COMPOSITIONAL CONTEXT ENCODING FOR OUT-OF-DISTRIBUTION GENERALIZATION IN MODULAR WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models used in model-based reinforcement learning must generalize to physical configurations unseen during training. A key challenge arises in *compositional out-of-distribution* (OOD) settings: the parameters of each mechanism span a wide range, but training varies each mechanism in isolation, whereas test time requires simultaneous extrapolation across all mechanisms. We show that a naive global context encoder fails in this setting because the joint OOD input it encounters at test time was *never seen in any form* during training, even though the parameters of each individual mechanism remain within the training distribution. We propose **Compositional Context Encoding** (**CCE**), which assigns one independent GRU encoder to each causal mechanism. Grounded in the Independence of Causal Mechanisms (ICM) principle, each encoder reads only the transition history of its own state subspace, so the joint OOD test input is automatically decomposed into per-encoder single-sided OOD inputs that the training distribution *does* cover. A lightweight auxiliary loss aligns inferred mechanism parameters with oracle values during training, enabling stable long-horizon rollouts. On a decoupled double-pendulum benchmark with 64 compositional OOD test configurations, CCE achieves **3.5–4.9**$\times$ lower 20-step rollout MSE than a parameter-free monolithic baseline across three random seeds. By contrast, a global-encoder variant performs *worse* than the parameter-free baseline, confirming that the compositional structure is essential.

1 INTRODUCTION

Model-based reinforcement learning (MBRL) promises data efficiency and transferability by learning an explicit model of the world (Sutton, 1991; Ha & Schmidhuber, 2018; Hafner et al., 2019). A world model trained over diverse conditions can in principle generalize to novel environments by adapting its internal representation of the physical parameters governing the dynamics. This capability, often realized through a context encoder that infers hidden parameters from recent observations (Rakelly et al., 2019; Zintgraf et al., 2021), is critical for deploying MBRL agents in the real world, where environment configurations shift continuously.

The compositional OOD challenge. Consider a system of k independent subsystems, each governed by its own physical parameters $\boldsymbol{\phi}_i$. A natural training protocol varies each subsystem over its full parameter range *in isolation*, holding all other subsystems at their default values. At test time the agent must cope with *all* subsystems simultaneously deviating from defaults—a joint OOD scenario exponentially larger than anything a single training trajectory covers. This setting captures realistic deployment: a robot arm in an unfamiliar lab may face unusual link masses, joint friction, and motor characteristics all at once.

Why global context encoders fail. A context encoder that reads a window of full-state transitions $(\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1})_{t=1}^{K}$ sees signals from *all* subsystems concatenated together. Under the joint OOD test distribution, this concatenated input lies outside the training distribution *as a whole*, even though the individual transitions of each subsystem are marginally in-distribution. The encoder therefore faces a genuine input-level distributional shift, leading to unreliable parameter inference and—critically—

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.

compounding errors in multi-step rollouts, where a systematically wrong parameter estimate steers the trajectory toward a physically inconsistent system. We demonstrate empirically that a global encoder trained on single-sided OOD data produces rollout MSE *worse* than a parameter-free baseline (Section 5).

Our approach. The key insight is that the Independence of Causal Mechanisms (ICM) (Schölkopf et al., 2021; Peters et al., 2017) directly implies a compositional structure in the context-encoding problem. Under ICM, the dynamics of mechanism i depend only on its own state subspace $\mathbf{s}_{i,t}$, its own action $a_{i,t}$, and its own parameters $\boldsymbol{\phi}_i$. Therefore, an encoder for $\boldsymbol{\phi}_i$ need read *only* $(\mathbf{s}_{i,t}, a_{i,t}, \mathbf{s}_{i,t+1})_{t=1}^{K}$—the slice of the full context window corresponding to mechanism i. At test time, even when the joint configuration $(\boldsymbol{\phi}_1^{\text{OOD}}, \ldots, \boldsymbol{\phi}_k^{\text{OOD}})$ was never jointly seen in training, each encoder i receives only transitions from subsystem i, whose individual parameter $\boldsymbol{\phi}_i^{\text{OOD}}$ *was* seen during training (single-sided OOD). Joint OOD is thus automatically decomposed into k single-sided OOD problems, each well within the training coverage of its dedicated encoder.

