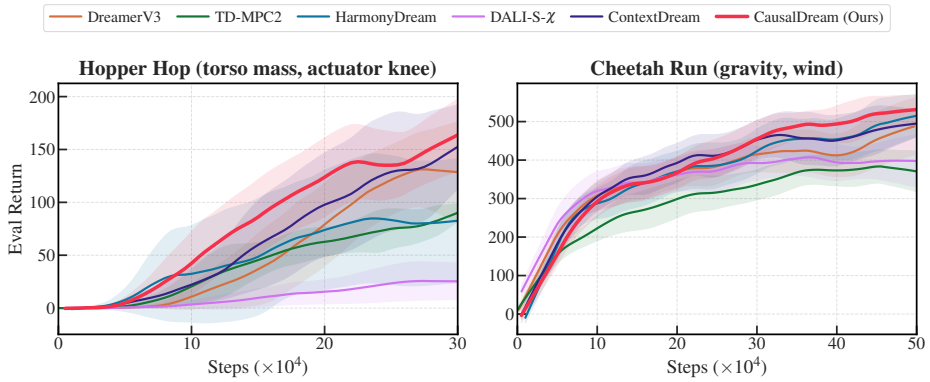


Figure 16: Performance under compound dynamics shifts. Left: *Hopper Hop* with varying torso mass and actuator knee torque. Right: *Cheetah Run* with varying gravity and wind. Curves show the mean over five seeds, with shaded regions indicating one standard deviation.

We further evaluate CausalDream under compound dynamics shifts, where two physical factors vary simultaneously: torso mass and actuator knee torque in *Hopper Hop*, and gravity and wind in *Cheetah Run*. As shown in Figure 16, CausalDream achieves the strongest overall learning performance in both settings. On *Hopper Hop*, it learns substantially faster than most baselines and reaches the highest final return, with ContextDream being the closest competitor. On *Cheetah Run*, the methods perform similarly during early training, but CausalDream continues to improve and establishes a clear advantage in the later stages.

The consistent advantage over ContextDream is particularly informative, since both methods use the same estimated exogenous information. This suggests that explicitly conditioning on multiple dynamics factors alone is insufficient, while CausalDream remains effective when several sources of dynamics variation act jointly.

E.6 Additional Long-term Video Prediction Results

To further assess the long-term video prediction performance of CausalDream, we conducted additional experiments on the *Quadruped Walk* and *LunarLander* tasks. As shown in Figure 17b and 17c, CausalDream consistently outperforms other world model methods on both tasks.

E.7 Latent Representations for Dynamics Deconfounding

To examine the latent representations underlying dynamics deconfounding, Figure 18 visualizes latent states on *Hopper Hop*. CausalDream exhibits clearer separation across torso-mass values than DreamerV3, consistent with the exogenous-information retention encouraged by CCR. The intervention-based evaluations in Section 5.3 further assess how accurately the model predicts action effects under fixed physical conditions.

E.8 Additional Generalization Results

The OOD evaluation measures CWM and policy generalization with ground-truth exogenous inputs supplied to CausalDream and ContextDream. The input-error evaluation in Section E.9 separately measures how incorrect exogenous inputs affect closed-loop performance.

To further evaluate generalization under unseen dynamics, we report the out-of-distribution episode returns on the remaining tasks, *Hopper Hop* and *Cheetah Run*, following the environment configurations specified in Table 8. As shown in Table 10, CausalDream achieves the highest mean return on *Hopper Hop* and performance comparable to ContextDream on *Cheetah Run*.

Table 10: OOD episode returns on *Hopper Hop* and *Cheetah Run* (higher is better). CausalDream and ContextDream receive ground-truth exogenous parameters. Cells show mean$_{\pm\mathrm{SD}}$ across seeds; bold and underline mark the best and second-best means per task, respectively.

