Shaping Persistent Representations from Independent Interactions

Anonymous authors
Paper under double-blind review

ABSTRACT

World models learn environment dynamics from interaction experience. How a system evolves depends not only on its current state and actions but also on properties that persist across interactions. Yet standard predictive training can lower error using local evidence alone, without organizing persistent information into reusable context. We introduce **SPRII**, a training principle that uses relations between interactions as weak supervision for persistent context while retaining the learner's native objective. SPRII pairs related interactions, such as different trajectories of the same system: states and actions can differ while persistent properties remain shared. This pairing guides context learning without numerical property labels. Two composable training components encourage related contexts to agree (**Align**) and use one interaction's context to predict another's future (**Cross**). We build an analysis framework tracing three linked stages: organizing accessible persistent information in context (**Formation**), using it in a fixed predictor (**Use**), and reducing task error (**Value**). Success at one stage does not guarantee the next. Controlled experiments show that more reliable relations yield better organization, yet requiring pairs to share an additional property can make a still-shared property less accessible. Substituting context changes a fixed predictor's outputs, while prediction horizon and readout shape the benefit obtained from history. SPRII yields average gains of over 10% in downstream task performance and over 15% in persistent-property readout relative to the corresponding baselines. Evaluations spanning thirteen settings, including controlled physical systems, public dynamics tasks, robotic and tactile data, and partner interaction, further demonstrate the principle's applicability across learner families and interaction settings.

Anonymous resources: Code and configurations | Project page and demonstrations

1 INTRODUCTION

World models learn environment dynamics from interaction experience to predict future states and support decisions (Hafner et al., 2019; 2020; Zhou et al., 2025). In many systems, future evolution depends on both the current interaction and properties that remain stable across interactions. Consider two spring-connected balls driven by external forces: $s_{t+1} = f(s_t, a_t; \theta)$. The state s_t and action a_t describe the current configuration and applied force, while θ contains system properties such as mass m, drag γ and stiffness k. Different trajectories of the same system vary in their states and actions but share θ. These are *persistent properties*; an *interaction* is a realized episode, and a *history* is its observed sequence of states and actions.

Learning persistent properties helps a model account for the system-specific dynamics behind observed motion. Related histories can be especially useful when current observations leave these dynamics ambiguous or when the prediction target is more sensitive to them, as can occur at longer horizons. An object at rest reveals little about how quickly it will slow after a push; an earlier interaction can provide that information.

Context-conditioned models already use experience to infer how dynamics differ across systems (Lee et al., 2020; Shaj et al., 2022; Kirchmeyer et al., 2022). However, accurate prediction need not organize system information for reuse. Strong signals from current states, actions and recent motion can reduce the incentive to learn reusable context. Physical properties recoverable from observations

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.

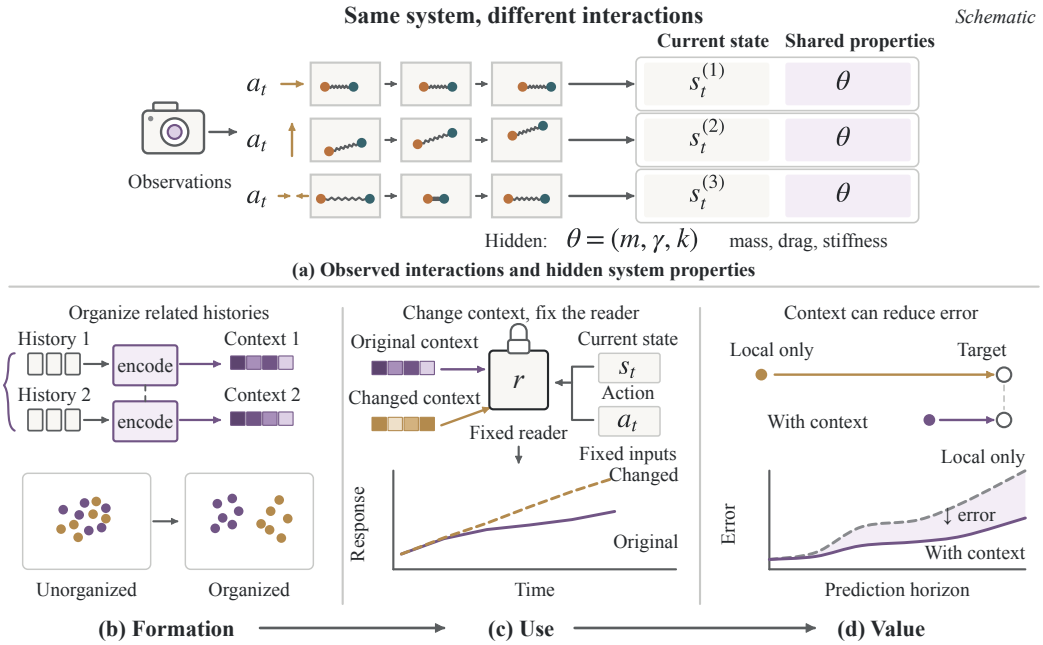


Figure 1: **From related interactions to useful persistent information.** All panels are schematic. (a) Interactions differ in state s_t and action a_t while sharing mass, drag and stiffness, $\theta = (m, \gamma, k)$. (b) Formation organizes persistent information. (c) Use tests its effect on a fixed reader's predictions. (d) Value tests whether it lowers task error. The linked questions distinguish information formation, predictive use and task benefit; success at one stage does not guarantee the next.

can remain difficult to read from predictive latents (Tan et al., 2026). Numerical property labels are often unavailable, leaving no direct supervision for organizing this information.

Relations between interactions offer weak but often available supervision. We may know that two histories concern the same system, object or partner without knowing its physical or behavioral parameters. Work on function representations and relational dynamics learning demonstrates the value of such relationships (Gondal et al., 2021; Guo et al., 2022). Pairing histories with shared properties but different states and actions directs learning toward information that persists across interactions. The relation need not supply numerical property values; same-system identity need not even specify which properties govern the shared dynamics.

We propose SPRII (Shaping Persistent Representations from Independent Interactions), a training principle that uses these relations to supervise *persistent context*. The predictor combines this history-derived context with the current input and actions. Two composable training components encourage related contexts to agree while maintaining variation (*Align*), and use one interaction's context to predict the future of another related interaction (*Cross*). SPRII makes the relation between histories an explicit training choice. Its components use agreement regularization (Bardes et al., 2022) and cross-trajectory prediction (Chen et al., 2026), retaining the native objective. Either component, or both, can be applied through an existing history state or a dedicated encoder (Figure 2).

Persistent information can be learned without being used, and used without improving prediction. We therefore introduce an analysis framework tracing three linked stages: whether shared information is organized and accessible in the learned context (*Formation*), whether changing that context affects a fixed predictor with current inputs and actions held fixed (*Use*), and whether its use reduces error for the specified task and readout (*Value*). Geometry and probes, donor substitutions, and task comparisons examine successive links in this chain; success at one does not guarantee success at the next. Figure 1 illustrates the chain, with analytical counterparts in Appendix I.

Controlled experiments reveal how these links depend on training and prediction choices. More reliable relations strengthen cross-interaction organization; changing which properties are shared redirects accessibility, even for a property that remains shared. Replacing context steers a fixed

reader's predictions along the substituted physical properties. In SpringWorld, history gains increase with horizon, and readout changes alter the benefit of frozen context (Figures 3–5).

Evaluation spans thirteen settings and latent-prediction, contrastive, recurrent state-space and neural-operator learners (Appendix A). Across five primary setting–learner comparisons in spring dynamics (SpringWorld) and object-interaction prediction (CoPhy; Baradel et al., 2020), error falls by 14.6% on average (range: 9.9–22.3%) relative to the native learner or a history-conditioned control without relation objectives (Table 1). On a controlled inertial system (D-Clean), 32-step error is 33.7% lower than CaDM under a common readout (Table 2). Recorded robotic and tactile data and partner interaction extend the study beyond simulated physical systems.

Our contributions are:

- **A relation-supervised training principle.** SPRII uses related interactions to supervise persistent context through composable Align and Cross components, retaining the native objective without numerical property labels.
- **A framework connecting information to task benefit.** Formation, Use and Value distinguish three linked stages, with operational tests and analytical counterparts; success at one does not ensure the next.
- **Findings on what is learned and when it helps.** Relation reliability and shared properties shape accessible information; context influences fixed predictors; horizon and readout shape the task benefit obtained from the same learned representation.
- **Validation across learners and interaction settings.** We evaluate predictive gains and persistent information in physical, robotic, tactile, partner-interaction and control settings across learner families.

2 A Relation-Supervised Training Principle

2.1 Interactions, relations and persistent properties

To predict a new spring trajectory (the recipient), a model can use an earlier trajectory of the same system (the donor). Denote these interactions by B and A. Recipient B has history H_B, current input C_B, permitted task conditioning u_B and target y_B; donor A supplies history H_A. The relation $\mathcal{R}_G(A, B)$ identifies shared persistent information, for example through system, task or partner identity. In controlled systems, G specifies the shared subset of θ, such as mass while drag varies.

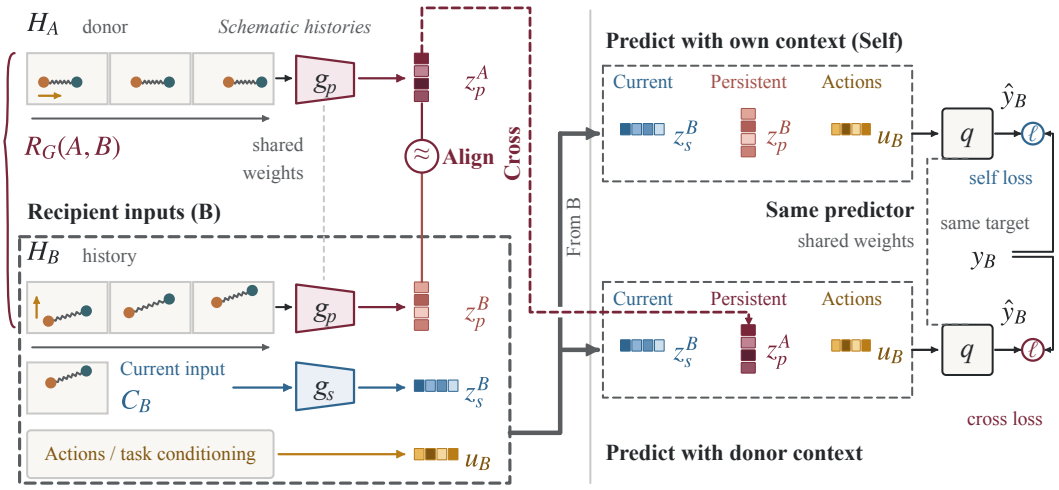
Training samples a recipient, then a related donor, defining pair distribution Π_G; representation training uses relation membership rather than numerical factor labels. Here, *independent interactions* denotes separately realized episodes or trajectories. The relation defines what is shared, while the sampling rule selects the legal donor. Sharing an additional factor can therefore change both the shared information and the selected pairs. Appendix B.1 specifies each relation and sampling law.

2.2 A persistent–current interface

SPRII separates history-derived context from recipient-specific current information through a shared history encoder g_p and a current-input encoder g_s:

$$z_p^A = g_p(H_A), \qquad z_p^B = g_p(H_B), \qquad z_s^B = g_s(C_B). \tag{1}$$

Here z_s denotes current-input features, such as image embeddings, rather than the physical state itself. A dedicated encoder or a projection of an existing history state implements g_p in one forward pass. Context can encode several shared conditions in distributed form; prediction uses it directly, while evaluation probes assess its information content. For action-conditioned prediction, u_B contains the permitted future actions, masks and horizon. The predictor q combines these inputs with a persistent context, whose source can be the recipient or a related donor.



(a) Histories and current inputs **(b) Prediction with shared q**

Figure 2: **Relation-supervised learning of persistent context.** Thick arrows carry recipient inputs (a) into both prediction branches (b). (a) Align regularizes contexts paired by $\mathcal{R}_G$. (b) Both calls share predictor q, current input C_B and conditioning u_B; Cross replaces only z_p^B with related donor context z_p^A. y_B is the loss target. Both components augment the native objective. Trajectories and tokens are schematic; Figure 8 shows the JEPA implementation.

2.3 Align and Cross

Consistency across related histories. Align brings related contexts together while preserving variation. We implement $\mathcal{L}_{\text{align}}$ with VICReg (Bardes et al., 2022): paired-context agreement, per-dimension variance and a cross-dimension covariance penalty. The relation determines which pairs receive this pressure (Appendix B.1).

Prediction with context from another interaction. Cross trains prediction with related donor context. Self and cross-interaction paths share the recipient's current inputs:

$$\begin{aligned} \hat{y}_B^{\text{self}} &= q(z_s^B, z_p^B, u_B), \\ \hat{y}_B^{A\to B} &= q(z_s^B, z_p^A, u_B). \end{aligned} \tag{2}$$

Switching the context source from H_B to H_A gives the Cross objective:

$$\mathcal{L}_{\text{cross}} = \mathbb{E}_{(A,B)\sim\Pi_G}\left[\ell_{\text{task}}\left(\hat{y}_B^{A\to B}, y_B\right)\right]. \tag{3}$$

The recipient target supplies supervision. Because current inputs may already suffice, we test reliance on context directly through the interventions in Section 3.3.

Joint training. The two operations augment the native objective,

$$\mathcal{L} = \mathcal{L}_{\text{base}} + \lambda_{\text{p}}\mathcal{L}_{\text{align}} + \lambda_{\text{x}}\mathcal{L}_{\text{cross}}, \tag{4}$$

where $\mathcal{L}_{\text{base}}$ retains the learner's prediction loss and regularizers. In our NOD and Overcooked instantiations, the native objective already uses other interactions, so we add Align. The training recipes in Appendix B.3 use Align, Cross or both, with operations and weights fixed before evaluation.

2.4 Applying the Principle to Existing Learners

In JEPA, an observation encoder and causal Transformer combine observation and action histories into z_p; an MLP of the last observation embedding produces z_s. The predictor uses both, future actions and a horizon embedding to predict the recipient's future embedding. CPC and RSSM retain their native contrastive and variational objectives; supervised dynamics and action learners retain their task losses (Appendices B.1–B.3; Figure 8).

At inference, SpringWorld encodes an independent same-system history without gradient updates; other settings identify donors by system, task or partner. SpringWorld's physical prediction and Use/Value tests use a task reader r fitted to frozen encoder features. This reader is separate from q, which predicts embeddings during source training.

3 EXPERIMENTS

SpringWorld shows elastic-coupling dynamics in 128×128 images; PokeWorld shows an actuated finger pushing an object. Both vary mass, drag and stiffness across systems while holding them fixed within each system. SpringWorld tests held-out prediction, Use and Value; PokeWorld varies which factors pairs share. Public benchmarks and robotic, tactile, multi-agent and control tasks test generality (Appendices A, B.4).