We instantiate this idea as **CCE** (Compositional Context Encoding), combining three components: (i) a *grouped modular world model* in which module i predicts $\mathbf{s}_{i,t+1}$ from $(\mathbf{s}_{i,t}, a_{i,t}, \hat{\boldsymbol{\phi}}_i)$; (ii) k independent GRU encoders, one per module, each reading only its own subspace transitions to produce a latent $\mathbf{z}_i$ from which $\hat{\boldsymbol{\phi}}_i$ is decoded; and (iii) a lightweight auxiliary loss $\|\hat{\boldsymbol{\phi}}_i - \boldsymbol{\phi}_i^*\|^2$ that directly aligns inferred with oracle parameters during training.

Contributions.

- We identify and formalize the *compositional OOD* failure mode of global context encoders in modular world models: it arises whenever training uses single-sided OOD data while evaluation requires joint OOD generalization, a practically common mismatch.
- We propose CCE, a principled encoder architecture grounded in the ICM factorization that automatically decomposes joint OOD generalization into per-mechanism single-sided OOD problems at inference time.
- On a controlled double-pendulum benchmark with known ICM structure, CCE achieves 3.5–4.9$\times$ lower rollout OOD MSE than a parameter-free monolithic baseline across three seeds, and we show that the auxiliary loss is necessary for stable long-horizon rollouts.

2 Background

World models. A world model $p_\theta(\mathbf{s}_{t+1} \mid \mathbf{s}_t, \mathbf{a}_t)$ predicts the next state given the current state and action (Sutton, 1991; Ha & Schmidhuber, 2018). Modern MBRL systems such as Dreamer (Hafner et al., 2019; 2021; 2023) parameterize the world model as a recurrent state-space model (RSSM) and use its imagined rollouts for policy optimization. Generalizing to unseen environments requires conditioning the world model on environment-specific parameters, either provided as oracle inputs or inferred from context.

Context-conditioned world models. A hidden-parameter MDP (Killian et al., 2017) assumes that the transition distribution is parameterized by a latent variable $\boldsymbol{\phi}$ that is fixed within an episode but varies across episodes. Context encoders estimate $\boldsymbol{\phi}$ from a short history of observed transitions and have been successfully applied in meta-RL (Rakelly et al., 2019) and Bayes-adaptive RL (Zintgraf et al., 2021). These approaches use a single encoder over the full state-action space; we show this design fails under compositional OOD and propose a structured alternative.

Independence of Causal Mechanisms. The ICM principle (Schölkopf et al., 2021; Peters et al., 2017) states that the causal mechanisms governing a system are *autonomous*: the conditional distribution of each variable given its causes is independent of those of other mechanisms. In factored dynamical systems, ICM implies that the transition of subsystem i depends only on $(\mathbf{s}_{i,t}, \mathbf{a}_{i,t}, \boldsymbol{\phi}_i)$, with no dependence on the states of other subsystems or on the parameters of other mechanisms. This factorization is the structural foundation of our encoder design.

Compositional generalization. Compositional generalization—generalizing to unseen combinations of individually-seen components—has been studied extensively in language (Lake & Baroni, 2018) and vision. In dynamical systems, modular architectures naturally support compositional generalization by limiting the exposure of each component to its own subsystem (Goyal et al., 2021). Our work extends this principle to the context-encoding stage of world models, showing that encoder structure—not just world-model structure—is decisive for compositional OOD generalization.

3 METHOD

3.1 PROBLEM SETUP

We study a system of $k = 2$ independent subsystems (see Section 4.1 for the concrete instantiation). The full state is $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2) \in \mathbb{R}^{d_s}$, the action is $\mathbf{a} = (a_1, a_2) \in \mathbb{R}^{d_a}$, and the physical parameters are $\boldsymbol{\phi} = (\boldsymbol{\phi}_1, \boldsymbol{\phi}_2)$. By ICM, the transition factorizes as:

$$p(\mathbf{s}_{t+1} \mid \mathbf{s}_t, \mathbf{a}_t, \boldsymbol{\phi}) = \prod_{i=1}^{k} p_i(\mathbf{s}_{i,t+1} \mid \mathbf{s}_{i,t}, a_{i,t}, \boldsymbol{\phi}_i). \tag{1}$$