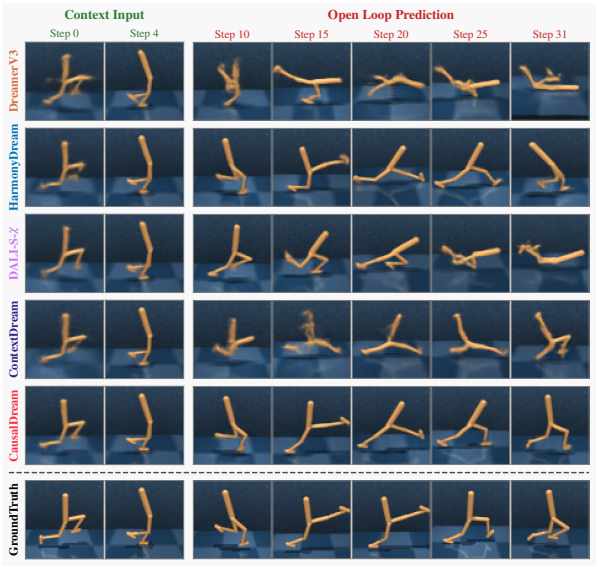
Method	Hopper Hop	Cheetah Run
DreamerV3	$55.93_{\pm 55.26}$	$457.20_{\pm 148.84}$
TD-MPC2	$50.50_{\pm 4.10}$	$237.29_{\pm 56.19}$
HarmonyDream	$28.00_{\pm 18.60}$	$371.21_{\pm 68.88}$
DALI-S-$\mathcal{X}$	$3.20_{\pm 2.90}$	$462.29_{\pm 75.87}$
OMPO	$11.30_{\pm 15.20}$	$426.98_{\pm 90.16}$
ContextDream	$\underline{93.73}_{\pm 29.57}$	$\mathbf{599.05}_{\pm 45.48}$
CausalDream	$\mathbf{96.60}_{\pm 16.30}$	$\underline{594.42}_{\pm 112.11}$

E.9 Robustness under Exogenous Context Errors

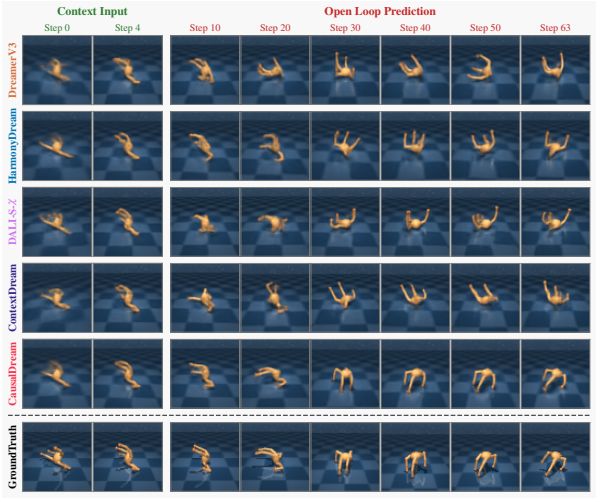
To evaluate how errors in the estimated exogenous variables affect downstream control, we simulate estimator errors on *Walker Run*, *Cheetah Run*, *Hopper Hop*, and *Cartpole Swingup*. Each task varies a scalar exogenous factor; we supply $u^c_{\text{in}} = u^c + \epsilon$ to the trained agent, with signed perturbations $\epsilon \in \{-2, -1, +1, +2\}$. The true environment parameter remains unchanged, and both u^c and u^c_{in} lie within the training range. We measure the resulting change in closed-loop return relative to the accurate input ($\epsilon = 0$).

Figure 19 reports the relative return change with respect to the clean input ($\epsilon = 0$), while Table 11 provides the corresponding absolute returns. This setting evaluates whether the learned policy remains effective when the estimated dynamics context is imperfect.

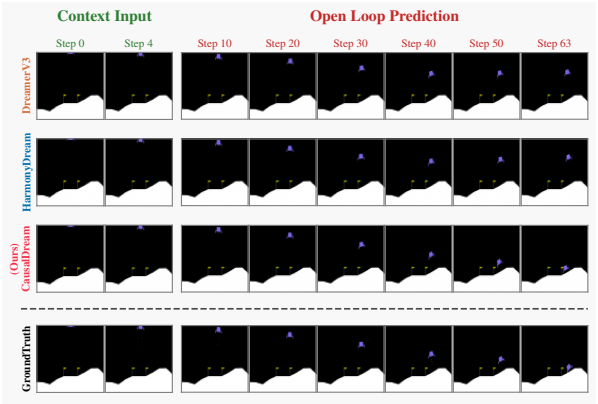
Overall, CausalDream is less sensitive to context-estimation errors in most settings while maintaining higher absolute returns than ContextDream. On *Walker Run* and *Cheetah Run*, both methods are relatively stable, but CausalDream preserves a consistently stronger policy across perturbations. The difference becomes more pronounced on the more sensitive *Cartpole Swingup*, where CausalDream substantially reduces the performance loss under large errors ($|\epsilon| = 2$). *Hopper Hop* is more challenging for both methods and exhibits stronger degradation, although CausalDream remains competitive across perturbation levels. These results suggest that the causal dynamics representation is



(a) Long-term prediction on *Walker Run* across horizons.