We first compare predictive performance, then examine Formation through geometry and probes, Use through fixed-predictor context substitutions, and Value through task-reader and horizon comparisons. A *source* is a trained representation model; a reader predicts from its frozen features. Controls are the history-conditioned host (Native), a split history interface without relation objectives (Structure), and random pairing with the same relation objective (Random). Readers share targets and permitted inputs within each comparison. Appendix B.6 specifies aggregation and selection; native embedding losses remain separate.

3.1 PREDICTIVE PERFORMANCE

A 96-frame same-system history conditions a SpringWorld query with no preceding motion history; frozen-source readers predict state increments over the next 0.8 seconds. On 256 sealed systems (held out until the final evaluation), all nine source/reader pairs favor SPRII over Native and TDS (Table 1A). CoPhy gains span three Collision learners and Balls RSSM, with budgets matched within each learner (Table 1B).

Table 1: **Primary predictive performance on SpringWorld and CoPhy.** Lower is better; bold marks the lowest comparable mean.[1]

A. SpringWorld: single-state queries, 256 sealed systems

Source recipe	MSE ↓	SPRII reduction	95% CI
Native JEPA	0.138 ± 0.002	22.3%	[13.2, 30.7]%
TDS	0.130 ± 0.013	17.5%	[8.4, 25.9]%
SPRII (Align + Cross)	**0.107** ± 0.004	Reference	

B. CoPhy: physical trajectory MSE

	Collision			Balls4
Source recipe	**CPC**	**RSSM**	**JEPA**	**RSSM**
Native	0.248 ± 0.008	0.268 ± 0.007	0.250 ± 0.004	1.292 ± 0.037
Structure	0.223 ± 0.006	0.228 ± 0.005	0.215 ± 0.007	1.280 ± 0.031
Random	0.221 ± 0.006	0.222 ± 0.007	0.265 ± 0.003	1.327 ± 0.011
SPRII (Cross)	**0.189** ± 0.007	**0.202** ± 0.007	**0.184** ± 0.005	**1.154** ± 0.025

SPRII lowers error relative to released NOD in all three Burgers distributions (Table 2A). With matched additional updates, SPRII retains a 4.1% ID gain; continued NOD has up to 1.1% lower OOD error (Appendix C.4). Applied to an amortized variant of CoDA, SPRII lowers mean zero-step error (Table 2A; Appendix C.5). Using a common reader design across the D-Clean methods, SPRII gives the lowest mean error at all four prediction horizons, including 33.7% lower error than CaDM at 32 steps (Table 2B).

[1]Table 1: SpringWorld uses standardized state-increment MSE, three sources × three readers, source mean ± SD and paired-system CIs. CoPhy reports physical trajectory MSE (mean ± SD over three joint seeds), on 1,994 Collision episodes and 1,000 Balls4 recipients (Appendices C, B.6).

Table 2: **Predictive comparisons on Burgers and D-Clean.** Lower is better; bold marks the lowest mean within each comparable host or reader group.[2]

A. Burgers: native-trajectory / zero-step MSE ($\times 10^{-3}$)

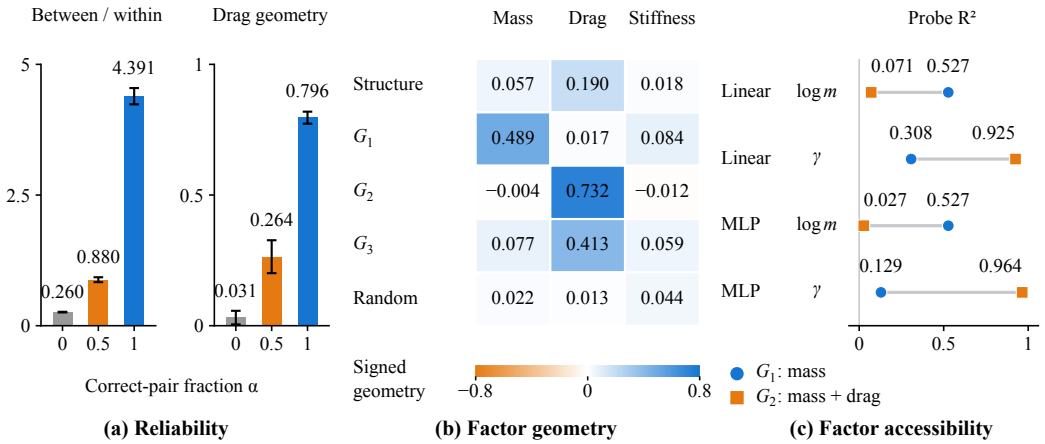
Host learner	Source recipe	ID	Viscous OOD	Inviscid OOD
NOD (arXiv'26)	Released configuration	9.096 ± 0.544	8.012 ± 0.815	11.857 ± 0.628
	+ SPRII	**8.999** ± 0.798	**7.487** ± 0.183	**11.510** ± 0.360
Amortized CoDA (ICML'22)	Plain	3.483 ± 0.567	6.390 ± 1.436	5.883 ± 1.700
	+ SPRII (Cross)	**3.250** ± 0.729	**6.060** ± 1.185	**5.501** ± 1.496

B. D-Clean: common-reader raw-state MSE ($\times 10^{-4}$)

Method	h1	h4	h16	h32
State-space TDS	0.1034	0.8009	4.557	13.43
FCRL (ICML'21)	0.1228 ± 0.0112	0.7620 ± 0.0809	4.833 ± 0.492	13.34 ± 1.014
CaDM (ICML'20)	0.0332 ± 0.0018	0.1474 ± 0.0081	0.7711 ± 0.0081	2.273 ± 0.0628
DALI context (NeurIPS'25)	0.3201 ± 0.0357	2.732 ± 0.3115	17.26 ± 1.991	45.92 ± 5.362
SPRII	**0.0314** ± 0.0029	**0.1166** ± 0.0051	**0.5232** ± 0.0117	**1.507** ± 0.1355

3.2 Formation: Relations Shape Strength and Accessible Content

PokeWorld permits controlled changes to pairing accuracy and to the properties pairs share, with source architecture and observation budgets fixed.



(a) Reliability **(b) Factor geometry** **(c) Factor accessibility**

Figure 3: **Relations shape the strength and content of persistent representations.** (a) Pair reliability strengthens separation and drag geometry; whiskers: source SD. (b) Partial geometry. (c) Linear/MLP probe R^2. $G_1 = m$, $G_2 = (m, \gamma)$, $G_3 = (m, \gamma, k)$. Means cover three sources. Geometry uses 400 confirmation systems; probes use 400 validation systems (Appendix E.1).

Reliable relations strengthen organization. On 400 confirmation systems, raising the correct-pair fraction from 0 to 1 increases the between/within-system context-distance ratio from 0.260 to 4.39. Partial drag geometry, measured by distance correlation controlling the other factors, rises from 0.031 to 0.796 (Figure 3(a)). Observation budgets, donor/query marginals and model settings are fixed.

[2]Table 2: Burgers reports mean ± SD over three sources, with 45/500/100 ID/viscous/inviscid queries. D-Clean uses 100 systems and three sources × three readers; SD is across source means. DALI is a context-component adaptation with 160,000 source updates; other D-Clean sources use 20,000. This is not an equal-total-compute comparison. Training and evaluation protocols are in Appendices C.3–C.5.

Changing the pairing rule redirects accessibility. Changing $G_1 = m$ to $G_2 = (m, \gamma)$ keeps mass shared but lowers its linear probe R^2 from 0.527 to 0.071; drag rises from 0.308 to 0.925. These probes use per-system context centroids (Figure 3(c)). Nonlinear probes and all three source replicas reproduce both directions, showing that changing the pairing rule redirects factor accessibility.

In the capacity-limited linear reference of Appendix I.2, with code dimension d smaller than history dimension p, the selected persistent subspace is a top-d eigenspace of

$$M_G = K^\top K - \beta D_G, \qquad \beta \geq 0,$$

where K maps history residuals to target residuals after removing information explained by current inputs and conditioning, and D_G is the second-moment matrix of paired-history differences. This balances predictive relevance against relation disagreement: changing the relation can redirect the selected directions even when earlier factors remain shared.

Align organizes context; Cross adds benefit. SpringWorld's component study uses 178 systems for geometry and 100 development systems (600 cases; one reader per source) for task error, separate from the sealed evaluation in Table 1. From Structure's separation ratio 0.240 and reader error 0.172, Align reaches 2.05 and 0.123; Cross reaches 0.252 and 0.168; both reach 2.36 and 0.110 (Appendix E.2). Align supplies most of the organization and task gain; adding Cross lowers error in all three sources. In Collision, CPC, RSSM and JEPA use Cross to train the donor-context route.

3.3 Use: persistent context enters fixed predictive computation

We next substitute donor context to test its effect on a fixed predictor. In SpringWorld, we intervene on a task reader fitted to frozen features, separately from the source predictor q.

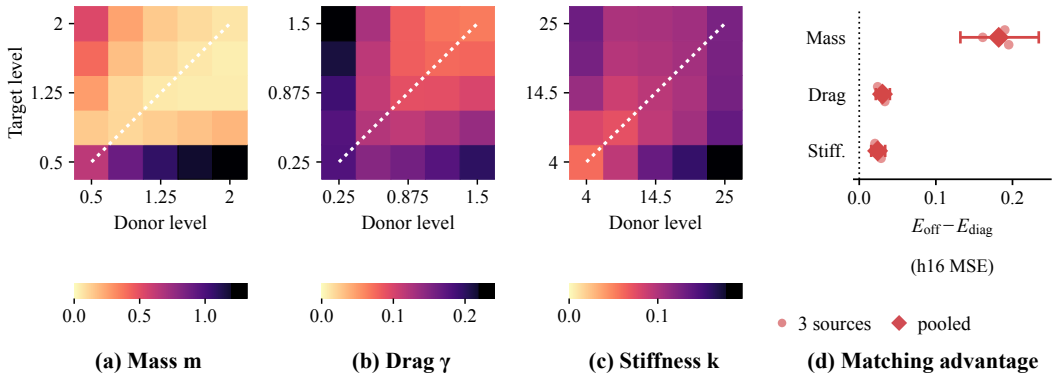


Figure 4: **Donor physics structures fixed predictive responses.** (a)–(c) Donor–target landscapes of h16 standardized-increment MSE, averaged over 64 development systems and three Align + Cross sources; color scales differ. Only donor context changes the fixed reader's prediction. (d) Mean off-diagonal minus diagonal error; positive favors matching. Points: sources; diamonds: means; intervals: pointwise 95% paired-system CIs, conditional on fitted models (Appendix F.1).

Donor physics changes which dynamics a fixed predictor fits. In SpringWorld, we hold the source, fitted reader, recipient initial state and future actions fixed, then vary one donor factor. Independently varying the corresponding simulator target factor yields a 5×5 donor–target landscape for each of 64 development systems. Simulator targets serve only to compute prediction error. Matching donor and target physics lowers mean 16-step (h16) error for mass, drag and stiffness in every source (Figure 4(a)–(d)). Donor context therefore steers the fixed reader's predictions toward the corresponding physical dynamics.

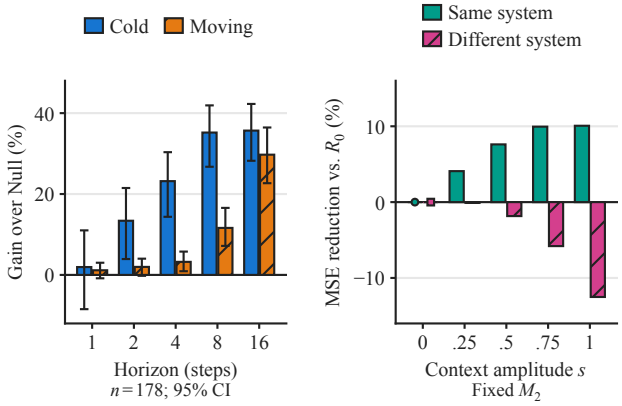
Align strengthens the advantage of matching context. Using the same matching-advantage measure as Figure 4(d), which shows the joint recipe, we extend the comparison to Structure, Align and Cross. The Align effect, averaged over the presence and absence of Cross, is positive for all three

factors. Each recipe uses its own source and fitted reader. Appendix F.1 gives complete effects and a decomposition of the same readers' physical output responses.

Relation design has a predictive counterpart. On 144 PokeWorld validation systems, a fixed reader's predictions respond more strongly along mass under G_1 and drag under G_2 (Appendix F.2).

3.4 VALUE: PREDICTION CONDITIONS AND THE READER SHAPE BENEFIT

We vary the horizon and reader to test the task benefit of frozen context (Appendix I.3), keeping source features unchanged.



(a) Horizon value **(b) Context value**

(c1) Residual readout (MSE)

Source	M_1	M_2	Gain (%)
0	0.608	0.550	9.6
1	0.627	0.585	6.8
2	0.620	0.574	7.5
Overall	**0.619**	**0.570**	**7.9**

(c2) Persistent vs. Decode

Task	MSE change (%) ↓
D-Clean	**−22.8**
SpringWorld	**+2.1**

Figure 5: **Persistent context has prediction- and readout-dependent value.** (a) SpringWorld gain over Null readers as horizon increases; whiskers: pointwise 95% system-bootstrap intervals over 178 systems. (b) Scaling context in fixed M_2 changes MSE relative to recipient-only R_0; $s = 0$ gives R_0. (c1) Query-only M_1 versus context-conditioned M_2: source means over three readers. (c2) MSE change from Decode to Persistent (%); negative favors direct context. In (c2), endpoints are D-Clean h32 and SpringWorld cold-start h16 (Appendix G).

History value changes with the prediction horizon. Null readers receive no history; Persistent readers receive learned context, Decode readers receive decoded physical parameters, and Oracle readers receive true parameters. SpringWorld's Cold and Moving queries both provide zero observed transitions and differ in initial motion. Persistent-over-Null gain rises from 2.0% at h1 to 35.7% at h16 for Cold, and from 1.2% to 29.7% for Moving (Figure 5(a)). All comparisons use the same 178 systems and fitted grid; horizon changes the target while query evidence stays fixed.

Decodable parameters do not determine the best readout. Persistent and Decode readers receive the same donor context, either directly or through decoded physical parameters, with a shared frozen query encoder. In D-Clean, direct context lowers h32 error by 22.8% relative to Decode despite independently validated decoder $R^2 > 0.996$. In SpringWorld, direct context instead has 2.1% higher error than Decode (Figure 5(c2)). The preferred context interface therefore depends on the task, even when physical parameters are accurately decodable (paired effects and readers: Appendix G.2).

A residual readout tests the added value of context. We freeze source features and a fitted recipient-only base R_0, then fit query-only (M_1) or context-conditioned (M_2) residual readers with common targets and schedules. On SpringWorld development systems, M_2 lowers mean MSE from 0.619 to 0.570 relative to M_1, a 7.9% reduction (Figure 5(c1)); all nine source/reader pairs improve (Appendices B.5, G.4).

With fixed M_2, scaling its persistent input by s lowers error for same-system context and raises it for different-system context (Figure 5(b)). Zero amplitude equals the frozen recipient-only base, distinct from the separately fitted M_1. These substitutions show sensitivity to system-specific content.

Training and evaluating the same reader architecture with mismatched context raises error in all nine source/reader pairs (Appendix G.4), confirming the importance of correctly associated history.

