VECTOR-SYMBOLIC OBJECT REPRESENTATIONS FOR CAUSAL WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Object-centric world models typically factor scenes into continuous latent slots whose representations and dynamics are learned end-to-end. However, unconstrained learned vectors force transition models to implicitly resolve geometric transformations and entity identities, while remaining opaque to inspect or manipulate. In this work, we demonstrate that object-centric world models do not require unconstrained, learned latent spaces to achieve robust multi-body dynamics. We introduce SSP-JEPA, which replaces learned latent slots with fixed vector-symbolic expressions that pairs an entity's identity with an encoding of its position, and predict with a graph network over these newly defined slots. Under this algebraic formulation, metric translation and relative pairwise displacement are exact, closed-form operations, allowing a graph transition network to operate directly over structured geometric states. As a consequence, reading out states, performing counterfactual scene edits, and extracting the model's internal interaction graph require no trained probes or proxy heuristics. Across spring-coupled particles (NRI) and CLEVRER, our fixed-representation model substantially outperforms learned-slot baselines (including C-JEPA and coordinate GNNs). SSP-JEPA reduces rollout error to $0.51\times$ of ballistic motion (versus $0.58\times$ for C-JEPA and $0.68\times$ for coord-GNN) and achieves $0.919_{\pm.005}$ ROC-AUC on physical coupling graph recovery (versus $0.810_{\pm.005}$ for C-JEPA), demonstrating that structured algebraic geometry provides a more effective inductive bias for physical reasoning than unconstrained learned embeddings.

1 INTRODUCTION

World models form internal representations of an environment and its dynamics and use them to plan and reason without acting (Craik, 1943; Sutton, 1991). This idea has been predominantly realized through latent-space predictors in recent world modeling work (Ha & Schmidhuber, 2018; Hafner et al., 2020; Zhou et al., 2024). A growing line of work structures the latent state around individual entities: an encoder factors an observation into per-object slots (Locatello et al., 2020; Kipf et al., 2022; Elsayed et al., 2022), and a predictor evolves each slot (Kipf et al., 2020; Wu et al., 2023; Geusen & Konukoglu, 2026), enabling the latent dynamics to isolate independent object states while explicitly routing interaction effects between them. A recent prominent example of this approach is Causal JEPA (Nam et al., 2026), which learns inter-object relations over temporal slot representations using a per-slot masked-history objective. However, because these slots are unconstrained, learned vectors, the predictor must simultaneously learn the geometry of the physical space alongside the interaction dynamics. Verifying what the dynamics have learned requires trained readout probes that risk conflating model capabilities with probe expressiveness (Belinkov, 2021); furthermore, the dependency structure the predictor relies on is observable only through proxies like attention weights, which have been shown to be insufficient for uncovering true causal dependencies (Geiger et al., 2021).

In our proposed SSP-JEPA model, we replace learned slots with vector-symbolic expressions: an object's identity vector bound to a Spatial Semantic Pointer (SSP) encoding of its position (Komer et al., 2019; Dumont & Eliasmith, 2020). Vector-symbolic representations support algebraic queries and have been used to encode scenes (Penzkofer et al., 2025) and to carry a world model's transitions (Chung et al., 2026; Dumont et al., 2023). We use a graph network over slots to predict multi-object state transitions. Two things follow, with no additional trained parameters. First, a slot is read by

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.

inverting the binding, and a scene, being the sum of its slots, is edited by removing, translating or inserting one; the edited state can be rolled forward and scored against a simulator re-run from the same edit (Section 5.3). Second, the same inversion applied to a pair of slots returns their relative position, and because the predictor is a graph network over object states, the interaction between any two objects can be toggled, which yields the model's interaction graph without training a probe.

Progress on object-centric world models is measured predominantly by prediction accuracy in pixel (Wu et al., 2023) or latent space (Kipf et al., 2020). Downstream physical fidelity is typically evaluated via decoded coordinates or segmentations (e.g. position MSE/mIoU), where the loss conflates representation quality with the capacity of the auxiliary decoder (LeCun, 2022). Furthermore, in benchmark environments like CLEVRER (Yi* et al., 2020) and NRI springs (Kipf et al., 2018), motion over small time steps is overwhelmingly ballistic. An uninformative baseline that copies the previous displacement forward achieves near-optimal single-step error, illustrating how one-step losses suffer from objective mismatch and fail to penalize compounding rollout drift (Lambert et al., 2020). We therefore evaluate in simulated domains with exact ground truth for each of three diagnostic questions asked of every trained model: closed-loop rollout against the observed future, graph recovery against the true coupling graph (which pairs of objects exert force on each other), and counterfactual intervention against a re-simulated scene. Every model is trained with masked-history prediction (Nam et al., 2026), so that an object whose history is hidden can only be predicted from the other objects interacting in the scene, and every model is scored through one shared readout.

We evaluate SSP-JEPA against coordinate-based GNNs and slot-based Transformers across simulated multi-body springs (NRI) and collision dynamics (CLEVRER). On all three diagnostic axes, the vector-symbolic model consistently outperforms the baselines: it cuts long-horizon rollout drift relative to constant velocity, reduces error at non-linear contact frames by nearly half, and recovers unobserved physical coupling graphs without auxiliary probes. Ablation experiments further reveal that these capabilities decouple: structured edge encodings govern relational graph recovery, while phase-space loss constraints preserve rollout stability.

Our contributions are:

1. Proposing SSP-JEPA, an object-centric world model whose slots are vector-symbolic expressions. Reading a slot, editing a scene by deletion, translation or insertion, and forming pairwise relative positions are parameter-free algebraic operations.
2. Proposing a three part diagnostic evaluation for object-centric world models, each question answered exactly by the simulator: closed-loop rollout against the observed future, interaction-graph recovery against the true coupling graph, and intervention against a re-simulation of the edited scene.
3. Demonstrating state-of-the-art results on NRI springs and CLEVRER. The vector-symbolic model leads a coordinate ablation and a slot Transformer on prediction and on graph recovery, and the ablation separates the component that carries graph recovery from the ones that carry rollout stability.

2 Background

2.1 Slot-based prediction and masked training

While JEPA models perform prediction on latent patch representations (Assran et al., 2023; 2025), object-centric dynamics predictors are prone to an inertial shortcut: because physical interactions are temporally sparse, models easily reach low error by predicting self-dynamics and ignoring interactions entirely (Villar-Corrales et al., 2023). While some architectures counter this by separating self-dynamics from relational updates (Villar-Corrales et al., 2023; Kipf et al., 2020), C-JEPA (Nam et al., 2026) instead targets the training objective.