Training distribution. Parameters $\boldsymbol{\phi}_i$ take values on a finite grid with L levels per factor. Training varies each subsystem over its full parameter grid while holding all others at default values:

$$\mathcal{D}_{\text{train}} = \big\{(\boldsymbol{\phi}_1, \boldsymbol{\phi}_2) : \boldsymbol{\phi}_1 \in \Phi,\ \boldsymbol{\phi}_2 = \boldsymbol{\phi}^{\text{def}}\big\} \cup \big\{(\boldsymbol{\phi}_1, \boldsymbol{\phi}_2) : \boldsymbol{\phi}_1 = \boldsymbol{\phi}^{\text{def}},\ \boldsymbol{\phi}_2 \in \Phi\big\}. \tag{2}$$

This protocol is practical: varying one subsystem at a time is inexpensive, while exhaustive joint enumeration costs exponentially more.

Test distribution. Evaluation uses *joint OOD* configurations where both subsystems simultaneously take high-range parameter values never jointly seen in training: $\mathcal{D}_{\text{test}} = \{(\boldsymbol{\phi}_1, \boldsymbol{\phi}_2) : \boldsymbol{\phi}_1, \boldsymbol{\phi}_2 \in \Phi_{\text{high}}\}$, where $\Phi_{\text{high}} \subset \Phi$ contains only high-level parameter values. Every $(\boldsymbol{\phi}_{1,j}, \boldsymbol{\phi}_{2,j})$ pair in $\mathcal{D}_{\text{test}}$ is absent from $\mathcal{D}_{\text{train}}$, yet each marginal $\boldsymbol{\phi}_{i,j}$ appears in training.

3.2 GROUPED MODULAR WORLD MODEL

We first describe the world model backbone shared by both the oracle baseline and CCE. *Grouped-ModularWM* partitions the state into k non-overlapping subspaces $\mathbf{s} = (\mathbf{s}_1, \ldots, \mathbf{s}_k)$ and instantiates one GRU-based module per subspace. Module i receives input $\mathbf{x}_i = [\mathbf{s}_i,\ a_i,\ \hat{\boldsymbol{\phi}}_i]$ and predicts:

$$\mathbf{h}_i = \text{GRU}_i(\mathbf{x}_i,\ \mathbf{h}_{i,0}), \qquad \hat{\mathbf{s}}_{i,t+1} = \text{MLP}_i^{\text{dec}}(\mathbf{h}_i), \tag{3}$$

where $\mathbf{h}_{i,0} = \mathbf{0}$ and $\hat{\boldsymbol{\phi}}_i \in \mathbb{R}^{d_\phi}$ is the inferred parameter vector for module i. Modules are trained with independent Adam optimizers; no gradients flow across modules. The full prediction is the concatenation $\hat{\mathbf{s}}_{t+1} = (\hat{\mathbf{s}}_{1,t+1}, \ldots, \hat{\mathbf{s}}_{k,t+1})$.

When $\hat{\boldsymbol{\phi}}_i = \boldsymbol{\phi}_i^*$ (oracle), GroupedModularWM provides the performance upper bound for any context-encoding approach.

3.3 COMPOSITIONAL CONTEXT ENCODER

Context window. Given K consecutive transitions, we form the context tensor $\mathcal{C} \in \mathbb{R}^{K \times (2d_s + d_a)}$ whose t-th row ($t = 1, \ldots, K$) is $[\mathbf{s}_{t_0+t},\ \mathbf{a}_{t_0+t},\ \mathbf{s}_{t_0+t+1}]$.

Global encoder (baseline). A global encoder reads the full row $\mathbf{c}_t = \mathcal{C}_t$ (dimension $2d_s + d_a$):

$$\mathbf{z} = \text{GRU}^{\text{global}}(\mathbf{c}_{1:K}), \qquad \hat{\boldsymbol{\phi}} = \text{MLP}^{\text{global}}(\mathbf{z}). \tag{4}$$

At test time each row $\mathbf{c}_t$ contains signals from all subsystems; their joint parameter values $(\boldsymbol{\phi}_1^{\text{test}}, \boldsymbol{\phi}_2^{\text{test}})$ are jointly OOD, and the encoder faces an input-level distributional shift.