The readout improves SpringWorld prediction, whereas M_1 has lower error in all five evaluated PokeWorld G_1/G_2 runs (Appendix G.4). Realized value therefore depends on the reader and prediction problem. Appendix I.4 formalizes available task value and fitted-reader realization. In pendulum control (Appendix G.5), context yields higher returns than CaDM in four of five conditions.

3.5 Generality

Relations can also express task, object and partner identity. In offline forecasting from recorded RH20T manipulation, task-matched context under Align + Cross improves force and motion forecasts relative to specified mismatched-donor controls (Appendix F.3). In real Baxter grasps, hardness- and shape-based relations favor readout of the corresponding property (Appendix E.3); in cooperative Overcooked play, same-partner alignment strengthens partner-behavior readout (Appendix H.2). The broader CoPhy matrix is reported in Appendix H.1. These settings change both the observations and what persists: physical dynamics in simulation, object properties in touch, and behavior across encounters with a partner. Each retains its own task, with relations as a common form of supervision.

4 Related Work

Context-conditioned dynamics models infer environment factors from transitions (Lee et al., 2020; Röder et al., 2025; Shaj et al., 2022). NOD conditions prediction on same-system trajectories (Chen et al., 2026). GG-ODE regularizes environment codes contrastively (Huang et al., 2023); RIA infers shared-environment relations through interventional prediction (Guo et al., 2022). FCRL and view-design studies examine how shared information shapes representations (Gondal et al., 2021; Tian et al., 2020; von Kügelgen et al., 2021); VICReg supplies Align's agreement objective (Bardes et al., 2022). These approaches establish ways to learn context or use histories in prediction. SPRII treats relation reliability and shared properties as training choices, then traces their effects through three complementary measurements: geometry and probes of the learned context, donor substitutions through fixed predictors, and task comparisons under different readouts. This distinguishes the information made accessible by a relation from its use and realized benefit (Appendix J).

5 Conclusion

SPRII uses relations between interactions to strengthen persistent context learning through Align, Cross or both, while retaining the learner's native objective. Relations shape what information becomes accessible; changing context steers a fixed predictor's outputs; and horizon and readout shape the resulting task benefit. Formation, Use and Value connect these findings and guide the evaluation of context through probes, interventions and task comparisons. Repeated interactions make this principle relevant across predictive learning and control. The central design choice is which experiences a learner is encouraged to treat as related. The relation should reflect the information needed by the task: sharing additional properties can redirect accessibility rather than simply enrich the representation. A practical starting point is to group histories using available system, object or partner identities, then test whether the resulting context is informative and useful to the consuming predictor. The principle can thus be adopted through existing history interfaces without committing to one architecture or assuming the same benefit in every task. Future work can discover relations from interaction logs, track properties that vary over longer timescales, and integrate context with planning, using the same framework to distinguish available information from realized task value.

Reproducibility. Evaluation protocols, numerical records and figure code/data are provided in the Supplementary Evidence Package (Appendix B.7).

AI use disclosure. AI tools assisted literature retrieval, editing, figure preparation and consistency checks. The authors are responsible for all scientific claims, numerical results and final verification.

REFERENCES

Youssef Amin, Christian Gianoglio, and Maurizio Valle. Embedded real-time objects' hardness classification for robotic grippers. *Future Generation Computer Systems*, 148:211–224, 2023. doi: 10.1016/j.future.2023.06.002. URL https://doi.org/10.1016/j.future.2023.06.002.

Youssef Amin, Christian Gianoglio, and Maurizio Valle. IntelliMan WP5: Grasping, Manipulation and Arm-Hand Coordination—Data Fusion and Sensing Technology, 2026. URL https://doi.org/10.5281/zenodo.18246104. Dataset, version 1.0.

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, 2023. URL https://arxiv.org/abs/2301.08243.

Randall Balestriero and Yann LeCun. LeJEPA: Provable and scalable self-supervised learning without the heuristics, 2025. URL https://arxiv.org/abs/2511.08544.

Fabien Baradel, Natalia Neverova, Julien Mille, Greg Mori, and Christian Wolf. CoPhy: Counterfactual learning of physical dynamics. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SkeyppEFvS.

Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning, 2022. URL https://arxiv.org/abs/2105.04906.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video, 2024. URL https://arxiv.org/abs/2404.08471.

Zituo Chen, Qiaofeng Li, Jiaxin Hu, and Sili Deng. Neural operator discovery from heterogeneous trajectories, 2026. URL https://arxiv.org/abs/2607.23337.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2 edition, 2006. ISBN 9780471241959. doi: 10.1002/047174882X.

Hao-Shu Fang, Hongjie Fang, Zhenyu Tang, Jirong Liu, Chenxi Wang, Junbo Wang, Haoyi Zhu, and Cewu Lu. RH20T: A comprehensive robotic dataset for learning diverse skills in one-shot, 2023. URL https://arxiv.org/abs/2307.00595.

Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *Journal of Machine Learning Research*, 20(177):1–81, 2019. URL https://jmlr.org/papers/v20/18-760.html.

Tobias Gessler, Tin Dizdarevic, Ani Calinescu, Benjamin Ellis, Andrei Lupu, and Jakob Nicolaus Foerster. OvercookedV2: Rethinking overcooked for zero-shot coordination, 2025. URL https://arxiv.org/abs/2503.17821.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.

Muhammad Waleed Gondal, Shruti Joshi, Nasim Rahaman, Stefan Bauer, Manuel Wuthrich, and Bernhard Schölkopf. Function contrastive learning of transferable meta-representations. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 3755–3765. PMLR, 2021. URL https://proceedings.mlr.press/v139/gondal21a.html.

Aditya Grover, Maruan Al-Shedivat, Jayesh Gupta, Yuri Burda, and Harrison Edwards. Learning policy representations in multiagent systems. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1802–1811. PMLR, 2018. URL https://proceedings.mlr.press/v80/grover18a.html.

Jiaxian Guo, Mingming Gong, and Dacheng Tao. A relational intervention approach for unsupervised dynamics generalization in model-based reinforcement learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=YRq0ZUnzKoZ.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2555–2565. PMLR, 2019. URL https://proceedings.mlr.press/v97/hafner19a.html.

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

Zijie Huang, Yizhou Sun, and Wei Wang. Generalizing graph ODE for learning complex system dynamics across environments. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023. URL https://arxiv.org/abs/2307.04287.

Yuheng Jing, Kai Li, Ziwen Zhang, Jiajun Zhang, Zeyao Ma, Jiaxi Yang, Lei Zhang, Zhe Wu, Jinmin He, Junliang Xing, and Jian Cheng. Benchmarking the limits of in-context reinforcement learning for ad-hoc teamwork, 2026. URL https://arxiv.org/abs/2605.24423.

Armand Kassaï Koupaï, Jorge Mifsut Benet, Yuan Yin, Jean-Noël Vittaut, and Patrick Gallinari. Boosting generalization in parametric PDE neural solvers through adaptive conditioning. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/82844e428d9163a9f94830dc03af4f9c-Abstract-Conference.html.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/d89a66c7c80a29b1bdbab0f2a1a94af8-Abstract.html.

Matthieu Kirchmeyer, Yuan Yin, Jérémie Donà, Nicolas Baskiotis, Alain Rakotomamonjy, and Patrick Gallinari. Generalizing to new physical systems via context-informed dynamics model. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 11283–11301. PMLR, 2022. URL https://proceedings.mlr.press/v162/kirchmeyer22a.html.

Michael Laskin, Luyu Wang, Junhyuk Oh, Emilio Parisotto, Stephen Spencer, Richie Steigerwald, DJ Strouse, Steven Hansen, Angelos Filos, Ethan Brooks, Maxime Gazeau, Himanshu Sahni, Satinder Singh, and Volodymyr Mnih. In-context reinforcement learning with algorithm distillation. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.14215.

Kimin Lee, Younggyo Seo, Seunghyun Lee, Honglak Lee, and Jinwoo Shin. Context-aware dynamics model for generalization in model-based reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5757–5766. PMLR, 2020. URL https://proceedings.mlr.press/v119/lee20g.html.

Christoph Molnar, Timo Freiesleben, Gunnar König, Julia Herbinger, Tim Reisinger, Giuseppe Casalicchio, Marvin N. Wright, and Bernd Bischl. Relating the partial dependence plot and permutation feature importance to the data generating process. In *Explainable Artificial Intelligence*, volume 1901 of *Communications in Computer and Information Science*, pp. 456–479. Springer, 2023. doi: 10.1007/978-3-031-44064-9_24.

Frank Röder, Jan Benad, Manfred Eppe, and Pradeep Kr. Banerjee. Dynamics-aligned latent imagination in contextual world models for zero-shot generalization. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2508.20294.

Alexander Rutherford, Benjamin Ellis, Matteo Gallici, Jonathan Cook, Andrei Lupu, Garðar Ingvarsson, Timon Willi, Ravi Hammond, Akbir Khan, Christian Schroeder de Witt, Alexandra Souly, Saptarashmi Bandyopadhyay, Mikayel Samvelyan, Minqi Jiang, Robert Lange, Shimon Whiteson, Bruno Lacerda, Nick Hawes, Tim Rocktäschel, Chris Lu, and Jakob Foerster. JaxMARL: Multi-agent RL environments and algorithms in JAX. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/5aee125f052c90e326dcf6f380df94f6-Abstract-Datasets_and_Benchmarks_Track.html.

Vaisakh Shaj, Dieter Büchler, Rohit Sonker, Philipp Becker, and Gerhard Neumann. Hidden parameter recurrent state space models for changing dynamics scenarios, 2022. URL https://arxiv.org/abs/2206.14697.

Kaizhen Tan, Xin Xu, Siru Tao, Yixiao Li, Hanzhe Hong, Yang Feng, and Heqing Du. What can latent world models know? physical parameter identifiability in multimodal predictive representations, 2026. URL https://arxiv.org/abs/2607.27017.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c2e5eaae9152079b9e95845750bb9ab-Abstract.html.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2018. URL https://arxiv.org/abs/1807.03748.

Julius von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/8929c70f8d710e412d38da624b21c3c8-Abstract.html.

David S. Watson and Marvin N. Wright. Testing conditional independence in supervised learning algorithms. *Machine Learning*, 110:2107–2129, 2021. doi: 10.1007/s10994-021-06030-6.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning, 2025. URL https://arxiv.org/abs/2411.04983.

APPENDIX CONTENTS

A Findings, Practical Use, and Evaluation Guide . . . 14
- **A.1 Task and property gains at a glance** . . . 14
- **A.2 Discussion: putting the principle to work** . . . 14
- **A.3 Findings and broader implications** . . . 15
- **A.4 Evaluation settings** . . . 16

B Method and Experimental Implementation . . . 19
- **B.1 Relations, objectives and training update** . . . 19
- **B.2 Shared controlled-JEPA implementation** . . . 20
- **B.3 Cross-learner instantiation** . . . 21
- **B.4 Data, splits and admissible inputs** . . . 21
- **B.5 Probes, fresh readers and fixed consuming routes** . . . 27
- **B.6 Measurement, aggregation and selection** . . . 28
- **B.7 Reproduction assets** . . . 30

C Predictive Performance and Closest Comparisons . . . 30
- **C.1 Cold-start prediction on sealed systems: SpringWorld** . . . 30
- **C.2 Held-out prediction: Collision and Balls** . . . 31
- **C.3 Matched comparisons with adapted constructions** . . . 32
- **C.4 Prediction and continuation on a released operator: Burgers** . . . 34
- **C.5 Relation supervision on a frozen CoDA host** . . . 36

D SpringWorld Case Study . . . 38
- **D.1 History-conditioned baselines** . . . 38
- **D.2 Contributions of Align and Cross** . . . 38
- **D.3 Computational cost** . . . 39
- **D.4 Budgets and configuration selection** . . . 39
- **D.5 Sensitivity to relation weights** . . . 39
- **D.6 Gradient relationships** . . . 40
- **D.7 Relation reliability: PokeWorld comparison** . . . 40

E Formation: Relation Design and Representation Organization . . . 41
- **E.1 Relation reliability, shared factors, and objectives: PokeWorld** . . . 41
- **E.2 Objective components and reference learners: SpringWorld** . . . 44
- **E.3 Relation-selective tactile organization: Baxter** . . . 46
- **E.4 Structured relation errors** . . . 46
- **E.5 Relations inferred from learned histories** . . . 48

F Use: Fixed-Route Context Interventions . . . 49
- **F.1 Physical donor landscapes: SpringWorld** . . . 49
- **F.2 Factor-specific responses and direction: PokeWorld** . . . 51
- **F.3 Task-specific donor use in recorded interactions: RH20T** . . . 52
- **F.4 Physical information in reader memory: Balls** . . . 56
- **F.5 Recipient-specific updates through frozen weights: Swimmer** . . . 57

G Value: Conditional Task Benefit and Reader Realization . . . 58
- **G.1 Prediction horizon and recipient evidence: SpringWorld** . . . 58
- **G.2 Direct context versus physical bottlenecks** . . . 58
- **G.3 Aggregate and component-level benefit: PokeWorld** . . . 60
- **G.4 Realizing value through frozen-source readouts** . . . 61
- **G.5 Closed-loop control with a development-selected recipe: Pendulum** . . . 64

H Generality Across Learners, Partners, and Fields . . . 65
- **H.1 Learner and scene transfer: CoPhy** . . . 65
- **H.2 Recurring-partner behavior and cooperation: Overcooked** . . . 69
- **H.3 History-count scaling in reaction-diffusion fields: FHN** . . . 70

I Theoretical Connections . . . 71
- **I.1 Setup and lifecycle** . . . 71
- **I.2 Formation: pairing, reliability and predictive selection** . . . 72
- **I.3 Use: fixed-predictor reliance** . . . 75
- **I.4 Value: conditional risk and reader realization** . . . 75

J Closest Methodological Connections . . . 77

A Findings, Practical Use, and Evaluation Guide

A.1 Task and property gains at a glance

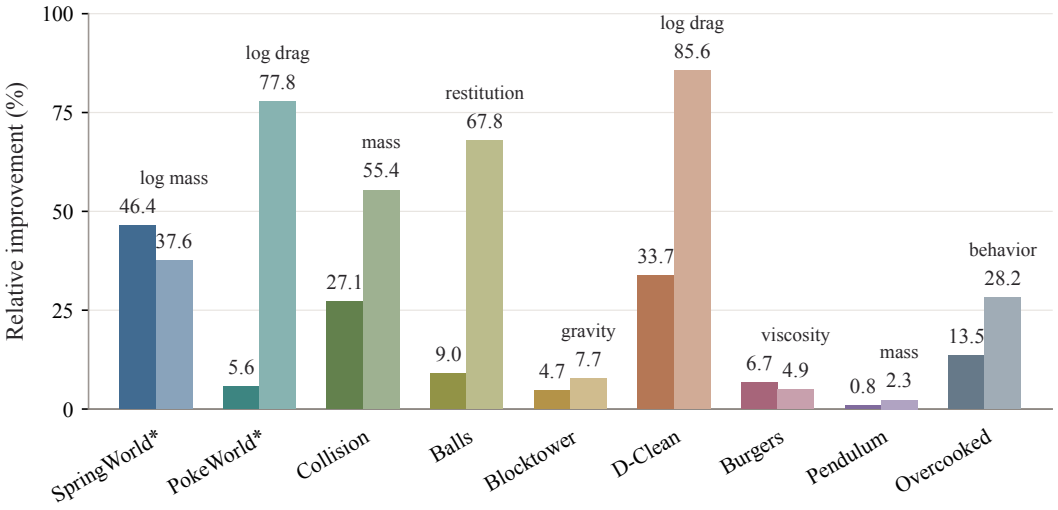


Figure 6: **Selected task and property gains. Left:** downstream task improvement. **Right:** improved readout of the labeled property. Both are relative to the corresponding baseline. Asterisks mark single-source examples; Figure 7 gives comparison details.