Objects are represented as slots using a Savi and VideSAUR slot encoders (Kipf et al., 2022; Zadaianchuk et al., 2023), and a bidirectional Transformer predictor is trained to predict each object's future slot representation conditioned on the surrounding scene context. During training, a randomly chosen subset of objects has its entire latent history masked, leaving an identity anchor. A masked object's future is then not recoverable from its own trajectory, and the loss can only be reduced by routing information from the other objects, which makes interaction use necessary. We adopt this

objective unchanged and compare against C-JEPA throughout, while modifying the representation that the predictor operates on to give our SSP-JEPA model.

2.2 Vector-symbolic algebras and Spatial Semantic Pointers

A Vector Symbolic Architecture (VSA) represents structured data through algebraic operations over high-dimensional vectors in $\mathbb{R}^D$ (Plate, 2003). We adopt Holographic Reduced Representations (HRRs; Plate, 2003), where bundling (superposition) is vector addition, binding is circular convolution $\circledast$, and similarity is measured via inner products. Binding composes two vectors into a representation quasi-orthogonal to both, while distributing over addition. Each vector a possesses an approximate inverse $a^{-1} = \mathcal{F}^{-1}\{\overline{\mathcal{F}\{a\}}\}$ under binding, allowing bound pairs to be separated algebraically: $(a \circledast b) \circledast a^{-1} \approx b$.

A Spatial Semantic Pointer (SSP) extends this algebra to continuous variables (Komer et al., 2019; Dumont & Eliasmith, 2020). A position $x \in \mathbb{R}^d$ is encoded as

$$\phi(x) = \mathcal{F}^{-1}\left\{e^{\,i\Theta x/\ell}\right\} \in \mathbb{R}^D,$$

where $\Theta \in \mathbb{R}^{D\times d}$ is a fixed conjugate-symmetric matrix of phases and ℓ is a length scale. Conjugate symmetry ensures $\phi(x)$ real. Because every Fourier coefficient has unit modulus, ($|e^{i\theta}| = 1$), $\phi(x)$ is unitary (norm-preserving under circular convolution), which makes its inverse exact: $\phi(x)^{-1} = \mathcal{F}^{-1}\{e^{-i\Theta x/\ell}\}$.

Two core properties govern this representation. First, circular convolution corresponds to continuous translation: $\phi(x) \circledast \phi(\Delta) = \phi(x + \Delta)$, rendering spatial shifts closed-form algebraic operations. Second, by Bochner's theorem and random Fourier feature theory (Rahimi & Recht, 2007; Frady et al., 2022), the inner product between two SSPs defines a shift-invariant kernel over continuous space, $\phi(x) \cdot \phi(y) \approx k(x - y)$; sampling the rows of Θ from an isotropic Gaussian yields an RBF similarity over continuous coordinates (Chouinard, 2026).

3 Method

Figure 1 outlines the pipeline. We first define how scenes and target displacements are represented as vector-symbolic slots (Section 3.1). We then detail the graph network that predicts their transitions (Section 3.2), the masked-history objective used to train it (Section 3.3), and the protocol used to evaluate rollout, graph recovery, and counterfactual interventions (Section3.4).

3.1 Slots and targets

Given an object i at continuous coordinate $x_t^i \in \mathbb{R}^d$, its state is represented by mapping the coordinate to an SSP $\phi(x_t^i)$ and binding it to a fixed, unitary identity vector OBJ_i:

$$z_t^i = \mathrm{OBJ}_i \circledast \phi(x_t^i).$$

Position is recovered directly by unbinding, $\phi(x_t^i) = \mathrm{OBJ}_i^{-1} \circledast z_t^i$, without a trained decoder.

Because object identities and coordinate encodings are fixed, the predictor only models motion. Since binding with an SSP amounts to translation, the target for object i is the encoding of its displacement,

$$d_t^i = \phi(\Delta x_t^i), \qquad \Delta x_t^i = x_{t+1}^i - x_t^i,$$

and the state update is one binding, $z_{t+1}^i = z_t^i \circledast d_t^i$.

3.2 Network architecture

Because the state is already factored into objects and pairwise relationships, a fully connected message-passing graph network over objects is the natural choice of predictor (Figure 1). Hyperparameters are given in Appendix B. The inputs are each object's slot z_t^j and its window of the W most recent displacement encodings.

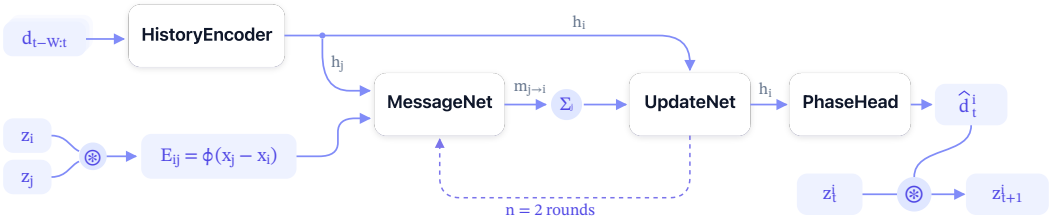


Figure 1: One prediction step. Each object's displacement history is encoded to a node feature h_i by the HistoryEncoder; edge features $E_{ij} = \phi(p_j - p_i)$ are computed from the slots by unbinding; MessageNet produces one message per ordered pair, summed over senders and combined with h_i by UpdateNet for n=2 shared-weight rounds; PhaseHead emits the predicted phases $\widehat{\Delta\Phi}_i$, which define the displacement SSP $\widehat{d}_t^i = \mathcal{F}^{-1}\{e^{i\widehat{\Delta\Phi}_i}\}$

HistoryEncoder. Each object's window of W displacement SSPs is flattened and encoded by an MLP into the node feature h_i, summarizing how that object has been moving.

MessageNet. The influence of a neighbour j on object i is computed from the neighbour's node feature h_j, and the pair's relative position. The relative position is obtained through unbinding identity from each slot, leaving their position encodings, which are then compared:

$$E_{ij} = \left(\mathrm{OBJ}_j^{-1} \circledast z^j\right) \circledast \left(\mathrm{OBJ}_i^{-1} \circledast z^i\right)^{-1} = \phi(x_j - x_i) \ \in \mathbb{R}^D,$$

the encoding of $x_j - x_i$. Because only relative positions enter the network, the predictor is translation-equivariant. The message from j to i is computed from the concatenation of h_j and E_{ij}:

$$m_{j\to i} = \mathrm{MLP}\big([\,h_j \| E_{ij}\,]\big) \ \in \mathbb{R}^H,$$

one per ordered pair, and the messages arriving at i are summed, $a_i = \sum_{j\neq i} m_{j\to i}$.