Compositional encoder (CCE). Under the ICM factorization (1), encoder i requires only the slice of each context row corresponding to subsystem i:

$$\mathbf{c}_{i,t} = [\mathbf{s}_{i,t_0+t},\ a_{i,t_0+t},\ \mathbf{s}_{i,t_0+t+1}] \ \in\ \mathbb{R}^{2|\mathbf{s}_i| + |a_i|}, \tag{5}$$

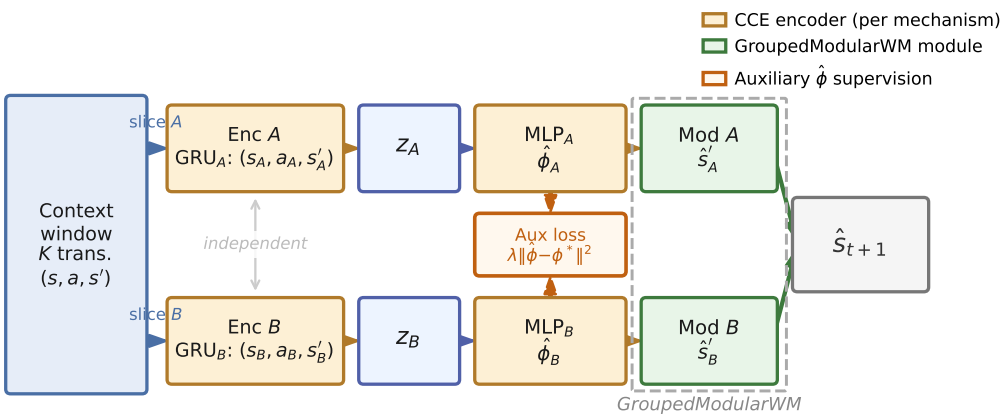


Figure 1: **CCE architecture.** A context window of K transitions is sliced per mechanism (Eq. 5). Encoder i processes only its slice to produce latent $\mathbf{z}_i$, from which an MLP decodes the inferred parameter $\hat{\phi}_i$. Each $\hat{\phi}_i$ conditions module i of GroupedModularWM. Dashed arrows: auxiliary supervision paths along which $\mathcal{L}_{\text{aux}}$ backpropagates into the encoder. All inter-module gradient flows are blocked; each encoder trains against the reconstruction loss of its own module only.

which encodes transitions exclusively from subsystem i. Each independent encoder processes its own slice:

$$\mathbf{z}_i = \text{GRU}_i(\mathbf{c}_{i,1:K}), \qquad \hat{\phi}_i = \text{MLP}_i(\mathbf{z}_i). \tag{6}$$

Because the slice $\mathbf{c}_{i,t}$ is governed entirely by ϕ_i, encoder i at test time sees *exactly the same distribution* as during training with single-sided OOD data on subsystem i. No distributional shift occurs, and k tractable single-sided OOD problems replace one intractable joint OOD problem.

Figure 1 illustrates the full CCE pipeline.

3.4 Training Objective

The primary loss is the sum of per-module next-state prediction errors:

$$\mathcal{L}_{\text{pred}} = \sum_{i=1}^{k} \frac{1}{B \cdot |\mathbf{s}_i|} \left\| \hat{\mathbf{s}}_{i,t+1} - \mathbf{s}_{i,t+1} \right\|^2. \tag{7}$$

An auxiliary loss optionally aligns the inferred parameters with oracle values ϕ_i^* provided at training time:

$$\mathcal{L}_{\text{aux}} = \lambda \sum_{i=1}^{k} \frac{1}{B \cdot d_\phi} \left\| \hat{\phi}_i - \phi_i^* \right\|^2, \tag{8}$$

with $\lambda = 0.1$ in all experiments. The total loss is $\mathcal{L} = \mathcal{L}_{\text{pred}} + \mathcal{L}_{\text{aux}}$.

Why auxiliary loss matters for rollouts. In single-step prediction, the state is reset to ground truth at each window boundary, so small errors in $\hat{\phi}_i$ have limited impact. In multi-step rollouts, the model propagates its own predictions; a systematic bias in $\hat{\phi}_i$ steers the trajectory toward the wrong physical system, and errors compound over the horizon. The auxiliary loss directly penalizes this bias, grounding inferred parameters near oracle values and enabling stable long-horizon rollouts. This is confirmed in Table 1 and the ablation in Section 5.2.