Figure 6 illustrates two useful outcomes of learning across related encounters: better task performance and clearer access to persistent properties. The examples span trajectory prediction, field forecasting, control and cooperation with recurring partners. These gains need not grow together: a property can become much easier to read while the task improves only modestly. The practical lesson is to examine both what a history representation contains and what a learner can do with it.

A.2 Discussion: putting the principle to work

(1) Where the idea applies. Many learning problems involve repeated encounters with a system whose immediate state changes more readily than some of its underlying properties. A new trajectory may begin at a different position; an object may be manipulated in a different way; a familiar partner may face a new situation. Earlier encounters could still contain useful evidence about how each will respond. The practical opportunity is therefore to connect experiences that would otherwise be treated as separate examples. This does not require repeating the same task or observing exactly the same conditions. It requires a reason to expect some relevant information to persist across the encounters being connected. Object identifiers, repeated experimental conditions, or recurring partners may provide such a connection without numerical labels for the properties themselves. In a new application, a useful starting point is to identify what recurs, over what period it is reasonably stable, and which earlier observations would be available when a new prediction is needed.

(2) How to put it into a model. A useful design begins by separating what the model must learn from the current situation from what earlier experience could already tell it. The current input describes the particular prediction being made, while a history representation provides a route for information to carry across encounters. An existing learner with a history state may already offer that route; another learner could obtain it from a separate encoder. The task itself can remain the organizing objective. Relation supervision encourages experience from one encounter to be useful in another, without changing what the model is ultimately asked to predict. In practice, this also makes the organization of recorded experience part of model design: a training example may be accompanied by earlier, related experience as well as its immediate inputs. The corresponding deployment arrangement should make clear where that history comes from and when it becomes available.

(3) How it may be used. Consider a robot encountering the same object several times. A brief earlier push or contact could provide context for predicting the outcome of a later movement, even when the object's current appearance gives little indication of its response. A predictive model could combine that experience with the new state and proposed action; a subsequent planning system could then use the resulting forecast when comparing actions. For a dynamics application, trajectories recorded under the same physical conditions could provide context for forecasting a new trajectory from different initial conditions. For repeated cooperation, earlier exchanges with a partner could inform predictions of that partner's behavior in a new situation. In these applications, history could be organized around an object, a physical regime, or a partner. The same accumulated knowledge could then support several predictions as tasks and immediate conditions change. Experience could remain useful beyond the episode in which it was obtained. These applications also distinguish the reusable idea from a particular benchmark: the historical observations, current inputs, and desired outputs can differ across applications. The studies below examine specific instances of this idea; the broader applications described here remain possibilities whose value would need to be established in their own setting.

A.3 Findings and broader implications

The studies trace how experience becomes reusable: what is learned, how it enters a prediction, and when it helps. Table 3 connects the observed patterns to broader ideas for designing learning systems.

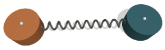
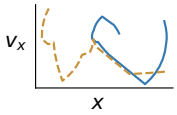
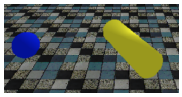
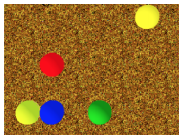
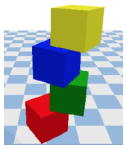
Table 3: **Findings and possible design implications.** The evidence column links each observation to its supporting experiments.

Finding	Broader implication	Evidence
• **More reliable relations** strengthen the organization of shared information.	How experiences are linked may matter alongside how much data is collected.	App. E.1
• **Sharing more properties** can make a still-shared property harder to read out.	A useful relation may be one that emphasizes the information a task needs.	App. E.1
• In SpringWorld, **alignment** drives much of the organization; **cross-prediction** adds further task gains.	Learning what should stay consistent and learning how to put it to work are distinct goals that can reinforce each other.	App. E.2
• **Changing the supplied history** redirects the same predictor along the changed physical properties.	A representation's role becomes clearer when we examine how decisions respond to it.	App. F.1; F.2
• In SpringWorld, history brings larger gains at **longer prediction horizons** under the same current-evidence budget.	Past experience may be most useful when it resolves uncertainty left by the present situation.	App. G.1
• **Direct context** works better in D-Clean; **decoded physical parameters** work better in SpringWorld.	A more interpretable representation need not be the most useful interface for a task.	App. G.2
• **Changing the readout** recovers gains from frozen SpringWorld representations, but not from every tested source.	Progress may come from finding a better way to use existing knowledge, as well as from learning new information.	App. G.4

A.4 EVALUATION SETTINGS

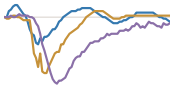
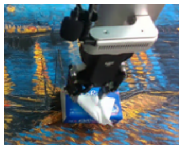
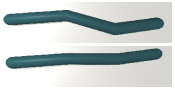
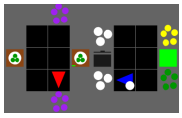
Evaluated instance identifies the learner, enabled operations and principal controls. A component study compares the interface alone (Structure), Align, Cross and their combination; application-specific choices are detailed in Appendix B.3.

Table 4: **Evaluation settings and their principal tests.** Images illustrate the settings, not model predictions.

Setting	Relation	Experience → prediction	Evaluated instance	Primary evidence
Controlled environments				
SpringWorld 	**Shared:** Mass m, drag γ, stiffness k. **Varied:** Interaction history.	**Input:** Independent history; current query/actions. **Target:** Future state increments.	**Learner:** JEPA. **Instance:** Align + Cross. **Controls:** Native; TDS. **Components:** Structure; Align; Cross.	**Primary:** Sealed new-system prediction. **Further:** Geometry; Use; Value.
PokeWorld 	**Shared:** Selected factors: m; (m, γ); (m, γ, k). **Varied:** Correct-pair fraction α, separately.	**Input:** Independent image/action history; current query. **Target:** Future observation embedding.	**Learner:** JEPA. **Instances:** Structure; Align; Cross; Align + Cross. **Controls:** Relation and objective composition.	**Primary:** Formation: relation reliability and shared factors. **Further:** Probes; Use; task readouts.
D-Clean 	**Shared:** Drag γ. **Varied:** Initial state; force sequence.	**Input:** Independent state/action history; recipient state. **Target:** Future prediction.	**Learner:** JEPA. **Source:** Align + Cross. **Readouts:** Null; Persistent; Decode; Oracle. **Comparisons:** TDS; NOD; FCRL; CaDM; DALI.	**Primary:** Formation; Value. **Further:** Closest methods.
Public benchmarks and real data				
CoPhy *Collision* 	**Shared:** Matched objects; scene-specific rule. **Varied:** Support interaction.	**Input:** Three support histories; three query frames. **Target:** Future trajectory.	**Learners:** CPC; RSSM; JEPA. **Instance:** Cross. **Controls:** Native; Structure; Random. **Extensions:** Other learners in H.1.	**Primary:** Held-out prediction. **Further:** Learner matrix.
CoPhy *Balls* 	**Shared:** Matched objects; scene-specific rule. **Varied:** Support interaction.	**Input:** Three support histories; three query frames. **Target:** Ball trajectories.	**Instances:** JEPA, CPC, RSSM: Cross; CoPhyNet: Align + Cross. **Controls:** Native; Random; Structure where tested.	**Primary:** Held-out RSSM prediction. **Further:** Memory probes and Use; learner matrix.
CoPhy *Blocktower* 	**Shared:** Matched objects; scene-specific rule. **Varied:** Support interaction.	**Input:** Three support histories; three query frames. **Target:** Multibody motion.	**Instances:** JEPA, CPC, RSSM: Cross; CoPhyNet: Align + Cross. **Controls:** Native; Random; Structure where tested.	**Primary:** Multi-body prediction. **Further:** Reader design.

Continued on the next page.

Table 4: **Evaluation settings and their principal tests (continued).**

Setting	Relation	Experience → prediction	Evaluated instance	Primary evidence
NOD *Burgers* 	**Shared:** Viscosity ν. **Varied:** Initial field.	**Input:** Independent trajectory; recipient field. **Target:** 1D field evolution.	**Learner:** Released NOD. **Instance:** Relation-trained stage; prediction-only continuation. **Controls:** NOD; Random; continued NOD; GEPS/CoDA (50 adaptation steps).	**Primary:** Public operator prediction. **Further:** Viscosity probes; fixed-donor use.
NOD *FHN* 	**Shared:** Reaction parameters (k, β). **Varied:** Initial field.	**Input:** Independent field histories; recipient field. **Target:** 2D field evolution.	**Learner:** NOD-Hier. **Instance:** Align; prediction-only adaptation. **Comparison:** Equal-history NOD.	**Primary:** 2D prediction; history aggregation. **Further:** Factor probes.
Baxter 	**Shared:** Hardness or shape. **Varied:** The other factor.	**Input:** Separate tactile grasp. **Target:** Future tactile representation.	**Learner:** Predictive encoder. **Instance:** Align + Cross. **Control:** Random relation.	**Primary:** Tactile factor selectivity. **Further:** Hardness and shape readouts.
RH20T 	**Shared:** Task identity. **Varied:** Recorded episode.	**Input:** Multimodal history; current inputs/actions. **Target:** Force/torque; TCP motion.	**Learner:** Supervised predictor. **Instances:** Cross; Align + Cross. **Controls:** Structure; pairing; inputs.	**Primary:** Robot forecasting and task-specific donor use.
Swimmer 	**Recipient:** Fixed system. **Donor:** Own-system or wrong-system update.	**Input:** Persistent update; current query/action. **Target:** Prediction through fixed weights.	**Learners:** Frozen JEPA/GRU. **Intervention:** Update substitution. **Training:** No new Align/Cross source fit.	**Primary:** Fixed-update use.
Overcooked 	**Shared:** Fixed partner identity. **Varied:** Disjoint episode.	**Input:** Prior episodes; current query. **Target:** Ego-action prediction.	**Learner:** AD action learner. **Instance:** Align only. **Control:** Matched history slot; variance/covariance terms.	**Primary:** Partner-behavior readout. **Further:** Cooperation across history budgets.
Pendulum 	**Shared:** System dynamics. **Varied:** Episodes; mass and length across systems.	**Input:** Transition history; current state. **Target:** Dynamics for online control.	**Learner:** CaDM interface. **Instance:** R7 relation recipe. **Controls:** CaDM; Random relation.	**Primary:** Return and success. **Further:** Same-checkpoint factor probes and Use.

How the paired gains are calculated. Figure 6 selects positive examples pairing task and property measurements under the same model comparison and source set. Error/cost gains are $100(1 - E_{\mathrm{SPRII}}/E_{\mathrm{ref}})$; probe R^2 gains are $100(R^2_{\mathrm{SPRII}} - R^2_{\mathrm{ref}})/(1 - R^2_{\mathrm{ref}})$; Overcooked return gains are $100(R_{\mathrm{SPRII}} - R_{\mathrm{ref}})/R_{\mathrm{ref}}$, using a positive reference. Reported measurements are averaged across sources before taking ratios. These examples include development results and do not estimate average suite-wide benefit.

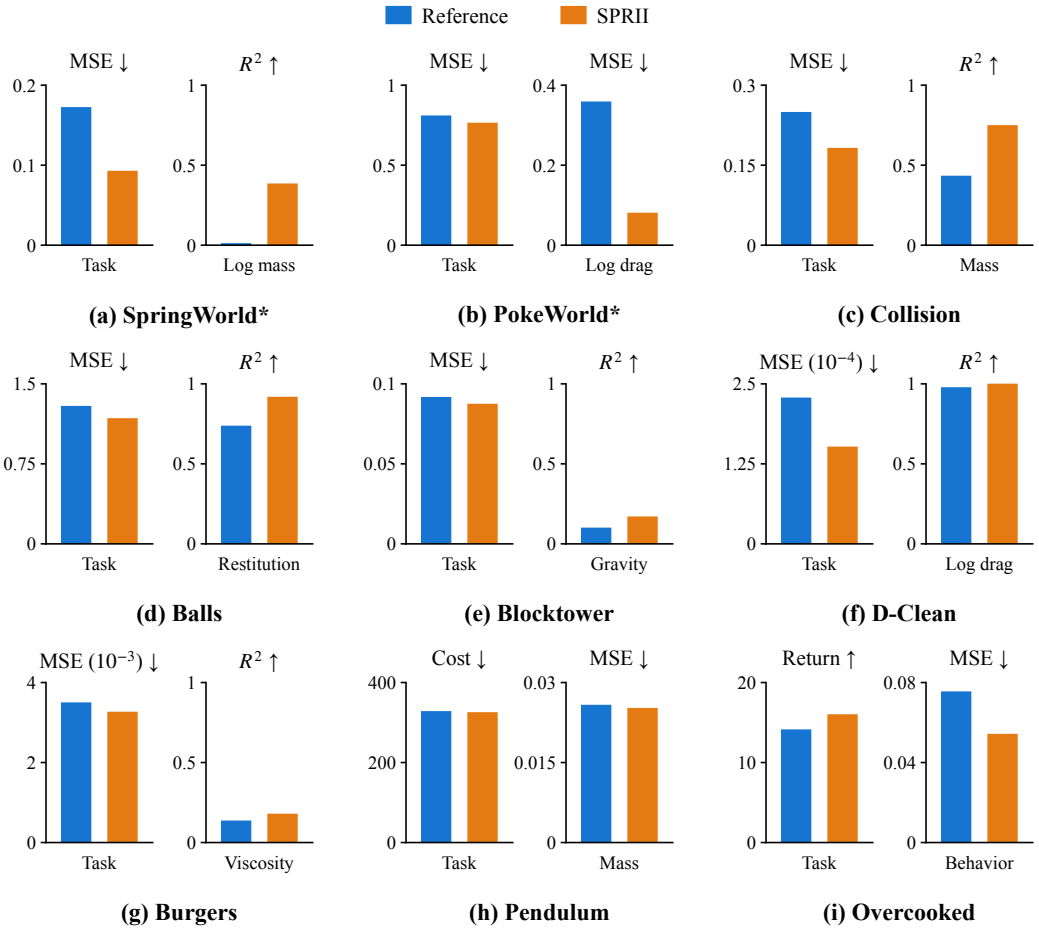


Figure 7: **Original-value task and property comparisons.** Each setting pairs task performance (left) with property readout (right). Blue: reference; orange: SPRII. Arrows mark the preferred direction.