(b) Long-term video prediction on the *Quadruped Walk* task.



(c) Long-term video prediction on *LunarLander*, comparing DreamerV3, HarmonyDream, and CausalDream.

Figure 17: Long-term video prediction of different world models

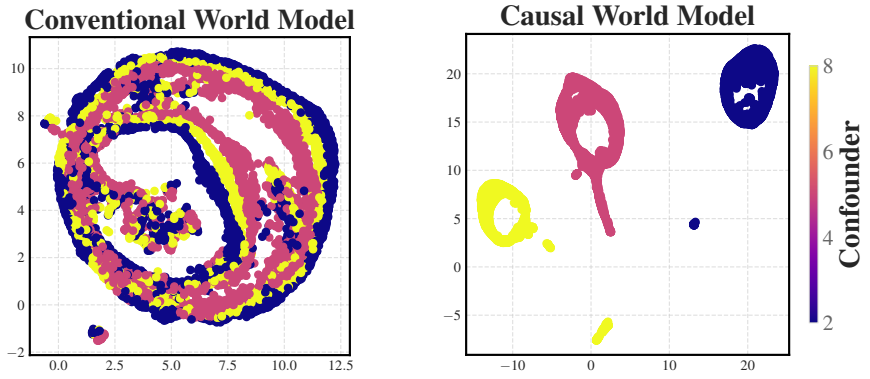


Figure 18: UMAP of the latent states of WMs for DMC *Hopper Hop* task under different *torso mass* values (the darker, the lower).

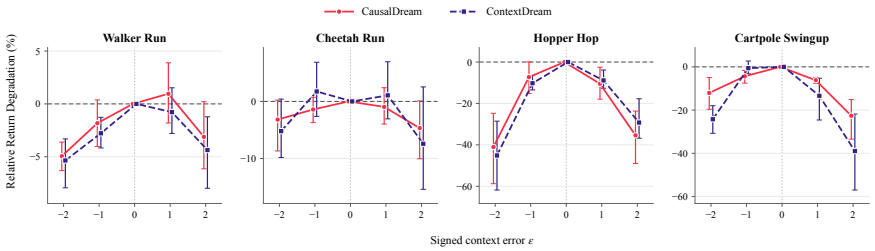


Figure 19: Closed-loop robustness to exogenous-variable estimation errors across four tasks. The vertical axis shows the relative return change from the clean-input setting ($\epsilon = 0$); error bars denote 95% confidence intervals.

not overly dependent on precise exogenous estimates and can retain effective control under moderate context uncertainty.

Table 11: Closed-loop returns under exogenous-variable estimation errors. Each entry reports the absolute return with the relative change from the clean-input setting ($\epsilon = 0$) in parentheses, computed as $100(R_\epsilon/R_0 - 1)\%$ from the displayed mean returns; higher is better and best returns are bolded.

Task	**Method**	**Clean ($\epsilon = 0$)**	**Erroneous Input Return (Degradation %)**			
		Baseline	$\epsilon = -2$	$\epsilon = -1$	$\epsilon = +1$	$\epsilon = +2$
Walker Run	CausalDream	**568.3**	**540.2 (-4.9%)**	**557.9 (-1.8%)**	**573.7 (+1.0%)**	**550.6 (-3.1%)**
	ContextDream	545.1	515.9 (-5.4%)	529.9 (-2.8%)	540.8 (-0.8%)	521.2 (-4.4%)
Cheetah Run	CausalDream	**794.1**	**768.8 (-3.2%)**	**782.9 (-1.4%)**	**786.5 (-1.0%)**	**757.0 (-4.7%)**
	ContextDream	747.2	708.2 (-5.2%)	760.2 (+1.7%)	755.1 (+1.1%)	691.7 (-7.4%)
Hopper Hop	CausalDream	**287.6**	**169.6 (-41.0%)**	**266.7 (-7.3%)**	**257.4 (-10.5%)**	185.8 (-35.4%)
	ContextDream	264.5	145.1 (-45.1%)	237.4 (-10.2%)	241.0 (-8.9%)	**187.2 (-29.2%)**
Cartpole Swingup	CausalDream	**834.2**	**733.6 (-12.1%)**	797.8 (-4.4%)	**782.6 (-6.2%)**	**645.6 (-22.6%)**
	ContextDream	818.6	620.2 (-24.2%)	**813.7 (-0.6%)**	709.2 (-13.4%)	499.7 (-39.0%)