Algorithm 1 CCE Training

Require: Data $\{(\mathcal{C}^{(j)}, \mathbf{s}^{(j)}, \mathbf{a}^{(j)}, \mathbf{s}'^{(j)}, \boldsymbol{\phi}^{*(j)})\}_j$; aux weight λ

for each mini-batch **do**
  $\mathbf{z}_i \leftarrow \mathrm{GRU}_i(\mathcal{C}_{i,1:K})$ for each i *// Eq. (6)*
  $\hat{\boldsymbol{\phi}}_i \leftarrow \mathrm{MLP}_i(\mathbf{z}_i)$ for each i
  $\hat{\mathbf{s}}_{i,t+1} \leftarrow \mathrm{GroupedModularWM}_i(\mathbf{s}_i, a_i, \hat{\boldsymbol{\phi}}_i)$ for each i *// Eq. (3)*
  Compute $\mathcal{L}_{\text{pred}}$, $\mathcal{L}_{\text{aux}}$ *// Eqs. (7), (8)*
  Back-prop $\mathcal{L}_{\text{pred},i} + \mathcal{L}_{\text{aux},i}$ through encoder i only (no cross-module gradients)
end for

3.5 Algorithm Summary

4 Experimental Setup

4.1 Decoupled Double-Pendulum Environment

We use a controlled benchmark environment in which ICM holds by construction. Two pendulums evolve independently with no coupling; pendulum i follows semi-implicit Euler integration:

$$\begin{aligned} \alpha_{i,t} &= a_{i,t}\phi_i^{\text{inv_mass}} - \phi_i^{\text{grav}} \sin\theta_{i,t} - \phi_i^{\text{damp}}\,\omega_{i,t}, \\ \omega_{i,t+1} &= \omega_{i,t} + \Delta t\,\alpha_{i,t} + \varepsilon_i, \\ \theta_{i,t+1} &= \theta_{i,t} + \Delta t\,\omega_{i,t+1}, \end{aligned} \tag{9}$$

with $\Delta t = 0.05$, $\varepsilon_i \sim \mathcal{N}(0, 0.01^2)$. State: $\mathbf{s} = (\theta_A, \omega_A, \theta_B, \omega_B) \in \mathbb{R}^4$. Action: $\mathbf{a} = (a_A, a_B) \in [-1, 1]^2$.

Physical parameters. Each pendulum has three parameters with four discrete levels each:

Parameter	Symbol	Level 0	Level 1 (default)	Level 2	Level 3
Gravity	ϕ^{grav}	1.0	2.5	4.0	6.0
Damping	ϕ^{damp}	0.2	0.5	1.0	1.8
Inverse mass	$\phi^{\text{inv_mass}}$	0.5	1.0	1.5	2.0

OOD split. The full per-pendulum grid has $4^3 = 64$ configurations. Training uses 127 unique joint configurations (single-sided OOD, Eq. 2), covering the full parameter grid of each pendulum while holding the other at its default configuration. Test uses 64 joint configurations where both pendulums take $\{l_2, l_3\}^3$ parameter levels (8 configurations per pendulum, $8 \times 8 = 64$ pairs), none of which appear in training.

4.2 Models Compared

We compare five models:

- **CCE (ours)**: CompositionalContextEncoder + GroupedModularWM, $\lambda = 0.1$, $d_{\text{latent}} = 32$, $K = 5$.
- **CCE-no-aux**: Same architecture but $\lambda = 0$.
- **OracleGrouped**: GroupedModularWM with oracle $\boldsymbol{\phi}^*$ as input; serves as the performance upper bound.
- **MonoPhi**: MonolithicWM with oracle $\boldsymbol{\phi}^*$ appended to the action; tests whether a non-modular backbone benefits from parameter conditioning.
- **MonoNoPhi**: MonolithicWM with no parameter conditioning; serves as the unconditioned lower bound.

Table 1: **OOD generalization results** on the decoupled double-pendulum benchmark (K=5, 20-step rollout). All values are MSE $\times 10^{-2}$ (lower is better). "Ratio" is MonoNoPhi / model rollout MSE (higher means the model is better). Rollout: mean $\pm$ std over three seeds (42, 43, 44). Single-step: mean $\pm$ std over seeds 42 and 44 (single-step metrics were not archived for seed 43; single-step performance is not the primary focus of this work). CCE-no-aux was run on seed 42 only (†).