Comparisons. (a) Native JEPA → Align + Cross; (b) random → correct pairing; (c) Native CPC → Cross; (d) Native RSSM → Cross; (e) Native JEPA → Cross; (f) CaDM → SPRII; (g) Plain CoDA → Cross; (h) CaDM → SPRII R7; (i) H+VC → Align.

Task metrics. (a) Cold-start MSE; (b) h16 state MSE; (c–e) trajectory MSE; (f) h32 MSE; (g) ID MSE; (h) ID control cost; (i) familiar-partner return. Collision and Blocktower probes read support S; Balls reads memory M; Blocktower's property is vertical gravity.

Sources and aggregation. (a) Table 36: source 0, reader 0; (b) App. G.4: source 0, three readers; (c,e) Table 62: two sources; (d) App. F.4: three sources; (f) App. C.3: three sources, three readers each; (g) Tables 18, 21: three sources, code-0; (h) App. G.5: three training seeds; (i) Table 67: three seeds, full history.

*SpringWorld is the source-0 development example, distinct from the sealed multi-source result. PokeWorld selects the largest task improvement among sources 0/1/2 ($+5.6\%$, $+1.4\%$, -15.0%), using the same source for the drag probe and 5,000 reader updates. CoPhy panels use paired development records; other panels use the stated means. The Balls memory probe includes the fitted reader projection. Overcooked reports net cooperation return including action costs, and partner behavior rather than a physical parameter. `SPRII_PAIRED_GAIN_SOURCES_20260926.zip` retains the inputs, precision and calculations.

B METHOD AND EXPERIMENTAL IMPLEMENTATION

This section specifies how the instances in Table 4 are built and measured. We first give the common relation objectives and training update, then the controlled JEPA implementation and learner-specific interfaces. Appendix B.4 defines the environments and data splits; Appendices B.5–B.6 define the probes, readers, interventions and aggregation used in the results.

B.1 RELATIONS, OBJECTIVES AND TRAINING UPDATE

The relation $\mathcal{R}_G$ specifies what histories share; the kernel $Q_G(A \mid B)$ specifies how a legal donor is selected. Drawing a recipient from P induces the pair law

$$\Pi_G(A, B) = P(B)Q_G(A \mid B). \tag{5}$$

Relation semantics and sampling are separate choices: $\mathcal{R}_G$ defines the shared conditions, whereas Q_G selects actual legal pairs. Our main setting pairs separately realized interactions. The formulation also admits broader pairing constructions, such as overlapping clips, whose shared episode structure offers different opportunities to rely on local cues. Partial-factor sharing is distinct from this sampling choice: two separately realized interactions may share selected factors while others differ. The supports of their pairing kernels consequently need not be nested, even when the shared-factor definitions are nested.

The factorized PokeWorld experiment uses

$$G_1 = m, \qquad G_2 = (m, \gamma), \qquad G_3 = (m, \gamma, k). \tag{6}$$

The first kernel changes drag and stiffness, the second changes stiffness, and the third changes the rollout while preserving the complete system. Baxter's hardness and shape relations provide a real-data counterpart. The network receives observations and task inputs; grouping metadata is used to construct the pairs. Parameter regression probes are fitted outside representation training.

Objectives and gradients. For paired code matrices $A, B \in \mathbb{R}^{n\times d}$, the alignment loss is

$$I(A, B) = \frac{1}{nd}\|A - B\|_F^2, \tag{7}$$

$$V(A, B) = \frac{1}{2d}\sum_{j=1}^{d}\left[\max\{0, 1 - \sqrt{\mathrm{Var}(A_{:j}) + \epsilon}\} + \max\{0, 1 - \sqrt{\mathrm{Var}(B_{:j}) + \epsilon}\}\right], \tag{8}$$

$$C(A, B) = \frac{1}{d}\sum_{i\neq j}\left[\mathrm{Cov}(A)_{ij}^2 + \mathrm{Cov}(B)_{ij}^2\right], \tag{9}$$

$$\mathcal{L}_{\text{align}} = 25I + 25V + C, \qquad \epsilon = 10^{-4}. \tag{10}$$

Variances and covariances use the sample denominator $n - 1$. Related branches are evaluated jointly for this loss; no stop-gradient is inserted in Align. The base learner determines target-encoder gradients for its prediction loss. In the controlled D-Clean/PokeWorld models, both prediction and target embeddings are produced by the online observation encoder.

The loss weights and pairing kernels are part of each instantiated training recipe. For the controlled factorial, the predictor and representation dimensions are shared across Structure, Align, Cross, and Align + Cross. For transferred learners, the native comparator and the structure-matched comparator answer complementary questions: the total effect of the adaptation and the increment from its relation term. Configuration choices and common evaluation settings are reported with each environment. Each application uses a fixed configuration or validation-selected recipe. Appendix B.6 specifies the checkpoint and selection rules; the profiles below state the enabled operations. History codes are substituted rather than added; fixed-predictor evaluations make no gradient updates.

Algorithm 1 retains the application's native loss and regularizers. Its task loss, readout, normalization and optimizer are application-specific. Disabled relation terms are zero; a legal donor obeys the temporal and input constraints and leaves the recipient target unchanged.

Algorithm 1 One SPRII training update.

Inputs: recipient sampler P, donor kernel Q_G, encoders and task readout q.

1 **Sample** recipients and related donor histories:
$B \sim P, \qquad H_A \sim Q_G(A \mid B).$

2 **Encode** the histories and current recipient input:
$z_{\mathrm{p}}^A = g_{\mathrm{p}}(H_A), \qquad z_{\mathrm{p}}^B = g_{\mathrm{p}}(H_B), \qquad z_{\mathrm{s}}^B = g_{\mathrm{s}}(C_B).$

3 **Compute** the native objective $\mathcal{L}_{\mathrm{base}}$ on ordinary recipient inputs.

4 **Initialize** $\mathcal{L}_{\mathrm{align}} \leftarrow 0$, $\mathcal{L}_{\mathrm{cross}} \leftarrow 0$.

5 **if Align is enabled:**
$\mathcal{L}_{\mathrm{align}} \leftarrow \mathrm{VICReg}(Z_{\mathrm{p}}^A, Z_{\mathrm{p}}^B).$

6 **if Cross is enabled:**
$\hat{y}_B^{A\to B} \leftarrow q(z_{\mathrm{s}}^B, z_{\mathrm{p}}^A, u_B).$
$\mathcal{L}_{\mathrm{cross}} \leftarrow \mathrm{mean}\ \ell_{\mathrm{task}}(\hat{y}_B^{A\to B}, y_B).$

7 **Combine** the native and enabled relation objectives:
$\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{base}} + \lambda_{\mathrm{p}}\mathcal{L}_{\mathrm{align}} + \lambda_{\mathrm{x}}\mathcal{L}_{\mathrm{cross}}.$

8 **Backpropagate** $\mathcal{L}$ and apply one optimizer update.

B.2 Shared controlled-JEPA implementation

Figure 8 details the controlled JEPA instantiation of the general interface. The controlled D-Clean and PokeWorld models use histories of 24 observations and predict encoded targets at horizons $\{1, 4, 16\}$. In split variants, $z_{\mathrm{s}} \in \mathbb{R}^{64}$ is an MLP of the last observation embedding. The persistent encoder concatenates encoded observations with 32-dimensional action features. A leading zero action aligns each later observation with its preceding action; learned position embeddings are added before processing the sequence with a four-layer causal Transformer of width 192, eight attention heads, and four-times-width feed-forward blocks, producing $z_{\mathrm{p}} \in \mathbb{R}^{64}$. The predictor receives a 128-dimensional context, padded future actions and masks, and a 32-dimensional horizon embedding; its two hidden layers have width 256. The monolithic Native replaces the split encoders with a 128-dimensional history context while retaining the predictor capacity.

D-Clean observations are 4D states encoded by a three-layer MLP. PokeWorld renders a 64×64 current frame and temporal-difference channel. Its visual encoder uses four stride-2 convolutional blocks with channels $32, 64, 128, 256$, followed by a linear projection to 128 dimensions and batch normalization. Models train for 20,000 steps with AdamW, learning rate 3×10^{-4}, weight decay 0.05, 500 warmup steps followed by cosine decay, bfloat16 autocasting, and gradient clipping at 1.0. A paired batch contains 48 systems and 96 windows ordered as donor/query halves.

SIGReg uses 17 quadrature knots and 1,024 random directions, with $\lambda_{\mathrm{reg}} = 0.02$ (Balestriero & LeCun, 2025). Persistence uses VICReg weights $25/25/1$ for invariance, variance, and covariance, unbiased per-branch variance, and $\epsilon = 10^{-4}$ (Bardes et al., 2022). The controlled objective weights $\lambda_{\mathrm{p}} = 1.0$ and $\lambda_{\mathrm{x}} = 0.1$ were frozen before the three-source factorial. Rel-InfoNCE uses the InfoNCE objective lineage of contrastive predictive coding and the two-view group-positive construction of supervised contrastive learning (van den Oord et al., 2018; Khosla et al., 2020), with the separate three-seed validation rule in Appendix E.1.

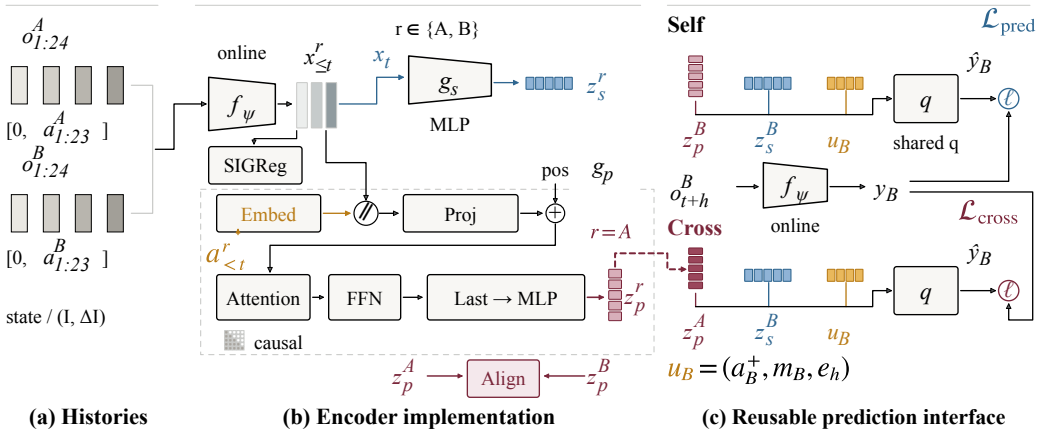


Figure 8: **Controlled-JEPA implementation of SPRII.** (a) Related observation/action histories. (b) A shared observation encoder feeds separate current and persistent branches; Align compares the persistent codes. (c) Self and Cross share the predictor and recipient target. Cross replaces only the persistent input with the donor's code. Self-prediction and SIGReg together form the native objective in Figure 2. Tokens and the causal-mask crop are schematic.

B.3 Cross-learner instantiation

The persistent–current interface is a structural inductive bias rather than a shared network architecture. A history encoder or an existing context state supplies the persistent route; recipient-specific inputs remain available to the native predictor. Table 5 summarizes each learner's native predictive route, measured representation and enabled relation operations.

The controlled family tests both operations separately and jointly. The CoPhy source-0 cross-scene development matrix uses one recipe per learner family. The separately validation-selected Collision JEPA recipe is specified in Appendix B.4. Overcooked's native action objective already consumes previous episodes, so its added operation is Align; NOD likewise retains its trajectory-decoupled prediction objective. Its alignment phase and prediction-only continuation are specified in Appendix C.4. Structure isolates the added interface and Random holds the operation fixed while changing the relation.

B.4 Data, splits and admissible inputs

Each profile specifies the observations, target, relation and data split. Table 4 links to these profiles; Table 7 separates the evaluation populations and consuming routes. CoPhy's three scenes and NOD's two field systems count separately among the thirteen settings.

SpringWorld

Environment and information. SpringWorld is a controlled visual elastic-coupling environment in which mass m, drag γ, and stiffness k persist across independent interactions. The main ranges are $m \in [0.5, 2]$ kg, $\gamma \in [0.25, 1.5]\,\mathrm{s}^{-1}$, and $k \in [4, 25]$ N/m. Observations comprise 128×128 grayscale images and causal frame differences. A 96-frame donor spans 4.75 seconds; the first frame difference is zero. Targets are eight-dimensional position/velocity increments standardized using training statistics. The cold-start query has no preceding motion history, and prediction covers the next 0.8 seconds. Physical labels are used for evaluation and downstream supervised heads.

The development readout protocol uses 144 training systems and 178 validation systems: 64 continuous new systems, 36 unseen discrete combinations, 30 new interpolated levels, and 48 single-factor extrapolation systems. The original development cold-start panel uses the first two groups, 100 systems and 600 cases. The full mixture uses 16,020 cases across the four groups and its fixed query conditions. They are distinct evaluation profiles rather than different training-set sizes.

Table 5: **Predictive routes and relation operations across learners.** U denotes complete source state, S independent support and M trained reader memory; each is probed separately.

Instance	Native predictive route and measured representation	Relation operations
• **Controlled JEPA**	24-step history P_{64}, current S_{64}; future online embedding MSE + SIGReg. D-Clean state MLP; Poke visual CNN.	Structure, Align, Cross, Align + Cross.
• **Spring JEPA**	96-frame visual/action history; embedding MSE + SIGReg; frozen source and separately fitted physical readers.	Align + Cross, with component controls.
• **CoPhy JEPA/CPC**	Frozen 784D perception features; latent prediction or future contrast; native dynamics state retained.	Cross in the main matrix.
• **CoPhy RSSM**	Reconstruction, prior prediction and KL; $U = [P_{64}, h_{128}, z_{32}]$, versus 160D Native U.	Cross; U, S, M measured separately.
• **CoPhyNet**	Coordinate MSE and stability classification; original 32D code split as P_{16}, T_{16}; reader receives full U.	Align + Cross on P.
• **Baxter**	40-step tactile history, P_{64}, S_{64}; future embedding MSE + SIGReg.	Align + Cross; hardness/shape relations.
• **RH20T**	Fused visual/sensor history; joint latent, FT and TCP MSE + SIGReg.	Cross / Align + Cross.
• **Overcooked AD**	Ego-action cross-entropy; disjoint prior episodes pooled in a 32D slot.	Align only; $\lambda_x = 0$.
• **NOD Burgers/FHN**	Released trajectory-decoupled/hierarchical field operators; native conditional prediction retained.	Align; prediction-only continuation recorded separately.
• **Amortized CoDA**	History-inferred context through a frozen CoDA decoder; native field prediction.	Cross; four-arm protocol in Appendix C.5.
• **Pendulum / CaDM**	Dynamics context supplied through the existing control interface; return, success and same-checkpoint probes.	Align (R7); fixed recipe in Appendix G.5.
• **Swimmer**	Recipient-specific persistent update through frozen JEPA/GRU predictors.	Fixed-input substitution; no new Align/Cross source training.