E.10 Additional Counterfactual Action-Effect Results

Table 12 reports the full Return, Reward, and Image effect MSE and candidate-selection regret for all four tasks, extending the Return effect MSE and regret summary in Table 2. The counterfactual evaluation provides a more direct examination of the causal gain of CausalDream. Unlike factual prediction, which can be accurate by exploiting correlations between environmental context and transitions, this evaluation asks whether the learned world model can correctly predict the effect of changing an action while the underlying dynamics condition remains fixed. Accurate prediction under

Table 12: Full counterfactual action-effect metrics on all four tasks ($H = 10$). CausalDream and ContextDream receive ground-truth exogenous parameters. Mean$_{\pm \mathrm{SD}}$ across seeds; bold/underline mark the best/second-best means among the three listed methods. Return/Reward/Image denote effect MSE. All values in 10^{-3}; lower is better.

	Hopper Hop				Walker Run			
Method	Return ↓	Reward ↓	Image ↓	Regret ↓	Return ↓	Reward ↓	Image ↓	Regret ↓
DreamerV3	$\underline{572.2}_{\pm 158}$	$\underline{49.32}_{\pm 14}$	$1.66_{\pm 0.20}$	$286.6_{\pm 125}$	$342.4_{\pm 84}$	$26.79_{\pm 5.3}$	$14.54_{\pm 0.61}$	$340.4_{\pm 34}$
ContextDream	$683.7_{\pm 207}$	$58.08_{\pm 21}$	$\underline{1.65}_{\pm 0.30}$	$\underline{283.6}_{\pm 80}$	$\underline{275.0}_{\pm 53}$	$\underline{22.71}_{\pm 3.7}$	$\mathbf{13.91}_{\pm 0.48}$	$\underline{296.0}_{\pm 8.1}$
CausalDream	$\mathbf{463.1}_{\pm 78}$	$\mathbf{46.91}_{\pm 11}$	$\mathbf{1.62}_{\pm 0.15}$	$\mathbf{220.0}_{\pm 107}$	$\mathbf{254.2}_{\pm 47}$	$\mathbf{22.60}_{\pm 3.7}$	$\underline{14.22}_{\pm 0.48}$	$\mathbf{291.3}_{\pm 21}$

	Cheetah Run				Cartpole Swingup			
Method	Return ↓	Reward ↓	Image ↓	Regret ↓	Return ↓	Reward ↓	Image ↓	Regret ↓
DreamerV3	$\underline{81.5}_{\pm 25}$	$4.35_{\pm 1.4}$	$0.69_{\pm 0.14}$	$\underline{93.5}_{\pm 21}$	$\underline{46.9}_{\pm 45}$	$1.62_{\pm 2.2}$	$0.27_{\pm 0.03}$	$30.8_{\pm 13}$
ContextDream	$82.5_{\pm 35}$	$\underline{4.23}_{\pm 1.4}$	$\underline{0.66}_{\pm 0.15}$	$\mathbf{87.5}_{\pm 31}$	$76.7_{\pm 72}$	$\underline{1.40}_{\pm 1.4}$	$\mathbf{0.24}_{\pm 0.04}$	$\mathbf{28.8}_{\pm 13}$
CausalDream	$\mathbf{79.9}_{\pm 12}$	$\mathbf{4.02}_{\pm 0.69}$	$\mathbf{0.63}_{\pm 0.13}$	$98.9_{\pm 25}$	$\mathbf{30.2}_{\pm 8.9}$	$\mathbf{0.78}_{\pm 0.24}$	$\mathbf{0.24}_{\pm 0.05}$	$\underline{29.2}_{\pm 7.8}$

such controlled interventions therefore provides evidence that the model captures action–outcome relationships beyond contextual conditioning alone.

Evaluation Protocol. We construct paired counterfactual branches from the same recorded simulator state. For each branch, only the first action is replaced by a candidate action, while the physical condition and all subsequent actions are kept unchanged. Simulator rollouts provide the corresponding ground-truth outcomes. The frozen scene bank is collected from CausalDream, DreamerV3, and ContextDream, and all evaluated models, including the *w/o CCR* and *w/o CFI* ablations, are evaluated on the same matched states and candidate actions. This fixed-action evaluation isolates the effect of the initial intervention, whereas CFI training samples subsequent actions from the policy conditioned on each branch's evolving latent state.