UpdateNet. Object i's node feature is updated from its own value and its aggregated incoming messages,

$$h_i \leftarrow \mathrm{MLP}\big([\,h_i \| a_i\,]\big),$$

so that h_i now summarizes both how i has been moving and what its neighbours contribute. Messages and updates are applied for R rounds with shared weights, the updated node features feeding the next round's messages, and the head reads the final h_i.

PhaseHead. A displacement encoding is unitary and is therefore determined by its Fourier phases (which are linear in the displacement itself),

$$\Delta\Phi = \Theta\,\Delta x/\ell \ \in \mathbb{R}^D,$$

with phase matrix Θ and length scale ℓ defined in Section 2. The head regresses these D phases from h_i, emitting $\widehat{\Delta\Phi}_i$. Advancing the state corresponds to multiplying each Fourier coefficient of slot z_t^i by the corresponding predicted phase factor:

$$\mathcal{F}\{z_{t+1}^i\}_k = \mathcal{F}\{z_t^i\}_k\, e^{\,i\widehat{\Delta\Phi}_{i,k}}.$$

This operation is exact circular convolution with the predicted displacement SSP $\widehat{d}_t^i = \mathcal{F}^{-1} e^{i\widehat{\Delta\Phi}_i}$, keeping the updated slot unitary. As detailed in Section 3.3, the network is trained end-to-end to align these predicted phases with target phase shifts. For evaluation, metric displacements are recovered via least squares: $\widehat{\Delta x}_i = \ell\,\Theta^+\widehat{\Delta\Phi}_i$, where Θ^+ is the pseudoinverse of Θ.

3.3 Training Objective

Models are trained with a masked-history objective adapted from C-JEPA (Nam et al., 2026), augmented with noise regularization to stabilize rollouts (Sanchez-Gonzalez et al., 2020). With probability p, up to $m_{\max}$ objects have their displacement history replaced by their identity token tiled over the window. This removes an entity's historical trajectory while preserving its identity anchor,

forcing its future motion to be inferred strictly through incoming relational messages. The overall training objective is $\mathcal{L} = \mathcal{L}_{\text{future}} + \mathcal{L}_{\text{history}} + \lambda \mathcal{L}_{\text{noise}}$, evaluated over predicted phases:

$$\mathcal{L}_{\text{f}} = \frac{1}{N}\sum_{i=1}^{N} \big\|\widehat{\Delta\Phi}_i - \Delta\Phi_i\big\|^2, \quad \mathcal{L}_{\text{h}} = \frac{1}{|M|}\sum_{i\in M} \big\|\widehat{\Delta\Phi}_i - \Delta\Phi_i\big\|^2, \quad \mathcal{L}_{\text{n}} = \frac{1}{N}\sum_{i=1}^{N} \big\|\widehat{\Delta\Phi}_i - \Delta\Phi_i^{\star}\big\|^2, \tag{1}$$

where $\Delta\Phi_i^{\star} = \Delta\Phi_i - \Theta\epsilon_i/\ell$ preserves the ground-truth destination under spatial perturbation $\epsilon_i \sim \mathcal{N}(0, \sigma^2 I_2)$. This corresponds to translating each input slot by ϵ, i.e., binding it with the perturbation SSP $\phi(\epsilon)$ (Sanchez-Gonzalez et al., 2020).

3.4 Measurements

We evaluate models across three diagnostic axes: closed-loop rollout prediction, interaction-graph recovery, and counterfactual intervention. All predictions $\widehat{\Delta x}_i$ and targets Δx_i are evaluated as metric displacements in continuous space, scoring every model through an identical readout. We write $\text{sim}(a, b) = \frac{a\cdot b}{\|a\|\,\|b\|}$ for cosine similarity.

Prediction. We evaluate closed-loop rollout stability by providing the model with a seed history of W true frames, after which it autoregressively consumes its own predictions. The score at horizon h is the mean position error normalized against an uninformative constant-velocity (CV) baseline: $\rho(h) = \bar{e}(h)/\bar{e}_{\text{CV}}(h)$, where $\rho < 1$ indicates that the model outperforms ballistic inertia.

Graph recovery. Physical multi-body environments possess an underlying ground-truth interaction graph $e \in \{0, 1\}^{N\times N}$, where $e_{ji} = 1$ indicates an active dynamic coupling between entity j and entity i. To determine whether a model's internal representations uncover this relational structure without relying on external probes, we perturb edge communications during single-step inference. For a target receiver i with masked history, we compare its predicted displacement with all messages intact ($\widehat{\Delta x}_{i,t}$) against its prediction when the directed message from j is ablated ($\widehat{\Delta x}_{i,t}^{\setminus j}$). The magnitude of the resulting shift reflects the model's reliance on sender j:

$$S_{ij} = \frac{1}{T}\sum_{t=1}^{T} \big\|\widehat{\Delta x}_{i,t}^{\setminus j} - \widehat{\Delta x}_{i,t}\big\|.$$

We evaluate S_{ij} as a binary classifier of true physical edges via the Area Under the ROC Curve (ROC-AUC). Baseline comparisons include chance (0.50) and a proximity baseline that ranks edges purely by spatial distance.

Intervention. We evaluate counterfactual consistency by removing an entity j mid-trajectory. At intervention step $t^{\star}$, the simulator branches by integrating physics without entity j, while the model branches by zeroing j's outgoing messages for n subsequent rollout steps. Letting $(\text{pred}, \text{pred}_{\setminus j})$ denote the model's factual and edited rollouts, and $(\text{fact}, \text{fact}_{\setminus j})$ denote the simulator's corresponding trajectories, the counterfactual effect is the difference induced by the removal. We compute counterfactual alignment as:

$$A_j = \text{sim}\big(\text{pred}_{\setminus j} - \text{pred},\ \ \text{fact}_{\setminus j} - \text{fact}\big) \in [-1, 1].$$

Subtracting factual trajectories eliminates common baseline rollout drift, isolating whether the model correctly routes causal interactions.