Base learner and adaptation. The JEPA reference is a monolithic predictor; its SPRII instance introduces separate history and current-context branches and adds Align with weight 1 and Cross with weight 0.1. The two use the same data bank, 10,000 source updates, 48 pairs per batch, AdamW learning rate 3×10^{-4}, and common regularization and schedule. The final-step checkpoint is used. Structure, Align, and Cross isolate the interface and individual operations.

All source representations are frozen before the common readout is trained for 10,000 updates with batch 256 and the same head seed. The readout has a 128-dimensional context interface, 64-dimensional slot, and two 256-unit hidden layers. Supervised GRU, Transformer, TCN, and DeepSets encoders provide strong reference representations under their own training recipes; they are reported alongside the matched JEPA comparisons.

Observation normalization uses a fixed pre-update statistics snapshot at each source-training update; training-history features update that snapshot only afterward. The representation is frozen for downstream readers and probes. The 96-frame donor protocol is specific to SpringWorld, not the 24-frame controlled D-Clean/PokeWorld input.

D-Clean

D-Clean is an analytically integrated two-dimensional inertial system, $\dot{v} = F/m - \gamma v$, with $m = 1$ and $\Delta t = 0.05$. A system fixes drag and varies initial state and force segments across eight independent 64-step interactions. The train/validation/test split contains 1,000/200/200 disjoint systems. The 4D state histories provide a clean setting for measuring factor accessibility and prediction separately. The history, rather than a single current state alone, supplies evidence about drag. The original source comparison is retained in Table 30; the state-space baseline and same-donor physical bottleneck use their own downstream populations.

Independent-method comparison. NOD uses its released MassSpring network components with D-Clean action inputs; CaDM uses a PyTorch implementation of its released model components, and FCRL is implemented from the paper. Each representation is frozen before fitting an identical downstream reader. Each method uses three source seeds and three reader seeds per source, with 20,000 source updates and 20,000 reader updates. Evaluation uses 100 systems from the existing validation report population, raw-state MSE and horizons $1, 4, 16, 32$. Readers are averaged within source before computing the three-source mean and sample SD. All source fits are included, including NOD source 1, whose code is constant. This comparison measures predictive value under the common reader; the Burgers comparison uses NOD's native field predictor. The corresponding source probe and fixed-reader donor interventions use the same fitted models.

PokeWorld

PokeWorld renders an actuated finger interacting with an object, with mass, drag and contact stiffness fixed within each system. Observations are current images and frame differences, both 64×64; actions are available to the history encoder and predictor. Four independent 64-step rollouts vary initial states and action modes. The original bank contains 2,000/400/400 disjoint train/validation/test systems.

The factorized bank contains $10 \times 10 \times 28 = 2,800$ physical tuples. A frozen allocation assigns 20/4/4 stiffness tuples in each mass–drag block to train/validation/confirmation, producing 2,000/400/400 systems. Exact tuples are disjoint across splits, while factor levels occur in each split.

Relations $G_1 = m$, $G_2 = (m, \gamma)$ and $G_3 = (m, \gamma, k)$ define which factors are shared. Training samples one relation-eligible donor per query and update. For G_3, the eligible histories are the other three rollouts of its system. The relation-eligible pool and the finite candidates retained by a sampler are different objects; Appendix E.1 reports both and describes the candidate-selection rule. Random uses balanced system derangements. The fidelity intervention fixes R8 and mixes correct and random relations at $\alpha \in \{0, 0.5, 1\}$ while preserving donor/query marginals. All model settings remain common within each intervention. The original acquisition bank and the later relation-control banks are separate protocols; Appendix E retains their distinct measurements.

CoPhy

Environment and task. CoPhy provides visually observed systems whose physical properties govern different initial conditions and interactions (Baradel et al., 2020). We treat its three scene families as distinct environments in the suite:

Balls. Balls emphasizes rebound dynamics and physical-factor readout from ball trajectories.

Collision. Collision emphasizes contact-dependent motion and is the held-out task used for the main prediction comparison.

Blocktower. Blocktower emphasizes multibody stability, falling, and breadth across a different interaction geometry.

Shared protocol. The three scenes share a visual front end and episode construction. We construct cross-experience prediction episodes with three independent support histories (S3) and the first three query frames (query3). Prediction starts at query frame four, for 27 frames in Balls/Blocktower and 12 in Collision. Complete validation contains 2,000 Balls, 4,000 Collision, and 8,088 Blocktower episodes. The original CoPhy task and this multi-support prediction task have different input and target intervals; the reported matrix uses the latter.

Preserved learners and common evaluation. The official supervised perception frontend is frozen. JEPA, CPC, and RSSM learn dynamics from its 784-dimensional visual features without physical labels or coordinate targets. CPC retains conditional future discrimination; RSSM retains a deterministic state, stochastic latent, conditional prior and posterior, reconstruction, and KL regularization (Hafner et al., 2019). Its Native model already trains multistep prior prediction. Source future prediction never reads the future posterior. Supervised CoPhyNet retains its original source-training objective.

For JEPA/CPC/RSSM, Native preserves the corresponding base architecture; Structure adds the persistent–current interface; SPRII adds Cross; Random keeps the same Cross objective with randomized relations. The supervised CoPhyNet SPRII instance instead adds both Align and Cross; its Random control randomizes the relation under that combined objective. Collision uses all-object persistent substitution in the reported source run. Native representation dimensions are preserved rather than compressed to a common persistent-only code. Sources are frozen before a common newly trained prediction head receives the complete legal source state and S3 history. For example, structured RSSM exports $U = [P_{64}, h_{128}, z_{32}]$ with 224 dimensions, whereas Native exports its 160-dimensional recurrent state. The supervised split CoPhyNet code is $U = [P_{16}, T_{16}]$. The training-time persistent branch and the complete state exposed to a head or probe are therefore distinct interfaces.

Every main head is trained for 100 epochs. The source budget is 100 epochs for CPC and RSSM. In the held-out Collision comparison, all four JEPA arms use 50 source epochs, matching the selected J2 source budget; the earlier unequal-budget control rows are not used in this comparison. The JEPA comparison has four arms and three joint source/reader seeds on the same 1,994 held-out Collision episodes. These fits extend evaluation of the existing test cohort. All conditions use common queries, targets and visible-object eligibility within each reported population. Balls/Collision use direct query-pose perception; Blocktower uses its cached feature and matching pose-head route. These paths are held fixed within scene.

Operation-specific source paths. CPC matches contrastive candidates by public time, object slot and visible type. RSSM filters the first three query frames with zero persistent input, preventing initialization from bypassing the tested history path; Cross predicts through its prior rollout. The supervised CoPhyNet route retains the original 32-dimensional AB code, split into P_{16} and T_{16}. Align acts on P_{16}; Cross replaces that part while keeping the recipient's remaining code and initial query state. The common frozen-source reader uses the full 32-dimensional representation. These implementations preserve native objectives without imposing a common code.

Baxter tactile grasps

We use the public Baxter tactile hardness bank of Amin et al. (2023) from the dataset release of Amin et al. (2026). Each grasp peak is an 80×16 sequence. The primary crossed bank retains cube and cylinder objects at the three hardness levels shared by both shapes, yielding six known configurations and 170 grasps per configuration. A single permutation of grasp identifiers (Table 8) is reused across configurations and split 110/30/30 into train/validation/confirmation interactions. The 40-sample files are alternate processed windows from the same acquisitions. The available input consists of grasp-peak tactile histories; pre-contact frames and action sequences are unavailable.

For $\mathcal{R}_H$, a donor shares the hardness level and changes shape; for $\mathcal{R}_S$, it shares shape and changes hardness level. Random uses a uniform derangement among the other five configurations. Every batch contains eight queries from each configuration and exactly one different-grasp donor per query, preserving balanced query and donor marginals.

The split architecture uses a per-sample MLP, a four-layer causal Transformer persistent encoder, 64-dimensional current-context and persistent codes, and prediction at $+1, +20, +40$ samples. All conditions use 10,000 updates, 48 query–donor pairs per batch, and seeds 0/1/2. Six randomly initialized supervised input-readout fits (early 8 samples versus full 80 samples) and nine world-model fits are frozen before the single confirmation evaluation. The source input is a 16-dimensional tactile sample and the causal history contains 40 steps. The reported source recipe enables Align + Cross for both partial-factor relations. Its six-configuration visualization and complete selectivity results appear in Appendix E.3.

OvercookedV2

Environment and base learner. OvercookedV2 is a cooperative cooking environment with coordination-dependent reward (Gessler et al., 2025; Rutherford et al., 2024). We use its native multi-episode AD setting (Jing et al., 2026): the controlled entity is a fixed neural partner whose behavior persists while locations, actions, and outcomes change across episodes. The training bank contains 2,560 histories and 37,376,000 transitions from 20 partners drawn from ten partner-training

sources. Nested half- and quarter-history subsets retain 1,280 and 640 histories, or 18,688,000 and 9,344,000 transitions, with 64 and 32 rather than 128 histories per partner. Each history contains 146 recorded 100-step episode prefixes. The layout is `grounded_coord_simple`. Two additional partners from one excluded training source form the held-out development cohort; they are absent from policy training. Relation metadata identifies the partner for sampling without exposing its network parameters to the learner.

Matched adaptation. The AD action cross-entropy already consumes prior interactions. The H condition adds a 32-dimensional persistent history slot; H+VC adds variance/covariance regularization; H+SPRII additionally aligns histories of the same partner. The matched conditions share architecture, initialization and ordered training batches within a seed. The action objective reads the persistent slot, so SPRII uses Align without an additional Cross loss. The primary comparison is H+SPRII versus H+VC at full, half and quarter history budgets. Each pair uses 20,000 updates, effective batch 1,024, microbatch 128, learning rate 3×10^{-4}, and training seeds 4200, 4201 and 4202. The losses are $L_{\mathrm{CE}} + 0.001(25V + C)$ and $L_{\mathrm{CE}} + 0.001(25I + 25V + C)$, respectively. Both retain two 100-step support episodes and a 300-step query. All 18 fitted models use their fixed 20,000-update endpoints. The history-budget intervention subsamples this fixed offline bank; partner training and collection costs lie outside this budget.

Evaluation and replication. Each frozen endpoint is evaluated for 20 episodes with each of the 20 familiar and two held-out development partners: 440 episodes per model, with no optimizer updates. The primary return averages episodes 6–20 within partner, then averages partners within each cohort. Return and probe summaries report the mean and sample SD over the three training seeds. Formation and functional probes use the same frozen endpoints and common sample plan; the Random condition has three probe seeds and no return evaluation. The matched/null/wrong-history donor intervention uses one frozen checkpoint.

History interface. The dynamic path retains the AD observation encoder, action/reward tokens and causal Transformer. Each disjoint support episode is processed separately by the shared backbone; its final valid token projects to a 32-dimensional persistent slot. Attention pooling combines available slots, and a fusion MLP reads that result with the recipient's causal query representation to predict the ego action. H, H+VC and H+SPRII share this interface. Their common action loss already consumes earlier episodes; the tested increment is alignment.

RH20T

RH20T records multimodal robot manipulation (Fang et al., 2023). The operative relation groups different episodes by task identity. It supplies repeatable task context, without assuming known physical-parameter equivalence. Models predict normalized force/torque and TCP motion from synchronized visual, robot-state and action inputs. The representation and prediction heads are trained jointly; this application uses supervised prediction targets.

Matched conditions. The complete matrix contains 14 conditions and three seeds each. All share prediction windows, normalization, targets and checkpoint step. "Indep" denotes a matched different-episode donor, "SameEp" a causal non-overlapping donor from the query episode, and "Random" balanced random training groups. LowDim is a distinct input/representation route whose controls are compared within that route. Evaluation keeps the query, target, horizon and actions fixed while changing self, matched and shuffled-task context. Task-macro results aggregate 25 held-out tasks.

Visual and low-dimensional sensor embeddings are fused before the causal history encoder. A common prediction trunk feeds latent, six-dimensional force/torque and three-dimensional TCP heads, all trained jointly. Cross changes the persistent input while retaining current context and task conditioning. The 25-task forecasting results are in Appendix F.3; the 24-task hard-negative bank is specified separately in Appendix F.3.

Articulated Swimmer

Articulated Swimmer is a three-link MuJoCo system with three link masses and two joint damping parameters. The task predicts state responses to actions. Each frozen architecture retains its own

training-only normalization. Appendix F.5 compares own- and wrong-system persistent updates within each architecture, keeping the source and predictor weights fixed.

Pendulum control

Pendulum tests whether the learned dynamics context supports an online controller when mass and length vary between systems. The development-selected R7 relation recipe uses weight 0.003 and variance target 0.1 with the existing CaDM control interface. It is evaluated intact on ID and four OOD groups, with three training seeds and ten 200-step episodes per group. Return, unchanged upright-success criteria and same-checkpoint physical probes are reported together in Appendix G.5; logged-action donor interventions are not presented as online controller interventions.

NOD-Burgers: one-dimensional field evolution

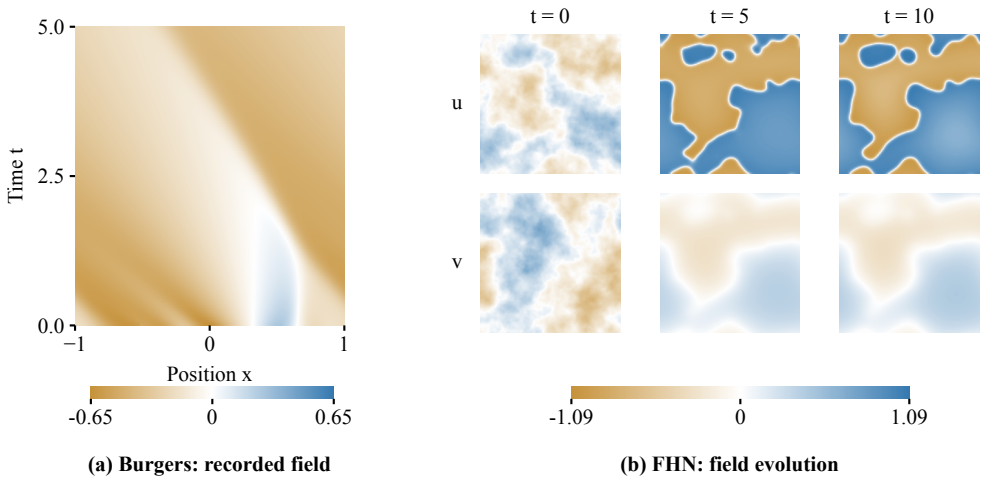
Burgers uses a spatial field rather than a rigid-body scene. Independent trajectories share viscosity ν while differing in initial conditions; the recipient field provides the query state. The released trajectory shown in Figure 9(a) has 401 spatial samples and 101 saved times. Native prediction, viscosity accessibility and donor-use evidence, including three-seed alignment and continuation comparisons, are specified in Appendix C.4.