For each task and training seed, the shared bank contains 108 scenes, with 36 from each source policy. Each model is initialized from five observation–action pairs. For Walker Run, Cheetah Run, and Hopper Hop, alternative initial actions are generated along three normalized Gaussian random directions, using both signs and perturbation magnitudes of 0.5 and 1.0. Actions are clipped to $[-1, 1]$; duplicates and perturbations with realized Euclidean distance below 0.1 are discarded, leaving at most 12 alternatives. For Cartpole Swingup, candidates are drawn from the nine-point grid $\{-1, -0.75, \ldots, 0.75, 1\}$ and combined with the original action, with duplicates removed. We compute the metrics separately for 10 stochastic latent rollouts and then average them, followed by averaging within each scene source and equally across the three sources.

Let $j = 0$ denote the original action sequence and $j = 1, \ldots, K$ denote the retained counterfactual alternatives in each scene, excluding the original action. We define the finite-horizon return and its intervention effect as

$$G_H^{(j)} = \sum_{k=1}^{H} \gamma^{k-1} r_k^{(j)}, \qquad \Delta G_H^{(j)} = G_H^{(j)} - G_H^{(0)}, \tag{17}$$

where $\gamma = 0.997$.

We evaluate the prediction error of the intervention-induced changes in return, reward, and observation:

$$E_G(H) = \frac{1}{K}\sum_{j=1}^{K}\left(\widehat{\Delta G}_H^{(j)} - \Delta G_H^{(j)}\right)^2, \tag{18}$$

$$E_r(H) = \frac{1}{K}\sum_{j=1}^{K}\left[(\hat{r}_H^{(j)} - \hat{r}_H^{(0)}) - (r_H^{(j)} - r_H^{(0)})\right]^2, \tag{19}$$

$$E_o(H) = \frac{1}{KD}\sum_{j=1}^{K}\left\|(\hat{o}_H^{(j)} - \hat{o}_H^{(0)}) - (o_H^{(j)} - o_H^{(0)})\right\|_2^2, \tag{20}$$

where D denotes the number of image entries. Lower values indicate more accurate prediction of the effect caused by the action intervention.

To assess whether these predictions are useful for action selection, we further report candidate-selection regret:

$$\mathcal{R}_H = \max_{0\le j\le K} G_H^{(j)} - \frac{1}{|\widehat{\mathcal{J}}|}\sum_{j\in\widehat{\mathcal{J}}} G_H^{(j)}, \tag{21}$$

where $\widehat{\mathcal{J}}$ denotes the candidate actions with the highest predicted return. Lower regret indicates a smaller return gap to the best candidate initial action under the shared action suffix. This metric evaluates finite-horizon candidate selection; closed-loop policy performance is assessed by the control experiments. Regret is averaged over all shared evaluation scenes, including those with identical candidate returns, which contribute zero regret for every method. All results are reported at $H = 10$ as mean and standard deviation across training seeds.

Full-Metric Results. Table 12 shows that CausalDream achieves the lowest mean Return and Reward effect MSE on all four tasks. The Reward effect results complement the Return effect gains reported in Section 5.3, showing improved prediction of intervention-induced changes in both immediate rewards and cumulative returns. For Image effect MSE, CausalDream achieves the lowest mean on *Hopper Hop* and *Cheetah Run*, ties ContextDream at the reported precision on *Cartpole Swingup*, and ranks second on *Walker Run*. Candidate-selection regret is lowest on *Hopper Hop* and *Walker Run*, while ContextDream has lower means on *Cheetah Run* and *Cartpole Swingup*, and DreamerV3 also has a lower mean on *Cheetah Run*. With matched exogenous-context access, the consistent Return and Reward effect improvements provide evidence that CCR and CFI strengthen action–outcome modeling beyond contextual conditioning alone, complementing their benefits in the control experiments.

F Theoretical Analysis

We analyze support-level recovery of exogenous variables, information preservation by CCR, and counterfactual rollout consistency under an episode-level structural causal model. All logarithms are natural, and information quantities are measured in nats.

F.1 Assumptions

We consider an episodically non-stationary environment where the transition dynamics are governed by episode-level exogenous factors. Let $u^c \in \mathcal{U}$ denote the exogenous variable for an episode, and let $\tau^{\text{probe}} = \{(o_t, a_t)\}_{t=1}^{L}$ denote the probe trajectory used for estimation. Throughout the paper, s_t denotes the environment state and $x_t \doteq [h_t, z_t]$ denotes the RSSM model state. This notation distinguishes the two state spaces.