Model	Single-step OOD MSE ($\times 10^{-2}$)		20-step Rollout OOD MSE ($\times 10^{-2}$)	
	Mean	Std	Mean	Std
OracleGrouped	0.0088	0.0018	0.696	0.337
MonoPhi (oracle ϕ^*)	0.0208	0.0018	1.124	0.330
CCE (ours)	**0.0194**	0.0017	**1.867**	0.327
CCE-no-aux†	0.0210	—	19.597	—
MonoNoPhi	0.1438	0.0001	7.296	0.361

Model	MonoNoPhi / model (rollout)	Direction
OracleGrouped	10.5×	upper bound
MonoPhi	6.5×	oracle-conditioned reference
CCE (ours)	**4.0**×	**inferred $\hat{\phi}$, no oracle at test**
CCE-no-aux†	0.4×	*worse than MonoNoPhi*

† Seed 42 only; not repeated across seeds as the failure was unambiguous (rollout MSE $\approx 2.7\times$ MonoNoPhi).

4.3 Training Details

All models use a GRU hidden size of 64 and an MLP decoder with a single hidden layer of 64 units (Cho et al., 2014). Context window length: $K = 5$ steps. Latent dimension: $d_{\text{latent}} = 32$; parameter dimension $d_\phi = 3$ (matching the oracle dimension when $\lambda > 0$). Training: 120 epochs, batch size 256, learning rate 3×10^{-3} with Adam (Kingma & Ba, 2015). Per training configuration: 20 trajectories of horizon 25; per test configuration: 10 trajectories. All results are averaged over three random seeds (42, 43, 44). Context windows are extracted sequentially from training trajectories (50 800 windows total); rollout evaluation uses a held-out test set collected with seed+2.

Rollout evaluation. We evaluate all models on 20-step rollouts starting from $\mathbf{s}_K$, the state immediately after the context window. The context $\mathcal{C}$ is constructed from the first K steps of each test trajectory and held *fixed* throughout the rollout; the world model predicts autoregressively. Rollout MSE is averaged over the 20-step horizon.

5 Results

5.1 Main Results

Table 1 reports single-step and 20-step rollout OOD MSE for all five models across three seeds. CCE achieves consistent and substantial rollout MSE improvements (Figure 2).

CCE vs. MonoNoPhi. CCE reduces 20-step rollout MSE by a factor of 3.5–4.9× across seeds (Table 2), demonstrating that per-mechanism parameter inference from a short context window substantially improves long-horizon predictive accuracy under compositional OOD.

CCE vs. OracleGrouped. CCE reaches 15–79% of oracle rollout performance across seeds. The wide range is primarily driven by oracle variance itself (oracle rollout MSE spans 0.342–1.149×10^{-2} across seeds), not by CCE instability. The residual gap reflects imperfect alignment of $\hat{\phi}_i$ with ϕ_i^*: the auxiliary loss substantially reduces this gap, but some estimation error remains.

CCE vs. MonoPhi. CCE (which infers ϕ from context at test time) achieves rollout MSE within $1.7\times$ of MonoPhi (which receives ϕ^* as oracle input to a non-modular model). This indicates that the

compositional encoder recovers most of the benefit of oracle parameter knowledge without requiring access to the true parameters at test time.

Table 2: **Per-seed breakdown** for CCE (aux λ=0.1) and CCE-no-aux. Ratio = MonoNoPhi rollout MSE / model rollout MSE.

Seed	CCE ($\times 10^{-2}$)	CCE-no-aux ($\times 10^{-2}$)	MonoNoPhi ($\times 10^{-2}$)	CCE ratio
42	2.250	19.597	7.789	3.46×
43	1.898	—	6.936	3.65×
44	1.452	—	7.163	4.93×
Mean	1.867	—	7.296	**4.01**×