Choice of horizon. Graph recovery is evaluated over single steps: ablated messages are only causally interpretable when conditioned on uncorrupted physical states, whereas closed-loop error accumulation would confound the attribution. In contrast, prediction and intervention are multi-step: constant velocity can only be outperformed once initial velocity becomes stale, and removing an object exerts downstream effects only as neighboring entities evolve under altered forces.

4 Experimental setup

We evaluate on two simulated domains, chosen so that each measurement of Section 3.4 has a ground-truth answer. Interaction-graph recovery and intervention require a known coupling graph

Table 1: Diagnostic evaluation on NRI springs (mean ± standard error over five seeds; 100 test episodes for recovery, 40 for rollout and intervention). Rollout error ρ is normalized to constant velocity ($h = 20$). *NRI is an upper-bound reference trained with trajectory-level edge supervision.

	Rollout Error ρ (↓)		Graph Recovery (↑)	Intervention (↑)
	$h = 10$	$h = 20$	ROC-AUC	Alignment A_j
NRI (reference)*	$0.23_{\pm.01}$	$0.14_{\pm.01}$	$0.995_{\pm.000}$	$0.991_{\pm.002}$
coord-GNN	$0.61_{\pm.02}$	$0.64_{\pm.03}$	$0.866_{\pm.003}$	$0.281_{\pm.023}$
C-JEPA	$0.77_{\pm.02}$	$0.65_{\pm.02}$	$0.810_{\pm.005}$	$0.588_{\pm.020}$
SSP-JEPA	$\mathbf{0.55}_{\pm.02}$	$\mathbf{0.51}_{\pm.02}$	$\mathbf{0.919}_{\pm.005}$	$\mathbf{0.638}_{\pm.019}$

and a re-runnable simulator, and are measured on NRI springs (Kipf et al., 2018): $N = 5$ particles, each pair independently coupled by a zero-rest-length Hooke spring with a probability of one half, $F_{ij} = -k\, e_{ij}\, (x_i - x_j)$. Because the coupling is sampled independently of the particles' positions, neither task can be solved from proximity alone.

Prediction is measured on both NRI springs and CLEVRER (Yi* et al., 2020), a dataset of rendered videos of colliding objects with per-frame trajectories and collision annotations; CLEVRER contributes the contact-resolved analysis, which springs cannot support because its interactions are continuous rather than discrete event-like. Generator settings, dataset sizes, and preprocessing are given in Appendix A.

We compare three models and use a fourth to establish a performance upper bound, all consuming object states. **SSP-JEPA** is the model of Section 3.2. **coord-GNN** removes the vector-symbolic representation from every part of the model: raw two-dimensional positions in place of slots, a displacement history in coordinates, the raw relative position as the edge feature, and a two-dimensional displacement target, with identity supplied by a learned per-object embedding. It shares SSP-JEPA's architecture and training, and isolates the representation.

C-JEPA (Nam et al., 2026) is a slot Transformer trained with the same masked objective. To isolate the predictive dynamics from visual representation learning and ensure a fair comparison across all models, we adapt C-JEPA to consume ground-truth object states rather than pixel video features. This introduces two modifications. First, we replace the latent-space target with a coordinate displacement head, aligning its training objective with the shared evaluation space. Second, while original slots extract appearance features that naturally bind identity, state-based positions lack visual keys; we therefore provide the same learnable identity embeddings used by **coord-GNN** so attention can reliably distinguish overlapping or co-located entities.

Unlike the previous models, **NRI** (Kipf et al., 2018) infers an explicit latent edge variable for each pair, with an encoder that reads a whole trajectory and a decoder that predicts dynamics conditioned on the sampled graph. Unlike the other three models, graph recovery is its training objective, so we use it explicitly as the ceiling for the task, read from its own edge posterior. Because the coupling graph required by this model does not exist for CLEVRER, we do not use it for that task.

All models are trained with five seeds on a single NVIDIA RTX 5090. Compute and Software Environment, optimization settings, per-architecture tuning, and run audits are in Appendix B.

5 Results

We evaluate SSP-JEPA against coordinate-based and slot-based baselines across all three diagnostic axes: closed-loop rollout stability, interaction-graph recovery, and counterfactual scene interventions (Table 1). Across both NRI springs and CLEVRER multi-body dynamics, SSP-JEPA outperforms all non-oracle baselines: it achieves a cumulative rollout error ratio relative to constant velocity of $\rho = 0.51$ on springs and 0.47 on CLEVRER, lowers single-step error at contact frames to $\rho = 0.46$ compared to 0.75 for C-JEPA, and recovers coupling graphs at 0.919 ROC-AUC without auxiliary readout probes.

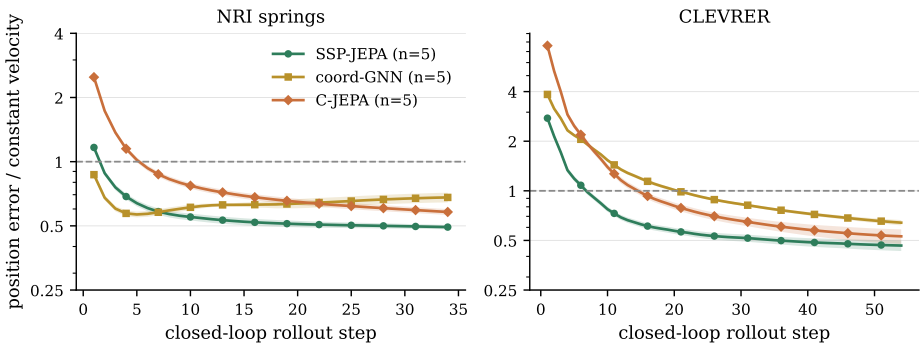


Figure 2: Closed-loop error ratio $\rho(h)$ across rollout steps on NRI springs (left) and CLEVRER (right); mean $\pm$ standard error over five training seeds across 40 held-out episodes. Dashed line at $\rho = 1$ indicates equivalence to constant velocity.