Assumption 1 (Structural causal transition)**.** *The environment transition follows a deterministic structural causal mechanism conditioned on the episode-level exogenous variable:*

$$s_{t+1} = f(s_t, a_t, u^c),$$

where u^c remains fixed within an episode and may vary across episodes.

Assumption 2 (Probe identifiability). *Let U^c denote the random episode-level exogenous factor and let $S_0 \sim \rho(\cdot \mid U^c)$ be the probe's initial state. The fixed probe policy and observation process induce a joint distribution of $(U^c, \tau^{\text{probe}})$. For each component U_i^c, there exists a measurable function d_i such that*

$$d_i(\tau^{\text{probe}}) = U_i^c \qquad \textit{almost surely}.$$

Thus, the observed probe trajectory determines the support-level exogenous label under the reset distribution.

Assumption 3 (Exact state abstraction). *For each u^c, fix a measurable state abstraction $e_{u^c} : \mathcal{S} \to \mathcal{X}$. There exists a measurable abstract transition F_{u^c} such that*

$$e_{u^c}(f(s, a, u^c)) = F_{u^c}(e_{u^c}(s), a)$$

for the states and actions considered. Thus, states with the same abstract representation have the same abstract successor under a common action. The rollout analysis uses this fixed abstraction as its latent coordinate system.

Assumption 4 (Realizability). *With $\tau = \tau^{\text{probe}}$, the estimator's output distributions can jointly represent the conditional mean soft-label vectors $\bar{p}_i(\tau) = \mathbb{E}[p_i(U_i^c) \mid \tau]$ defined in Section F.2: there exists θ_0 with $q_{\theta_0,i}(\tau) = \bar{p}_i(\tau)$ almost surely for every component i. The Causal World Model class can represent the abstract transition family $P^*(\cdot \mid x, a, u^c) = \delta_{F_{u^c}(x,a)}$, where δ_v denotes the point mass at v.*

Assumption 5 (Rollout support coverage). *Let $\Omega \subseteq \mathcal{X} \times \mathcal{A} \times \mathcal{U}$ denote the domain of transition comparison. Every state–action–exogenous tuple reached by the factual and counterfactual rollouts considered lies in Ω.*

Assumption 6 (Componentwise estimation). *For a multi-dimensional exogenous variable $u^c = (u_1^c, \ldots, u_N^c)^\top$, the estimator produces one categorical distribution for each component. Its training objective is the sum of the componentwise soft-label cross-entropies.*

F.2 Support-Level Recoverability of Exogenous Variables

For component U_i^c, let $\mathcal{B}_i = \{b_{i,1}, \ldots, b_{i,K_i}\}$ be its finite support of distinct values. For $\sigma > 0$, define the soft-label vector $p_i(u)$ by

$$p_{i,k}(u) = \frac{\exp(-(b_{i,k} - u)^2/(2\sigma^2))}{Z_i(u)}, \qquad Z_i(u) = \sum_{j=1}^{K_i} \exp(-(b_{i,j} - u)^2/(2\sigma^2)).$$

Write $\tau = \tau^{\text{probe}}$ and let $q_{\theta,i}(\tau)$ denote the estimator's predicted probability vector for component i. The population soft-label risk is

$$\mathcal{R}_{\text{est}}(\theta) = \mathbb{E}_{U^c,\tau}\left[-\sum_{i=1}^{N}\sum_{k=1}^{K_i} p_{i,k}(U_i^c) \log q_{\theta,i,k}(\tau)\right], \qquad \bar{p}_i(\tau) = \mathbb{E}[p_i(U_i^c) \mid \tau].$$

The expectation includes the randomized initial state and the resulting probe observations.

Proposition 7 (Support-level recoverability). *Under Assumptions 4 and 6, every global minimizer θ^* of $\mathcal{R}_{\text{est}}$ satisfies $q_{\theta^*,i}(\tau) = \bar{p}_i(\tau)$ almost surely for each component i. Under Assumption 2, this yields $q_{\theta^*,i}(\tau) = p_i(U_i^c)$ and*

$$\hat{u}_i^c = b_{i,k_i^*} = U_i^c, \qquad k_i^* = \arg\max_k q_{\theta^*,i,k}(\tau) = \arg\min_k |b_{i,k} - U_i^c|$$

almost surely.