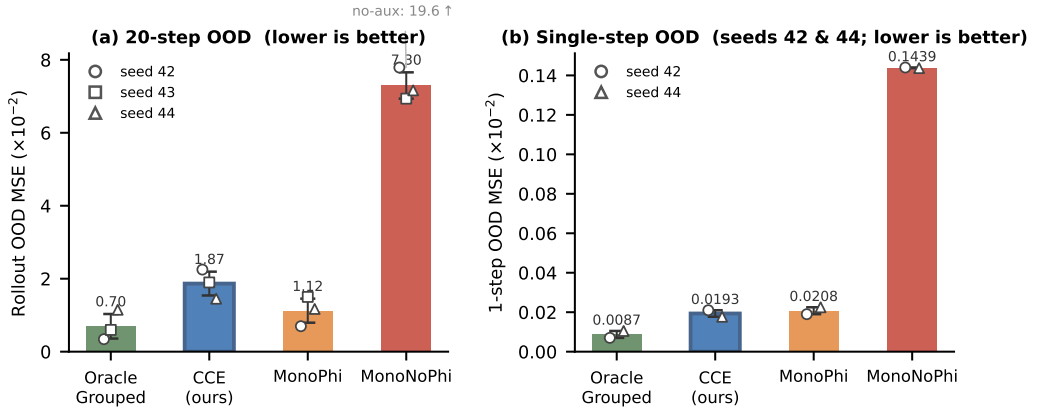


Figure 2: **(a) 20-step rollout OOD MSE** and **(b) single-step OOD MSE** for each model. Bars show means; error bars show population standard deviation (rollout: 3 seeds; single-step: 2 seeds, 42 and 44). Individual-seed results are overlaid as dots. CCE (ours) achieves a 4.0× mean improvement in rollout OOD over MonoNoPhi while remaining competitive on single-step accuracy.

5.2 Ablation: Auxiliary Loss

Table 1 shows a striking difference between CCE and CCE-no-aux: removing the auxiliary loss ($\lambda = 0$) causes rollout MSE to increase $10.5\times$ (from 1.867 to 19.597 $\times 10^{-2}$), making it substantially worse than the parameter-free MonoNoPhi.

Notably, *single-step* OOD MSE is nearly identical for CCE and CCE-no-aux (0.0194 vs. 0.0210 $\times 10^{-2}$ on seeds 42 and 44), indicating that the encoder without auxiliary supervision learns features sufficient for one-step prediction but not for accurate parameter recovery. This dissociation reveals a fundamental limitation of purely prediction-based encoder training: the next-state loss surface may be satisfied by multiple encoder solutions, not all of which produce $\hat{\phi}_i$ close to ϕ_i^*. The auxiliary loss resolves this ambiguity by directly penalizing deviation from the oracle parameter.

Mechanism. In single-step evaluation, the state is reset to ground truth at each window boundary, so a slightly wrong $\hat{\phi}_i$ produces a bounded, independent error per step. In rollouts, the model feeds its own predictions back as input; a biased $\hat{\phi}_i$ systematically steers the trajectory toward a physically different system, and errors grow with the horizon. This explains why the auxiliary loss is critical for rollout quality but not for single-step quality.

5.3 Single-Step vs. Rollout: Diagnostic Value

The results highlight an important lesson for evaluation methodology: *single-step OOD MSE is an unreliable proxy for rollout quality* when context-encoded parameters are held fixed over the horizon.

CCE-no-aux achieves near-oracle single-step accuracy while failing catastrophically in rollout. We recommend that future work on context-conditioned world models report multi-step rollout metrics alongside single-step metrics.

6 Related Work

Context-conditioned model-based RL. PEARL (Rakelly et al., 2019) uses a permutation-invariant encoder over transition tuples to infer a latent task variable for off-policy meta-RL. VariBAD (Zintgraf et al., 2021) performs Bayesian adaptation by maintaining the task posterior as a belief state. Both rely on a single global encoder; we show this design fails under compositional OOD and propose an ICM-structured alternative.

Modular and object-centric world models. Slot Attention (Locatello et al., 2020) learns object-centric representations via competing slots. C-SWM (Kipf et al., 2020) learns structured world models with per-object graph networks. These works focus on *discovering* modularity from data; we instead assume known ICM structure and focus on how to encode context compositionally.

Causal representation learning. Schölkopf et al. (2021) survey causal principles for representation learning, including ICM. Bengio et al. (2013) discuss how modularity and independence aid generalization. Our work operationalizes ICM as a hard structural constraint on encoder architecture rather than a regularization objective.

Compositional generalization. Lake & Baroni (2018) study compositional generalization in sequence models for language. In RL, compositional generalization has been studied through the lens of independent causal mechanisms (Goyal et al., 2021). Our contribution is applying this perspective to the context-encoding stage of world models, demonstrating that encoder architecture—not just world-model architecture—is decisive for compositional OOD generalization.