5.1 Prediction

Closed-loop rollout. We quantify multi-step dynamics using the error ratio $\rho(h) = \bar{e}(h)/\bar{e}_{\mathrm{CV}}(h)$, where $\bar{e}(h)$ is the mean Euclidean position error at horizon h and $\bar{e}_{\mathrm{CV}}(h)$ is the constant-velocity baseline. Because constant velocity simply repeats its initial displacement while true velocities continuously change, its accumulated error grows over time and $\rho(h)$ falls as a model learns physical acceleration (Figure 2). On NRI springs, SSP-JEPA outperforms constant velocity ($\rho < 1$) starting at step $h = 2$, reaching $\rho = 0.51_{\pm.02}$ at $h = 20$, compared to $0.64_{\pm.03}$ for coord-GNN and $0.65_{\pm.02}$ for C-JEPA (Table 1). On CLEVRER, SSP-JEPA crosses the CV baseline by $h = 7$ and achieves $\rho = 0.47_{\pm.03}$ at horizon $h = 20$, outperforming both coord-GNN ($0.64_{\pm.02}$) and C-JEPA ($0.53_{\pm.05}$).

Dynamics at contact. Collisions account for only 2.0% of object-steps in CLEVRER, yet represent the regime where inertial extrapolation fails: constant-velocity error at contact frames is $46\times$ higher than on ballistic frames. Evaluated at the collision step conditioned on true history, SSP-JEPA achieves $\rho = 0.46_{\pm.01}$, significantly outperforming C-JEPA ($0.75_{\pm.01}$) and coord-GNN ($0.94_{\pm.01}$). This demonstrates that the vector-symbolic state representation models non-linear impulse dynamics rather than merely fitting smooth, low-acceleration drifts.

5.2 Interaction-graph recovery

SSP-JEPA recovers the unobserved physical coupling graph with an ROC-AUC of $0.919_{\pm.005}$, significantly surpassing coord-GNN ($0.866_{\pm.003}$) and C-JEPA ($0.810_{\pm.005}$), with every SSP-JEPA seed outperforming every baseline seed (Table 1). Notably, all models are trained solely via predictive self-supervision without receiving supervision on graph connectivity. Because springs attract coupled particles, spatial proximity alone provides a baseline recovery score of 0.679; SSP-JEPA's substantial margin over both the proximity floor and coordinate baselines confirms that its relational messages capture causal mechanical interactions rather than geometric co-location.

5.3 Intervention

To test whether models capture causal mechanisms rather than surface correlations, we evaluate counterfactual predictions by removing an object mid-episode (Section 3.4). The model continues its rollout with the removed object's outgoing messages zeroed, while the simulator integrates the ground-truth physics in its absence. SSP-JEPA achieves a counterfactual alignment of $0.638_{\pm.019}$, outperforming C-JEPA ($0.588_{\pm.020}$) and coord-GNN ($0.281_{\pm.023}$; Table 1).

5.4 Ablations

To isolate the contribution of each architectural and training choice, we perform leave-one-out ablations on SSP-JEPA by substituting vector-symbolic representations with cartesian coordinates or modifying training objectives (Table 2). The *Edge representation* ablation replaces the SSP edge en-

Table 2: Leave-one-out ablations on NRI springs (mean ± standard error over five seeds). Rows substitute vector-symbolic components with Cartesian equivalents or remove loss terms. Rollout error ρ is reported at horizon $h = 20$.

	Graph recovery ROC-AUC (↑)	Rollout error, $h = 20$ ρ (↓)	Intervention Alignment A_j (↑)
SSP-JEPA	$\mathbf{0.919}_{\pm.005}$	$\mathbf{0.51}_{\pm.02}$	$0.638_{\pm.019}$
→ Cartesian edges	$0.886_{\pm.005}$	$0.55_{\pm.01}$	$0.597_{\pm.030}$
→ Cartesian history	$0.919_{\pm.005}$	$0.55_{\pm.02}$	$\mathbf{0.693}_{\pm.034}$
→ Metric displacement loss	$0.927_{\pm.003}$	$12.2_{\pm0.3}$	$-0.010_{\pm.009}$
$-\mathcal{L}_{\text{noise}}$	$0.914_{\pm.006}$	$4.95_{\pm0.19}$	$0.088_{\pm.016}$
coord-GNN	$0.866_{\pm.003}$	$0.64_{\pm.03}$	$0.281_{\pm.023}$

coding $E_{ij} = \phi(x_j - x_i)$ with raw two-dimensional relative displacements $x_j - x_i$. In the *history representation* ablation, we replace the window of displacement encodings with raw two-dimensional displacements. The *metric loss* ablation retains the D-dimensional phase head but computes mean-squared error on the decoded displacement, $\|\ell\,\Theta^{+}\widehat{\Delta\Phi}_i - \Delta x_i\|^2$, rather than directly on the phases. Finally, we remove $\mathcal{L}_{\text{noise}}$ from the training objective in the *no noise* ablation.

The three evaluation axes decouple across different design choices. Graph recovery degrades only when the edge representation is substituted with Cartesian coordinates ($\Delta = -0.033$), confirming that vector-symbolic edge encodings provide the structured relational inductive bias necessary to isolate forces. Conversely, substituting history encodings leaves graph recovery unchanged.

Rollout stability and counterfactual intervention, by contrast, depend entirely on the training loss and regularizer. Substituting the direct phase loss with a decoded-displacement loss causes rollout error to surge to $12\times$ constant velocity and reduces intervention alignment to zero. This failure occurs because the gradient of the metric loss with respect to $\widehat{\Delta\Phi}_i$ is constrained to the two-dimensional column space of Θ; the remaining $D-2$ phase dimensions receive zero gradient and undergo unconstrained drift during autoregressive rollout. Dropping $\mathcal{L}_{\text{noise}}$ similarly degrades closed-loop rollout ($4.95\times$ CV), confirming that explicit contractive noise injection is indispensable for stabilizing long-horizon trajectory integration.

6 Related work

Object-centric world models. Slot-based predictors factor visual scenes into per-entity representations and roll them forward using recurrent transitions or temporal attention (Kipf et al., 2020; Veerapaneni et al., 2019; Wu et al., 2023). To prevent slot dynamics from collapsing into independent single-object trajectories, subsequent architectures incorporate explicit relational inductive biases (Villar-Corrales et al., 2023) or masked-prediction objectives (Nam et al., 2026). In these models, relational dependencies are either inferred through learned attention maps or routed through latent interaction modules. We build on the masked-history paradigm of Nam et al. (2026), but operate over fixed algebraic slots where relational messages can be ablated directly at inference.

Relational inference and physical simulators. Learning physical dynamics over interacting systems typically follows two paradigms. Graph network simulators assume the true interaction topology is known in advance and route message passing over ground-truth edges (Battaglia et al., 2016; Sanchez-Gonzalez et al., 2020). Conversely, Neural Relational Inference (NRI) and its extensions (Kipf et al., 2018; Webb et al., 2019; Löwe et al., 2020) explicitly infer dynamic graphs by training a variational encoder to predict latent categorical edge variables over complete trajectories. In contrast, SSP-JEPA maintains no explicit edge variables and receives no supervision on scene topology. The interaction graph is instead uncovered by evaluating how individual message pathways govern receiver predictions.