Proof. Conditioning on τ replaces each target $p_i(U_i^c)$ in the expected cross-entropy by $\bar{p}_i(\tau)$. The cross-entropy identity therefore gives

$$\mathcal{R}_{\text{est}}(\theta) = \sum_{i=1}^{N} \mathbb{E}_\tau[H(\bar{p}_i(\tau)) + D_{\text{KL}}(\bar{p}_i(\tau) \| q_{\theta,i}(\tau))],$$

where H is categorical entropy. The entropy term is independent of θ. Assumption 4 makes all KL terms zero simultaneously attainable. Since each KL divergence is nonnegative, every global minimizer must satisfy $q_{\theta^*,i}(\tau) = \bar{p}_i(\tau)$ almost surely for all i.

Under Assumption 2, $U_i^c = d_i(\tau)$ almost surely. Hence,

$$\bar{p}_i(\tau) = \mathbb{E}[p_i(U_i^c) \mid \tau] = p_i(d_i(\tau)) = p_i(U_i^c) \qquad \text{almost surely.}$$

For $\sigma > 0$, the Gaussian target is maximized at the support point nearest to U_i^c. Because $U_i^c \in \mathcal{B}_i$ and the support values are distinct, this point is uniquely U_i^c itself. Taking the mode of each predicted vector proves the claim. □

Remark 8 (Probe policy and identifiability)**.** *The fixed probe policy and reset distribution jointly determine the trajectory distribution in Assumption 2. Proposition 7 characterizes its population-optimal soft-label predictor, with exact label recovery following from probe identifiability. Figure 7 reports empirical estimation performance on Walker Run and Hopper Hop.*

Remark 9 (Role of factor-aligned supervision)**.** *Proposition 7 relies on the simulator-provided labels to fix the coordinate system of u^c. A generic self-supervised context encoder may still learn a representation sufficient for predicting transitions, potentially up to an invertible or entangled reparameterization of the underlying dynamics factors. Such representations can be effective for adaptation, but do not by themselves establish the component-wise correspondence assumed by our support-level recoverability result. The privileged supervision used by CausalDream therefore serves as an identification anchor for the exogenous variables appearing in the SCM.*

F.3 Information Preservation by the Causal Consistency Regularizer

Let $\tau^{\text{latent}} = \{(h_t, z_t, a_t)\}_{t=1}^T$ be one latent trajectory and let $\hat{\mathbf{u}}^c$ be its episode-level estimated exogenous variable. The CCR objective in Equation 3 reconstructs this variable in symlog coordinates. For finite-support estimates, the following result relates reconstruction error to information retention.

Proposition 10 (CCR encourages exogenous information retention)**.** *Let $\mathcal{B}^c$ be the finite support of $\hat{\mathbf{u}}^c$, with $K = |\mathcal{B}^c| \geq 2$, and let $\delta = \min_{u \neq u' \in \mathcal{B}^c} \|\operatorname{symlog}(u) - \operatorname{symlog}(u')\|_2 > 0$. If $D = \mathbb{E}[\mathcal{L}_{\text{CCR}}]$ and $\epsilon = 4D/\delta^2 \leq 1 - 1/K$, then*

$$I(\tau^{\text{latent}}; \hat{\mathbf{u}}^c) \geq H(\hat{\mathbf{u}}^c) - h_2(\epsilon) - \epsilon \log(K-1), \tag{22}$$

where h_2 is binary entropy. Thus, sufficiently small CCR reconstruction error provides a lower bound on the exogenous information recoverable from the latent trajectory.

Proof. Decode $g_\phi(\tau^{\text{latent}})$ to its nearest point in the transformed support $\operatorname{symlog}(\mathcal{B}^c)$. If the decoded point differs from the true one, the reconstruction error is at least $\delta/2$. Markov's inequality therefore bounds the decoding error probability by $P_e \leq 4D/\delta^2 = \epsilon$. Fano's inequality gives $H(\hat{\mathbf{u}}^c \mid \tau^{\text{latent}}) \leq h_2(P_e) + P_e \log(K-1) \leq h_2(\epsilon) + \epsilon \log(K-1)$ under the stated range for ϵ. Substituting this bound into $I(\tau^{\text{latent}}; \hat{\mathbf{u}}^c) = H(\hat{\mathbf{u}}^c) - H(\hat{\mathbf{u}}^c \mid \tau^{\text{latent}})$ proves the claim. □