7 Conclusion

We introduced Compositional Context Encoding (CCE), an encoder architecture for modular world models that achieves compositional OOD generalization by grounding encoder design in the Independence of Causal Mechanisms principle. By assigning one independent encoder per causal mechanism, CCE transforms an intractable joint OOD problem into k tractable single-sided OOD problems. Combined with auxiliary parameter supervision, CCE achieves 3.5–4.9$\times$ lower rollout MSE than a parameter-free monolithic baseline on a decoupled double-pendulum benchmark, without requiring oracle parameters at test time.

Limitations and future work. Our evaluation assumes known modularity (i.e., which state dimensions belong to which module). Learning the module assignment jointly with the world model is an important next step. We also evaluate on a two-module system with discrete parameter levels; scaling to many modules and continuous parameter spaces remains an open challenge. Finally, integrating CCE with a policy-optimization loop (e.g., Dreamer) would test whether rollout improvements translate to downstream task performance.

AI Use Disclosure

AI-assisted writing tools (large language models) were used for grammar checking and phrasing suggestions during manuscript preparation. All technical content, experimental results, code, and scientific conclusions are the sole work of the authors. No AI tools were used to generate experimental data or produce any numerical results reported in this paper.

References

Yoshua Bengio, Aaron Courville, and Pierre Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

Anirudh Goyal, Alex Lamb, Jordan Hoffmann, Shagun Sodhani, Sergey Levine, Yoshua Bengio, and Bernhard Schölkopf. Recurrent independent mechanisms. In *International Conference on Learning Representations (ICLR)*, 2021.

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

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning (ICML)*, 2019.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *International Conference on Learning Representations (ICLR)*, 2021.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Taylor W. Killian, Samuel Daulton, Finale Doshi-Velez, and George Konidaris. Robust and efficient transfer learning with hidden-parameter markov decision processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. In *International Conference on Learning Representations (ICLR)*, 2020.

Brenden M. Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning (ICML)*, 2018.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, 2017.

Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, and Deirdre Quillen. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *International Conference on Machine Learning (ICML)*, 2019.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

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

Luisa Zintgraf, Kyriacos Shi, Maximilian Kurban, Katja Hofmann, and Shimon Whiteson. Varibad: A very good method for bayes-adaptive deep rl via meta-learning. In *International Conference on Learning Representations (ICLR)*, 2021.

A Environment Details

The decoupled double-pendulum environment is implemented in pure NumPy with no external physics libraries. Full source code will be released upon acceptance.

State initialization. At the start of each trajectory, $\theta_i \sim \text{Uniform}(-\pi/2, \pi/2)$ and $\omega_i \sim \text{Uniform}(-1.5, 1.5)$ independently for each pendulum.

Action space. Actions $a_i \in [-1, 1]$ are sampled uniformly at random during data collection.

Data statistics. Training: 127 unique joint configurations $\times$ 20 trajectories each $= 2\,540$ trajectories. Context windows extracted sequentially with window stride 1: $50\,800$ training windows. Test: 64 unique joint OOD configurations $\times$ 10 trajectories each $= 640$ trajectories; $12\,800$ test windows. Rollout evaluation uses 20 trajectories per test configuration, collected with a separate random seed.

B Hyperparameter Details

Table 3: Full hyperparameter list.

Hyperparameter	Value
GRU hidden size (d_h)	64
MLP decoder hidden size	64
Latent dimension (d_{latent})	32
Inferred parameter dimension (d_ϕ)	3
Context window length (K)	5
Batch size	256
Epochs	120
Learning rate	3×10^{-3}
Optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$)
Auxiliary weight (λ)	0.1
Rollout horizon (eval)	20
Trajectory horizon (train)	25
Random seeds	42, 43, 44

C Comparison with Global Context Encoder (ContextAwareGroupedWM)

To confirm that the CCE improvement is attributable to the compositional encoder structure rather than other architectural differences, we also trained a global encoder variant (ContextAwareGroupedWM) that uses a single GRU reading the full context row $[\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1}]$ and feeds a single latent to a shared MLP that produces all $\hat{\boldsymbol{\phi}}_i$ jointly. This model fails similarly to CCE-no-aux, achieving rollout MSE worse than MonoNoPhi (0.40$\times$ on seed 42), consistent with our analysis that the problem is joint OOD distributional shift at the encoder input, not the downstream parameter-conditioned world model.