NOD-FHN: two-dimensional reaction–diffusion

The FitzHugh–Nagumo (FHN/DR2D) environment evolves two coupled fields u, v on a periodic 128×128 grid. Related trajectories share reaction parameters (k, β) with independent Gaussian-random-field initial conditions. The Python port follows the released fourth-order spatial stencil and RK4 update ($\Delta t = 0.001$), saving 101 frames through $t = 10$. Conditioning uses frames $0, 25, 50, 75, 100$ from an independent trajectory; the hierarchical operator predicts the recipient's field evolution. The missing original initial-condition bank was regenerated, so these are reproducible Python simulations, not the authors' exact data.

Training schedules and selection. Both methods use batch size 8 and RMSprop, with initial learning rate $0.5/\sqrt{P}$, where P is the number of trainable parameters, and a factor of 0.95 decay every 500 updates. NOD uses prediction-only stages ending at 20,000, 30,000, 40,000 and 45,000 updates. SPRII uses alignment weight 0.01 with a 2,000-update linear warmup through update 10,000, followed by prediction-only stages ending at 30,000, 40,000 and 50,000. Both recipes reset the learning rate to 10^{-5} after update 40,000 while retaining RMSprop moments and scheduler state. Each continuation restarts its seeded sampling stream.

Seed 42 selects the recipes using mean ID relative-L_2 error across H1/H5/H50 on initial condition 5. Seeds 43/44 repeat the selected stages and endpoints without further checkpoint selection; reporting uses initial conditions 15/24/45. The NOD and SPRII recipes therefore have different update budgets. Appendix H.3 reports the three-source comparison and conditioning-history experiments.



(a) Burgers: recorded field **(b) FHN: field evolution**

Figure 9: **External field environments.** (a) A released Burgers training trajectory at $\nu = 0.01$. (b) A separately simulated FHN trajectory at $k = 0.03, \beta = 0.20$. FHN panels share a signed field scale; Burgers uses a separate scale. These are environment illustrations, not prediction results.

B.5 Probes, fresh readers and fixed consuming routes

A *complete-method comparison* changes the trained source recipe; a *fresh-reader comparison* fits alternative task readers on a frozen source; a *fixed-route intervention* changes context through the same fitted weights. They answer different questions even when the same source model appears in all three. The measurement definitions below complement the task-specific controls in Section 3.

Table 6: Terminology used in the main text and appendix.

Term	Meaning
Persistent context	History-derived representation supplied to a predictor; source weights are frozen for the stated diagnostic evaluations.
Probe	Supervised diagnostic fitted to a frozen code; measures accessibility.
Task reader	Predictor fitted on fixed source features; measures realized task value.
Fixed-predictor intervention	Donor replacement with all source and consuming weights fixed.
Native	History-conditioned learner with its original context interface and objective.
Structure	Separate persistent/current interface, without relation objectives.
Random	The instantiated relation objective with a wrong-donor pairing control.

Input evidence and representation accessibility. An estimator trained on the model-visible history checks whether factor evidence is recoverable from the input; a privileged-input estimator is a separate reference. A representation probe fits a supervised factor readout of a frozen representation, using the training systems for fitting. Each result identifies its input: code centroids, actual donor windows, complete source state U, independent-support code S, or learned reader memory M. Probes measure accessibility, while cross-history geometry measures organization; both inform Formation. A memory probe includes the trained reader projection, while a source probe reads the frozen encoder alone.

Fresh task readers and fixed-route use. Null, Persistent, Decode and Oracle fit separate task readers with their stated inputs; Oracle supplies true parameters to a learned reader, not an optimal-risk bound. A fixed Wrong intervention substitutes context through an unchanged reader, keeping source, predictor and any output decoder fixed. The resulting output or risk change measures Use under that substitution. Fresh-reader task errors measure realized Value under the compared inputs and fitting recipe. A separately trained Shuffled/Wrong arm differs from fixed-input substitution.

Frozen-source realization routes. Let h denote the frozen recipient feature, p the frozen persistent code, and $R_0(h)$ the separately fitted, then frozen recipient-only base reader. The 128-unit residual

reader computes $v = \mathrm{GELU}(W_v h)$ and $q = W_q h + b_q + W_{\mathrm{proxy}} h$. With elementwise product $\odot$,

$$M_1(h) = R_0(h) + W_o[v \odot \tanh(q)], \tag{11}$$

$$M_2(h, p) = R_0(h) + W_o[v \odot \{\tanh(q + W_p p) - \tanh(q)\}]. \tag{12}$$

The persistent projection and output projection are bias-free, so $M_2(h, 0) = R_0(h)$ by construction (up to floating-point evaluation order). A zero-context penalty therefore cannot alone identify donor specificity. The query-only arm registers the same persistent projection but does not use it or update it by gradients: the architecture is shared, active capacity is not exactly matched. The Oracle route replaces the learned code with a projection of true parameters and uses the same residual family.

The `phys` variants multiply the residual output coordinatewise by a fixed, nonnegative weight vector derived from the prescribed physical sensitivity procedure. They do not add a physical loss or supply true parameters to M_2. Source encoders and the 256-unit base are frozen during residual fitting. Reader fitting uses MSE, AdamW, gradient clipping at 1, and nominal 1,000-step base and route budgets with batch size 128 and learning rate 10^{-3}; development error selects the checkpoint. The physical target and evaluation population are held fixed within each pair.

The results of these routes appear in Appendix G.4. We evaluate $M_2(h, sp)$, using s for the post-fitting context amplitude and reserving α for training-pair reliability. The zero-amplitude identity is $M_2(h, 0) = R_0(h)$, not $M_1(h)$.

B.6 Measurement, aggregation and selection

Geometry and factor accessibility. Controlled codes are standardized using training statistics. Within-system distances measure stability across interactions; between-system distances measure separation. The reported between/within ratio is averaged at the source level, not computed from pooled distances. The Balls memory assay uses the inverse within/between ratio and labels it explicitly. Partial factor geometry correlates representation distance with factor distance while controlling the stated other factors; it is distinct from supervised-probe R^2. Probe fitting and selection use training systems. Transient position and velocity targets use regression, while contact uses AUROC.

Prediction targets and scales. The controlled source objective predicts each learner's own observation embedding, so self losses characterize native training. Physical-space comparisons use downstream targets: SpringWorld readers predict train-standardized eight-state increments; D-Clean readers in the state-space and same-donor studies report raw-state MSE on different populations. RH20T uses normalized force/torque MSE and TCP error in cm.

Evaluation populations. We use *sealed test* for the separately held-back evaluations identified as such in the experiment records, *held-out official episodes* for the CoPhy test cohort, and *confirmation* for the frozen relation-design evaluations. Development results retain their own populations and selection rules. SpringWorld's 100-system common-adapter cold-start comparison and its 178-system cold-query horizon curve are different evaluations: their h16 history gains are 36.10% and 35.68%, respectively. The sealed method comparison uses a separate 256-system population.

Table 7 lists the distinct evaluation populations and consuming routes used by the functional analyses.

Source and reader dispersion. For cell error E_{sr}, the mean first averages R_s fitted readers within source, $\bar{E}_s = R_s^{-1} \sum_r E_{sr}$, then averages S sources equally. The displayed source SD is

$$s_{\mathrm{source}} = \sqrt{\frac{1}{S-1} \sum_{s=1}^{S} (\bar{E}_s - \bar{E})^2}, \qquad \bar{E} = \frac{1}{S} \sum_{s=1}^{S} \bar{E}_s.$$

A 3×3 grid therefore represents three independently trained sources, each evaluated with three fitted readers. When source and reader seeds vary jointly, SD describes the complete fitting pipeline. In incomplete grids, the fitted reader counts are reported rather than treating the grid as balanced.

Paired effects. Complete-method improvement is $100(\bar{E}_{\mathrm{control,matched}} - \bar{E}_{\mathrm{SPRII,matched}})/\bar{E}_{\mathrm{control,matched}}$. Within-source fresh-reader history gain instead compares

Table 7: **Evaluation populations, fitted-model coverage and consuming routes.**

Analysis	Population	Source, reader and measurement
SpringWorld		
Sealed comparison	256 continuous primary; 36 factorial secondary	3 sources × 3 readers; matched inputs. The 292-system pool is auxiliary.
Original reader	100-system cold start; 178-system mixture	Separate grids: source 0/reader 0 reference; three-source component mean with reader 0; original 3 × 3 comparison.
Common adapter	100-system bottleneck; 178-system horizon/probes	FCRL uses each source's own query encoder; the four bottleneck arms share one source's query encoder.
Landscapes	64 development systems	Three sources, reader 0; fixed donor–target interventions.
Controlled D-Clean and PokeWorld		
D-Clean TDS	100-system report	3 × 3 grid; selected state-space source and common raw-state reader.
D-Clean same-donor	200 validation systems	Common Align + Cross query encoder; 3 × 3 grid, with 27 Null/Persistent/Oracle and 9 Decode cells.
Poke geometry/probes	400 confirmation; 400 validation systems	Centroid probes and actual donor-window probes are separate measurements.
Poke physical-risk use	96 development systems	Wrong-factor physical MSE; native predictor and state decoder fixed.
Poke native direction	144 validation systems	G1/G2, three sources each; dimensionless projection in each source's native embedding.
Poke fresh-reader value	400 development systems	Separately fitted task readers; Shuffled is a trained arm, not a fixed-input substitution.
Transferred learners, robot data and partners		
Collision test	1,994 held-out episodes	JEPA, CPC and RSSM each have 3 joint source/reader seeds per arm; variation across trained fits is reported as source SD.
Balls extension	2,000 development recipients	The reporting cohort includes source/readout selection examples.
RH20T	25-task forecast; 24-task hard negatives; separate 25-task input validation	Specificity and absolute risk share the hard-negative bank. Input controls use the separate validation bank.
Overcooked	20 familiar and 2 held-out development partners	One layout; 3 training seeds for return/probes at each history budget. Return mean and sample SD are over seeds; donor interventions retain one checkpoint.

separately trained Null and Matched readers. Fixed-weight wrong-minus-matched error measures reliance on the supplied donor. Relative effects use ratios of mean errors, not averages of case-wise percentages. In the same-donor bottleneck discussion, relative differences use Decode as the denominator: Persistent is 22.8% lower in D-Clean and 2.12% higher in SpringWorld. Raw Decode-minus-Persistent differences retain their original units and intervals. Intervals pair conditions within the appropriate unit before averaging fitted seeds and resampling that unit. Analyses specify their system/task/recipient unit, stratification, multiplicity and bootstrap seed.

Table 8: Fixed randomization settings used in the appendix.

Experiment	Operation	Seed
Baxter	Grasp-identifier split permutation	20260822
RH20T additional 25-task bank	Paired task bootstrap	20260923

Checkpoint and configuration selection. CoPhy's cross-scene development and held-out Collision comparisons use the distinct recipes specified in Appendix B.4. Each analysis specifies whether it uses a final-step checkpoint, a validation-selected reader or a released-default reference.

Coverage and selection. The source-0 CoPhy matrix has second-source JEPA fits in all three scenes and CPC Collision, plus three-source RSSM Collision. The held-out Collision comparison includes four arms with three joint source/reader seeds each for JEPA, CPC and RSSM. Balls separately has three-source RSSM (four arms) and CoPhyNet (three arms, reader seed 0) extensions. These are distinct fitted grids; Table 7 and the experiment-specific captions retain their evaluation populations. Overcooked has three training seeds for returns and probes at each history budget; its donor intervention remains a single-checkpoint study (Appendix B.4). Swimmer's three frozen bases span separate architectures.

CoPhy development reporting includes reader-selection examples; CoPhyNet Balls also uses its source-selection cohort. The held-out Collision episode comparison uses a separate population. Recipient-bootstrap intervals describe variation across recipients conditional on fitted source/readers; source SD describes fitted-model variation. The package defines donor graphs.

B.7 Reproduction assets

The Supplementary Evidence Package accompanies the manuscript as a single indexed archive. Its `README.md` and experiment index map the experimental questions to full numerical tables, figure data and source identities; `full_tables.pdf` provides a readable record. The package includes checkpoint/split hashes, selection records, mappings between implementation and manuscript method names, and per-recipient Balls factor-assay errors. Public datasets and released implementations retain their original access and license terms.

C Predictive Performance and Closest Comparisons

This section expands Tables 1–2: the primary task comparisons come first, followed by closest adapted constructions and the released NOD operator. Each comparison retains its own inputs, readout and evaluation population; these task effects motivate the representation analyses in Appendix E.

Primary-comparison average. The abstract's 14.6% is the equal-weight mean of five relative error reductions in Table 1: SpringWorld versus Native JEPA (22.3%), Collision CPC/RSSM/JEPA versus Structure (15.4%, 11.4%, 14.2%), and Balls RSSM versus Structure (9.9%). Calculations use the archived precision before rounding; each setting–learner comparison receives one weight. This summarizes these five primary comparisons, not all thirteen settings.

C.1 Cold-start prediction on sealed systems: SpringWorld

Align + Cross reduces cold-start error relative to Native and TDS on the 256 continuous systems in the primary sealed SpringWorld test. All methods receive an independent 96-frame same-system

donor and six cold-start queries per system at $h = 16$. The final 10,000-update sources and readers are frozen: three source seeds and three readers per source for each recipe. Queries average within system, then readers within source, then sources equally. Paired 95% intervals use 10,000 stratified physical-system draws, conditional on these fitted models.

All three source means and all nine paired cells favor SPRII against each comparator. The 292-system pool weights systems rather than strata equally; it does not replace the continuous primary endpoint.

Table 9: **Sealed SpringWorld prediction and paired contrasts.** Results average three sources and three readers per source. Brackets show 95% physical-system intervals conditional on these fitted models. Positive gaps favor Align + Cross; the system-weighted pooled result is auxiliary.

A. Absolute prediction error

Source	Primary: 256 systems	Secondary: 36	Pooled: 292
Native	0.13820 [0.11850, 0.15920]	0.21370	0.14750
TDS	0.13030 [0.11180, 0.15020]	0.19670	0.13840
Align + Cross	**0.10740** [0.09036, 0.12670]	**0.12730**	**0.10990**

B. Paired contrasts within each population

Population	Comparator	MSE gap [95% interval]	Reduction [95% interval]
Primary	Native	0.03084 [0.01735, 0.04429]	22.31% [13.15, 30.69]
Primary	TDS	0.02284 [0.01039, 0.03487]	17.54% [8.40, 25.87]
Secondary	Native	0.08645 [0.02195, 0.16640]	40.45% [16.34, 54.55]
Secondary	TDS	0.06944 [0.01587, 0.13300]	35.30% [12.10, 49.13]
Pooled	Native	0.03769 [0.02285, 0.05302]	25.55% [16.90, 33.32]
Pooled	TDS	0.02859 [0.01560, 0.04176]	20.65% [12.00, 28.45]

C.2 Held-out prediction: Collision and Balls

SPRII lowers error across CPC, RSSM and JEPA, improving on Native, Structure and Random in every fitted seed. Reductions relative to Structure are 15.4%, 11.4% and 14.2%, respectively (Table 10). Three independent donor histories and the first three query frames predict twelve future frames under the S3/query3 task. A frozen donor plan admits 1,994 of 2,000 official normal-test episodes; six lack three distinct legal donors in one object slot.