The proposition concerns information recoverable from the complete latent trajectory, including actions. For example, consider $h_t = [v_t, \operatorname{symlog}(\hat{\mathbf{u}}^c)]$, where v_t is the subvector used for dynamics and output prediction. Let $v_{t+1} = F(v_t, a_t)$, with F denoting its update function in this construction, and let the prediction heads depend only on v_t. The stored parameter coordinates then permit exact CCR reconstruction even though the predictive subvector evolves independently of the exogenous factor. In the joint objective, CCR supervises exogenous-information retention, while the dynamics and prediction losses supervise transition and output accuracy. Their combined effect is assessed through the predictive evaluations and ablations.

F.4 Model-Consistent Counterfactual Rollout

Transition agreement. We state rollout consistency in terms of transition agreement on Ω. The implemented free-nats terms $\max(1, D_{\text{KL}})$ are constant below the threshold, so transition agreement is specified separately from minimization of the joint training loss.

Theorem 11 (Model-consistent counterfactual rollout)**.** *Under Assumptions 1, 3, and 5, let $\hat{u}^c = u^c$ be the recovered support-level exogenous variable and initialize the model at $x_0 = e_{u^c}(s_0)$. Suppose its transition kernel agrees with the abstract dynamics on Ω:*

$$P_\phi(\cdot \mid x, a, u^c) = P^*(\cdot \mid x, a, u^c) = \delta_{F_{u^c}(x,a)}, \qquad (x, a, u^c) \in \Omega.$$

For any finite intervention sequence $a^{\mathrm{CF}}_{0:H-1}$ satisfying rollout support coverage, define

$$s^{\mathrm{CF}}_0 = s_0, \quad s^{\mathrm{CF}}_{t+1} = f(s^{\mathrm{CF}}_t, a^{\mathrm{CF}}_t, u^c),$$
$$x^{\mathrm{CF}}_0 = x_0, \quad x^{\mathrm{CF}}_{t+1} \sim P_\phi(\cdot \mid x^{\mathrm{CF}}_t, a^{\mathrm{CF}}_t, \hat{u}^c).$$

Then $x^{\mathrm{CF}}_t = e_{u^c}(s^{\mathrm{CF}}_t)$ almost surely for every $t = 0, \ldots, H$. Hence, replacing the action while holding the initial state and u^c fixed produces the abstract image of the corresponding SCM intervention trajectory.

Proof. The initial equality follows from $x_0 = e_{u^c}(s_0)$. Suppose $x^{\mathrm{CF}}_t = e_{u^c}(s^{\mathrm{CF}}_t)$. Rollout support coverage places $(x^{\mathrm{CF}}_t, a^{\mathrm{CF}}_t, u^c)$ in Ω. Kernel agreement and $\hat{u}^c = u^c$ therefore give

$$\begin{aligned} x^{\mathrm{CF}}_{t+1} &= F_{u^c}(x^{\mathrm{CF}}_t, a^{\mathrm{CF}}_t) \\ &= e_{u^c}\big(f(s^{\mathrm{CF}}_t, a^{\mathrm{CF}}_t, u^c)\big) = e_{u^c}(s^{\mathrm{CF}}_{t+1}) \end{aligned}$$

almost surely, where the second equality uses exact state abstraction. Induction establishes the claim for the full horizon. Under Assumption 1, setting each action to a^{CF}_t and retaining u^c gives the stated SCM intervention. The argument also applies to a common policy conditioned on (x_t, u^c): at each step, matched abstract states induce the same action distribution and the same successor distribution. □

F.5 Scope of the Analysis

The rollout result applies to trajectories that remain in Ω and satisfy the stated transition-agreement condition. The random intervention sampler ranges over the action space; its augmentation benefit is evaluated through the policy-learning and intervention-frequency experiments.

The results connect support-level exogenous recovery, information retention, and rollout consistency on the covered domain. Theorem 11 expresses transition agreement in a fixed abstract coordinate system. For an invertible measurable change of coordinates Γ with measurable inverse, setting $e'_{u^c} = \Gamma \circ e_{u^c}$ and $F'_{u^c}(x', a) = \Gamma(F_{u^c}(\Gamma^{-1}(x'), a))$ preserves the abstraction identity and the rollout conclusion after transforming the model kernel accordingly. Thus, the consistency statement concerns the represented dynamics, independently of the choice of latent coordinates. Predictive accuracy and generalization of the trained model are evaluated in the experiments.