Dynamics in VSAs. Vector Symbolic Architectures (VSAs) and continuous Spatial Semantic Pointers (SSPs) represent structured and continuous variables through algebraic operations in fixed-width vector spaces (Plate, 2003; Komer et al., 2019; Frady et al., 2021). In navigation and spatial map-

ping, SSP transitions have modeled single-agent kinematics through continuous path integration and SLAM, where an agent's spatial pointer is updated by self-motion odometry (Dumont & Eliasmith, 2020; Dumont et al., 2023). More recently, Zhang et al. (2026) formalize a hippocampal-entorhinal-inspired world model that factorizes identity and space by binding non-spatial items to continuous phase-coded grid cell representations, driving transitions via the agent's velocity. Similarly, Chung et al. (2026) model transitions in discrete gridworlds by binding state and action codes. However, these models apply transitions to isolated entities driven by given actions or velocities. We instead model multi-body physical systems, where transitions must be inferred from relational forces and contact events between interacting objects.

Evaluating world models beyond prediction. Standard world model benchmarks evaluate predictive fidelity through downstream task performance, such as reinforcement learning rewards or Visual Question Answering accuracy (Chen et al., 2022; Bear et al., 2021). However, predictive accuracy can correlate poorly with physical consistency (Lambert et al., 2020), and latent world models often exhibit distorted counterfactual responses when subjected to scene interventions (Vakalis, 2026). Paired counterfactual evaluations exist for static visual attributes (Karnam & Sathish, 2026), but assessing dynamic causal structure in learned world models remains an open challenge. Our diagnostic protocol provides a direct, probe-free measurement of rollout stability, relational structure, and counterfactual alignment scored against ground-truth physical simulations.

7 Limitations and future work

Perception is assumed. Every model presented in this work consumes low-dimensional object states. Consequently, our findings evaluate predictive dynamics in isolation rather than end-to-end visual parsing. Deploying SSP-JEPA on video observations requires an upstream perceptual front-end that factors raw scenes into invariant entity identities and continuous coordinates, an inductive bias existing in invariant slot-attention encoders (Biza et al., 2023; Huang et al., 2026). Regressing an SSP state from such an encoder would carry these measurements to video input, and we leave it to future work.

Point states. The predictor maps a history to a single displacement, so the state it carries through a rollout is a point estimate. Planning under partial observation needs a distribution over futures. A key advantage of Vector Symbolic Architectures is that vector superposition (bundling) naturally represents probability distributions over continuous domains without changing dimensionality (Furlong & Eliasmith, 2023). Extending the prediction head to output a superposition of displacement encodings would allow the architecture to propagate full belief distributions over future states using the same phase-advancement update rule. We leave this to future work as well.

8 Conclusion

We introduced SSP-JEPA, an object-centric world model that replaces unconstrained learned latent slots with fixed vector-symbolic expressions. By binding an entity's invariant identity to a Spatial Semantic Pointer of its position, translation and pairwise relative displacements become closed-form algebraic operations instead of learned unconstrained transformations. This structure resolves a core challenge in world modeling: inspecting internal representations, extracting dynamic coupling graphs, and executing counterfactual scene edits can be performed directly without trained readout probes or proxy heuristics.

Empirically, our results on NRI springs and CLEVRER demonstrate that transition models do not need to learn continuous spatial representations from scratch to achieve robust dynamics. Guided by an exact phase-space objective, SSP-JEPA substantially improves closed-loop rollout stability, lowers predictive error at non-linear contact frames, and recovers unobserved physical interaction graphs with higher fidelity than both a coordinate GNN and C-JEPA. Our ablations show that these capabilities decouple cleanly: structured edge representations carry relational graph recovery, while phase-space loss constraints preserve long-horizon manifold stability. Together, these findings demonstrate that embedding known physical geometries directly into the algebraic structure of state representations provides a transparent, verifiable, and empirically superior inductive bias for physical reasoning.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15619–15629. IEEE, 2023.

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, Nicolas Ballas, Fair at Meta, Mila Québec, AI Institute, and Polytechnique Montr'eal. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *ArXiv*, abs/2506.09985, 2025. URL https://api.semanticscholar.org/CorpusID:279306055.

Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics. In *NeurIPS*, 2016.

Daniel M. Bear, Elias Wang, Damian Mrowca, Felix J. Binder, H Tung, R. T. Pramod, Cameron Holdaway, Sirui Tao, Kevin A. Smith, Li Fei-Fei, Nancy Kanwisher, Joshua B. Tenenbaum, Daniel Yamins, and Judith E. Fan. Physion: Evaluating physical prediction from vision in humans and machines. In *NeurIPS Datasets and Benchmarks*, 2021.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48:207–219, 2021. URL https://api.semanticscholar.org/CorpusID:236924832.

Ondrej Biza, Sjoerd Van Steenkiste, Mehdi SM Sajjadi, Gamaleldin F Elsayed, Aravindh Mahendran, and Thomas Kipf. Invariant slot attention: Object discovery with slot-centric reference frames. In *ICML*, 2023.

Zhenfang Chen, Kexin Yi, Yunzhu Li, Mingyu Ding, Antonio Torralba, Joshua B. Tenenbaum, and Chuang Gan. Comphy: Compositional physical reasoning of objects and events from videos. In *ICLR*, 2022.

Jakeb Chouinard. Neurally-plausible radial basis kernels using distributed fourier embeddings. *ArXiv*, abs/2605.08458, 2026. URL https://api.semanticscholar.org/CorpusID:288254050.

William Youngwoo Chung, Calvin Yeung, Hansen Jin Lillemark, Zhuowen Zou, Xiangjian Liu, and Mohsen Imani. Geometric priors for generalizable world models via vector symbolic architecture. *arXiv preprint arXiv:2602.21467*, 2026.

Kenneth J. W. Craik. *The Nature of Explanation*. Cambridge University Press, 1943.

Nicole Sandra-Yaffa Dumont and Chris Eliasmith. Accurate representation for spatial cognition using grid cells. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 42, 2020.

Nicole Sandra-Yaffa Dumont, P. Michael Furlong, Jeff Orchard, and Chris Eliasmith. Exploiting semantic information in a spiking neural SLAM system. *Frontiers in Neuroscience*, 2023. SSP-SLAM; velocity-controlled oscillator update.

Gamaleldin Fathy Elsayed, Aravindh Mahendran, Sjoerd van Steenkiste, Klaus Greff, Michael Curtis Mozer, and Thomas Kipf. Savi++: Towards end-to-end object-centric learning from real-world videos. *ArXiv*, abs/2206.07764, 2022. URL https://api.semanticscholar.org/CorpusID:249712218.

E Paxon Frady, Denis Kleyko, Christopher J Kymn, Bruno A Olshausen, and Friedrich T Sommer. Computing on functions using randomized vector representations (in brief). In *Proceedings of the 2022 Annual Neuro-Inspired Computational Elements Conference*, pp. 115–122, 2022.

E. Paxon Frady et al. Edward paxon frady and denis kleyko and christopher j. kymn and bruno a. olshausen and friedrich t. sommer. *Proceedings of the 2022 Annual Neuro-Inspired Computational Elements Conference*, 2021.

P. Michael Furlong and Chris Eliasmith. Modelling neural probabilistic computation using vector symbolic architectures. *Cognitive Neurodynamics*, 18:1–24, 2023. doi: 10.1007/s11571-023-10031-7.

Atticus Geiger, Hanson Lu, Thomas F. Icard, and Christopher Potts. Causal abstractions of neural networks. In *Neural Information Processing Systems*, 2021. URL https://api.semanticscholar.org/CorpusID:235358214.

Jakob Geusen and Ender Konukoglu. Object-centric lejepa, 2026. URL https://arxiv.org/abs/2607.02404.

David Ha and Jürgen Schmidhuber. World models. 2018. doi: 10.5281/ZENODO.1207631. URL https://zenodo.org/record/1207631.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. 2020. URL https://arxiv.org/abs/1912.01603.

Haoxiang Huang, Zhekai Wang, Xiang Liu, Sen Cui, and Changshui Zhang. Geoco-savi: Geometry-consistent slot attention for explicitly editable object representations, 2026. URL https://arxiv.org/abs/2609.06628.

Anuraag Gadehothur Karnam and Tarunesh Sathish. EditCLEVR: A paired-scene intervention benchmark for compositional faithfulness of object-centric representations. In *ICML 2026 Workshop on Combining Theory and Benchmarks*, 2026. arXiv:2607.22705.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International conference on machine learning*, pp. 2688–2697. Pmlr, 2018.

Thomas Kipf, Elise van der Pol, and Max Max Welling. Contrastive learning of structured world models. In *ICLR*, 2020.

Thomas Kipf, Gamaleldin F. Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. In *ICLR*, 2022. SAVi.

Brent Komer, Terrence C. Stewart, Aaron R. Voelker, and Chris Eliasmith. A neural representation of continuous space using fractional binding. In *CogSci*, 2019.

Nathan Lambert, Brandon Amos, Omry Yadan, and Roberto Calandra. Objective mismatch in model-based reinforcement learning. In *Proceedings of the 2nd Conference on Learning for Dynamics and Control (L4DC)*, volume 120 of *PMLR*, 2020. VERIFY pages.

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

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *ArXiv*, abs/2006.15055, 2020. URL https://api.semanticscholar.org/CorpusID:220127924.

Sindy Löwe, David Madras, Richard S. Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. In *CLEaR*, 2020. URL https://api.semanticscholar.org/CorpusID:219955853.

Heejeong Nam, Quentin Le Lidec, Lucas Maes, Yann LeCun, and Randall Balestriero. Causal-JEPA: Learning world models through object-level latent masking. In *2nd Workshop on Compositional Learning: Safety, Interpretability, and Agents*, 2026. URL https://openreview.net/forum?id=zKhi6pqma9.

Anna Penzkofer, Karim Habashy, Chris Eliasmith, and Andreas Bulling. Sspictr: A biologically-plausible image representation. In *ECAI Workshop on Artificial Intelligence and Cognition (AIC)*, pp. 1–13, 2025.

Tony A. Plate. *Holographic Reduced Representation: Distributed Representation for Cognitive Structures*. CSLI Publications, 2003.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20, 2007.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter W. Battaglia. Learning to simulate complex physics with graph networks. In *ICML*, 2020.

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

Donna Vakalis. The intervention gap in latent world models. *arXiv preprint arXiv:2608.29998*, 2026.

Rishi Veerapaneni, John D. Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua B. Tenenbaum, and Sergey Levine. Entity abstraction in visual model-based reinforcement learning. *ArXiv*, abs/1910.12827, 2019. URL https://api.semanticscholar.org/CorpusID:204904886.

Angel Villar-Corrales, Ismail Wahdan, and Sven Behnke. Object-centric video prediction via decoupling of object dynamics and interactions. In *2023 IEEE International Conference on Image Processing (ICIP)*, pp. 570–574, 2023. doi: 10.1109/ICIP49359.2023.10222810.

Ezra Webb, Ben Day, Helena Andrés-Terré, and Pietro Lio'. Factorised neural relational inference for multi-interaction systems. In *International conference on machine learning*, 2019.

Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models. In *ICLR*, 2023.

Kexin Yi*, Chuang Gan*, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. Clevrer: Collision events for video representation and reasoning. In *International Conference on Learning Representations*, 2020.

Andrii Zadaianchuk, Maximilian Seitzer, and Georg Martius. Object-centric learning for real-world videos by predicting temporal feature similarities, 2023. URL https://arxiv.org/abs/2306.04829.

Tianqiu Zhang, Muyang Lyu, Xiao Liu, and Si Wu. Structure abstraction and generalization in a hippocampal-entorhinal inspired world model. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=AYXgo5FjYz.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *ArXiv*, abs/2411.04983, 2024. URL https://api.semanticscholar.org/CorpusID:273878040.

A Datasets and preprocessing

NRI springs. Episodes come from the spring simulator of Kipf et al. (2018). $N = 5$ particles move in a box with elastic walls at ± 5. Each unordered pair is coupled independently with probability one half, and a coupled pair is joined by a zero-rest-length Hooke spring, $F_{ij} = -k\, e_{ij}(x_i - x_j)$ with $k = 0.1$, clipped at ± 100. Initial positions are drawn from $\mathcal{N}(0, 0.5)$ and initial speed is 0.5 in a random direction. Trajectories are integrated at $\Delta t = 10^{-3}$ for $5{,}000$ steps and saved every 100, giving 49 frames per episode. We generate $20{,}000$ training and $1{,}000$ validation episodes.

CLEVRER. We use the per-frame object annotations, not the rendered video. Each video supplies 128 annotated frames, subsampled at stride 2 to 64, with windows drawn every 6 frames. Videos are

kept if they hold between 2 and 6 objects over at least 8 frames, leaving 1,400 training videos. At tolerance 0, annotated collisions are 2.0% of object-steps.

Encoding and evaluation sets. Positions are encoded with a Gaussian-sampled phase matrix Θ at $D = 127$ and length scale $\ell = 0.2$, over $[-5.5, 5.5]^2$ on springs and $[-13.5, 12.5] \times [-12.5, 11.5]$ on CLEVRER. The identity vocabulary is drawn once from a fixed seed and shared across every model and seed. On springs, recovery uses 100 held-out episodes, rollout and deletion use 40, and the edits of Appendix **??** use 60. On CLEVRER, rollout uses 40 held-out videos and recovery uses 94.

B Training protocol and Model details

All experiments were conducted on a single workstation running 64-bit Windows 11 (Build 26200) equipped with an NVIDIA GeForce RTX 5090 GPU (Driver 595.95, CUDA 12.8, cuDNN 9.1.9). Models were implemented in Python 3.10.20 using PyTorch 2.11.0 (`cu128`), PyTorch Geometric 2.8.0, NumPy 1.26.4, and SciPy 1.15.3.

All models use Adam, no weight decay, batch size 64, gradient clipping at 1.0, masking probability 0.3, and input-noise scale $\sigma = 0.01$ entering the loss with $\lambda = 1$. Table 3 provides architecture dependant hyperparameters. On CLEVRER the same settings are used with $W = 10$ and 1,400 training videos, where SSP-JEPA has 752,511 parameters.

Table 3: Per-architecture settings on NRI springs; everything not listed is shared. Learning rate is the only tuned quantity, and 3×10^{-4} is used for C-JEPA because a six-layer Transformer diverges at the rate the two graph networks use. NRI sees the same 14 frames, with its own ten-step decoder objective.

	SSP-JEPA	coord-GNN	C-JEPA	NRI
learning rate	5×10^{-4}	5×10^{-4}	3×10^{-4}	5×10^{-4}
epochs	8	8	8	8
history window W	15	15	15	14
max masked objects	2	2	3	n/a
hidden width	256	256	128	256
depth	2 MLP layers	2 MLP layers	6 blocks	2 MLP layers
message rounds R	2	2	n/a	n/a
parameters	915,071	402,962	1,257,218	–
seeds	5	5	5	3

C Message-magnitude readout

The readout of Section3.4 perturbs the model. A second readout does not: it ranks the ordered pair (i, j) by $\|m_{j\to i}\|$, averaged over t. It is defined only for models that build an explicit per-pair message tensor, so it has no counterpart for C-JEPA.

Table 4: The two readouts on NRI springs, ROC-AUC against the true coupling graph (mean $\pm$ standard error over five seeds, 100 held-out episodes). The proximity floor is 0.679 for both. The edge-encoding row is scored on checkpoints trained at the earlier protocol (learning rate 10^{-3}, six epochs), which is why its ablation figure differs slightly from Table 2.

	ablation S_{ij}	magnitude $\|m_{j\to i}\|$
SSP-JEPA	$0.919_{\pm.005}$	$0.861_{\pm.004}$
− edge encoding	$0.882_{\pm.006}$	$0.556_{\pm.013}$
− history encoding	$0.919_{\pm.005}$	$0.829_{\pm.004}$
coord-GNN	$0.867_{\pm.003}$	$0.558_{\pm.006}$

The two readouts dissociate on the component Section5.4 identifies (Table 4). Removing the edge encoding costs 0.037 under the ablation readout and 0.305 under the magnitude readout, which

lands it below the proximity floor, exactly where coord-GNN sits. The SSP edge feature makes the coupling legible in the size of a message; a raw relative position leaves it recoverable only by intervening on the network. Residualising against inter-object distance leaves the gap intact, $0.807_{\pm.006}$ against $0.731_{\pm.012}$, $p = 5.3 \times 10^{-4}$ paired over five seeds.

D OBJECT-COUNT EXTRAPOLATION

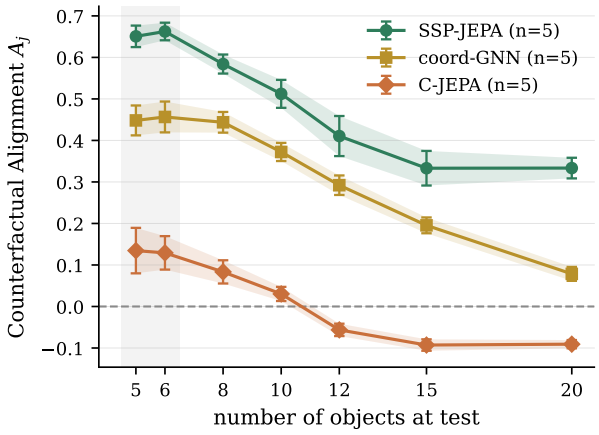


Figure 3: Direction of the predicted interaction at held-out object counts, with messages summed over senders (mean ± standard error over five seeds, 60 held-out episodes; the shaded band is the trained range, the dashed line is constant velocity). Above zero the predicted correction to coasting points with the true one, below zero it points against it.

To score the interaction compute counterfactual alignment with constant velocity prediction

$$A_i^{\mathrm{CV}} = \mathrm{sim}\big(\widehat{\Delta x}_i - \Delta x_i^{\mathrm{CV}},\ \ \Delta x_i - \Delta x_i^{\mathrm{CV}}\big).$$

Constant velocity scores zero by construction, so zero means no interaction model and a negative value means the correction points against the true interaction.

The models hold their order at every count and separate cleanly (Figure 3). SSP-JEPA falls from $0.66_{\pm.02}$ at $N = 6$ to $0.33_{\pm.03}$ at $N = 20$, keeping half of its in-range direction at a $3.3\times$ extrapolation. coord-GNN falls from $0.46_{\pm.04}$ to $0.08_{\pm.02}$, keeping a sixth. C-JEPA starts lowest at $0.13_{\pm.04}$, crosses zero between $N = 10$ and $N = 12$, and settles at $-0.09_{\pm.01}$, predicting a correction opposed to the true interaction.