All four arms cover three joint source/reader seeds. CPC and RSSM use source-100 checkpoints; every JEPA arm uses source-50 checkpoints. Readers use a 100-epoch budget. This Collision normal-test comparison uses the above donor eligibility and checkpoint selection in Appendix B.

Table 10: **Held-out Collision prediction.** Physical trajectory MSE on 1,994 eligible episodes, mean $\pm$ sample SD across three joint source/reader seeds. Training budgets match within each learner family.

	Physical trajectory MSE				Reduction	
Family	Native	Structure	Random	SPRII	vs. Structure	vs. Random
CPC	0.248 ± 0.008	0.223 ± 0.006	0.221 ± 0.006	$\mathbf{0.189} \pm 0.007$	15.4%	14.7%
RSSM	0.268 ± 0.007	0.228 ± 0.005	0.222 ± 0.007	$\mathbf{0.202} \pm 0.007$	11.4%	9.0%
JEPA	0.250 ± 0.004	0.215 ± 0.007	0.265 ± 0.003	$\mathbf{0.184} \pm 0.005$	14.2%	30.4%

The Supplementary Evidence Package contains fitted-seed results, available model bindings and source identities. Table 65 reports additional CoPhy configuration comparisons.

Held-out Balls4 prediction with RSSM. On the same 1,000 four-ball recipients and eligibility mask, all four RSSM arms use 100 source epochs and 100 reader epochs, with three independent source/reader seeds. The existing checkpoints were evaluated after a 12-cell development-consistency check; this is an extension of the existing test evaluation, not a newly sealed population. Cross lowers mean error by 10.7% against Native, 9.9% against Structure and 13.1% against Random, with the

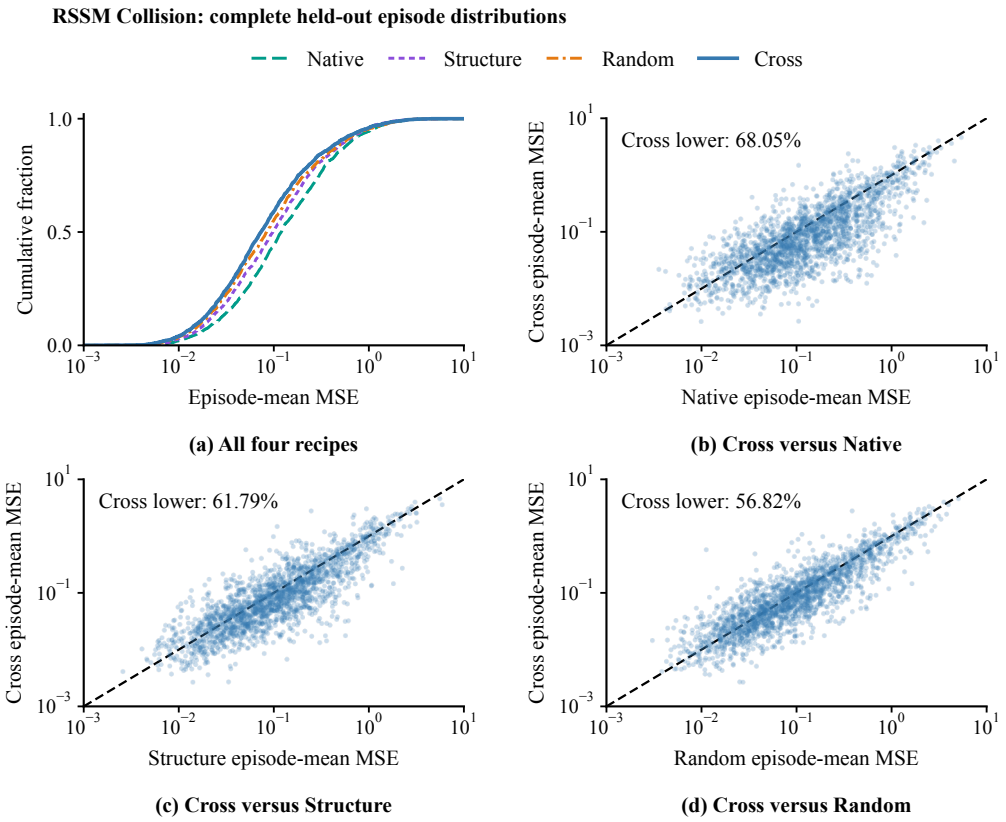


Figure 10: **Cross lowers RSSM error on most held-out episodes.** (a) Error distributions on 1,994 episodes. (b)–(d) Paired comparisons with each control. Episode MSEs are averaged over three source/reader runs. Points below the diagonal favor Cross; fractions report their share. Axes are logarithmic.

same direction in all three seeds. This result concerns four-ball trajectory prediction, not extrapolation to different ball counts.

Table 11: **RSSM on 1,000 held-out Balls4 recipients.** MSE: mean ± sample SD over three seeds.

Method	MSE	Source / reader epochs
Native	1.292 ± 0.037	100 / 100
Structure	1.280 ± 0.031	100 / 100
Random	1.327 ± 0.011	100 / 100
SPRII (Cross)	1.154 ± 0.025	100 / 100

C.3 Matched comparisons with adapted constructions

SPRII gives lower matched-reader error than the closest constructions under the task interfaces specified in Appendix B. Tables below group results by reader and evaluation population.

SpringWorld trajectory-decoupled prediction. SPRII lowers matched cold-start error relative to both full and compact visual TDS under the original SpringWorld reader. The NOD-inspired TDS adaptation predicts recipient visual embeddings from a same-system donor, current input and future actions, using $\mathcal{L}_{\text{cross}} + 0.02\mathcal{L}_{\text{SIGReg}}$ and no self or alignment term. It receives the same 10,000 source updates and independent 96-frame histories as SPRII. Compact TDS selects a zero-dimensional history slot ($d = 0$) from $0, \ldots, 6$ on a separate selection-development bank. A common original physical reader evaluates three sources by three readers on 100 development systems.

Pooled Matched reductions are 29.73% and 34.40%, respectively, with the same direction in all three source means. This visual TDS construction uses the trajectory-decoupled principle of Chen et al. (2026).

Fixed-recipe temporal FCRL-style comparison. The fixed temporal FCRL-style adaptation realizes little history gain, whereas SPRII lowers matched error under the common-adapter comparison. This temporal adaptation of same-function contrastive learning (Gondal et al., 2021) uses two independent histories per system, a shared visual/history encoder and a discarded contrastive projection. Temperature 0.07 is fixed without a sweep. Each of three sources uses 10,000 updates; three fresh readers follow the common-adapter recipe. Each method retains its own frozen query encoder, so the absolute comparison is between complete source recipes.

SPRII lowers Matched error by 58.44% [52.90,63.74]% under this adaptation and budget. Training-only physical probes remain near zero for the contrastive codes on the separate 178-system mixture.

D-Clean state-space TDS. SPRII has lower h32 common-reader error than state-space TDS even though both frozen sources retain strong drag accessibility. State-space TDS predicts true state increments from independent donor history, current state and actions at horizons $1, 4, 16$. A separate source selects $d = 4$ from $0, \ldots, 4$ using native h16 error. Three sources by three readers then evaluate 100 report systems. At h32, common-reader raw-state Matched MSE is 0.00015072 for SPRII and 0.00134349 for TDS (88.78% lower); the paired gap is 0.00119278 [0.00046947,0.00223990], with five-comparison adjusted interval [0.00038066,0.00267765]. All source means agree in direction. TDS retains drag $R^2 = 0.9712$–0.9767 (SPRII: 0.9965–0.9967), so larger task error does not mean absent physical information. Native direct h16 errors (0.00026174–0.00028337) and iterative h32 errors (0.00182717–0.00198696) use different input constructions and are reported separately in the Supplementary Evidence Package. This 100-system comparison does not share the population of the 200-system same-donor experiment in Appendix G.2. The separate released-component comparison below uses the same report population.

D-Clean: common-reader dynamics baselines. SPRII lowers long-horizon error relative to NOD, FCRL, CaDM and the DALI context adaptation (Table 12). CaDM learns a context encoder through forward and backward dynamics prediction (Lee et al., 2020); SPRII reduces its h16 and h32 errors by 32.16% and 33.70%, respectively. Both reductions hold across all three sources. The h32 reduction against FCRL is 88.70%, also with the same direction in every source. The implementation and common-reader protocol are specified in Appendix B.

The same source checkpoints support factor-accessibility and fixed-reader use measurements. Training-fitted damping probes attain $R^2_{\log \gamma} = 0.9739$ for CaDM and 0.9962 for SPRII. Their h32 wrong-minus-matched MSEs are 0.05329 and 0.05305, respectively: both fitted readers use the supplied history. The Supplementary Evidence Package lists all horizons and source values.

The DALI comparison adapts its context-learning component to the same frozen-code common-reader interface; it is not a native control-return ranking. The selected 160,000-update source uses a separate 100-system selection bank; all three source seeds and their three 20,000-update readers are included on the 100-system report bank. Other sources use 20,000 updates. SPRII lowers h32 mean error by 96.7% relative to this longer-budget DALI source, in the same direction for all three sources. The 20k/80k/160k DALI rows in Table 12 retain the budget dependence; unequal source budgets are not described as equal total computation. At 160k, DALI's damping probe has $R^2 = 0.785$ and its h32 wrong-minus-matched MSE is 0.0398, so its larger prediction error does not mean absent context information or use.

Source dispersion and native controls. For SPRII/NOD/FCRL/CaDM, h32 source SD is 0.135/104.076/1.014/0.063 in 10^{-4} raw-state MSE units. Damping-probe R^2 is 0.996/0.643/0.852/0.974 and fixed-reader wrong-minus-matched MSE is 0.05305/0.03556/0.04891/0.05329.

Table 12: **D-Clean prediction with a common reader.** Raw-state MSE in units of 10^{-4}, mean $\pm$ source SD after averaging three readers per source (3 sources). Source budgets vary as shown. The 100-system report population was previously evaluated; it is not a new sealed test.

Source	Updates	h1	h4	h16	h32
SPRII	20,000	0.0314 ± 0.002915	0.1166 ± 0.005082	0.5232 ± 0.01173	1.507 ± 0.1355
CaDM	20,000	0.03324 ± 0.001838	0.1474 ± 0.008051	0.7711 ± 0.008115	2.273 ± 0.06284
FCRL	20,000	0.1228 ± 0.01122	0.762 ± 0.08086	4.833 ± 0.492	13.34 ± 1.014
NOD	20,000	0.3894 ± 0.6261	3.814 ± 6.412	23.79 ± 40.18	61.88 ± 104.1
DALI	20,000	0.5579 ± 0.02027	5.307 ± 0.2763	35.49 ± 1.735	89.3 ± 3.886
DALI	80,000	0.3314 ± 0.035	2.962 ± 0.2441	20.54 ± 1.948	53.9 ± 5.067
DALI	160,000	0.3201 ± 0.03572	2.732 ± 0.3115	17.26 ± 1.991	45.92 ± 5.362

Table 13: **Prediction with adapted baselines and context controls.** A/B report Spring standardized MSE; C reports D-Clean raw-state MSE, each averaged equally over three sources and three readers. Fixed Wrong uses the Matched reader. D uses native TDS decoders rather than C's common reader; its direct and rollout errors are separate diagnostics.

A. Spring: original reader

Reader	Source	Fresh Null	Matched	Fixed Wrong
Original	Align + Cross	0.1775	0.1121	0.2439
Original	TDS	0.1716	0.1596	0.1787
Original	Compact TDS ($d = 0$)	0.1709	0.1709	0.1709

B. Spring: common adapter

Reader	Source	Fresh Null	Matched	Fixed Wrong
Common adapter	Align + Cross	0.1778	0.1136	—
Common adapter	FCRL-style	0.2767	0.2735	—

C. D-Clean: common-reader h32 context controls

Source	Matched	Fresh Null	Fixed Wrong	Fixed Zero
SPRII	0.0001507	0.01821	0.05320	0.03105
State-space TDS	0.0013430	0.01821	0.05257	0.03143

Other source comparisons and budgets are in Table 12.

D. D-Clean: TDS native-decoder diagnostics

Measurement	Source 0	Source 1	Source 2
Direct h16	0.0002706	0.0002834	0.0002617
Free rollout h32	0.001948	0.001827	0.001987

C.4 Prediction and continuation on a released operator: Burgers

The fixed SPRII endpoint lowers native prediction error relative to NOD across ID, viscous OOD and inviscid OOD. We use the released NOD implementation and training configuration (Chen et al., 2026), with source seeds 1234, 5678 and 9012. The field environment and available viscosity grid are specified in Appendix B.4 and Figure 9(a).

Training and selection. SPRII retains the native prediction objective during relation training, then continues prediction with $\lambda_{\text{align}} = 0$. Zero-alignment training was checked against the unchanged NOD path using matched seeds, batches and initialization. Both source searches span 200 epochs and select checkpoints by ID development loss. The SPRII pre-continuation checkpoints receive another 25, 113 and 127 epochs, respectively, at 45 updates per epoch; the final 25 epochs use learning rate 5×10^{-5}. All reported SPRII values use these fixed final endpoints. Continued NOD starts from its own selected checkpoints and receives the same additional updates. This control matches continuation updates; total ancestral training and search budgets are recorded separately in the Supplementary Evidence Package.

Native prediction. SPRII reduces error by 1.075%, 6.554% and 2.921% across the three distributions (Table 14). Native MSE follows the released evaluator's unweighted mean of batch MSE over the full trajectory, with 45 ID, 500 viscous-OOD and 100 inviscid-OOD examples. These populations were used in earlier comparisons; the recipe was selected on development data. Source SD and paired-source intervals accompany the table.

Table 14: **Burgers prediction and continuation controls.** Native trajectory MSE in units of 10^{-3}, mean $\pm$ sample SD across three sources. SPRII includes prediction-only continuation; controls match its added updates. Released-default NOD has no added updates. Dashes denote unreported source SD.

Track	ID	Viscous OOD	Inviscid OOD
Primary comparison			
Released-default NOD	9.096 ± 0.544	8.012 ± 0.815	11.86 ± 0.628
SPRII fixed endpoint	$\mathbf{8.999} \pm 0.798$	$\mathbf{7.487} \pm 0.183$	$\mathbf{11.51} \pm 0.360$
Matched additional-update controls			
Continued NOD	9.379 ± 0.340	7.408 ± 0.130	11.49 ± 0.215
Random continuation	9.538 (–)	8.900 (–)	11.50 (–)

Paired-source 95% t intervals for SPRII minus released-default NOD ($n = 3$, df=2; MSE $\times 10^{-3}$): ID $[-2.472, 2.277]$, viscous OOD $[-2.982, 1.932]$, and inviscid OOD $[-2.783, 2.091]$.

Relative to continued NOD, SPRII lowers ID error by 4.06%, with 1.06%/0.20% higher viscous/inviscid OOD errors. Relative to Random continuation, it lowers ID and viscous-OOD errors, while inviscid error is nearly tied. The complete three-source controls appear in the same table.

Organization and fixed-predictor use. The primary-checkpoint measurements in Table 15A use training-standardized latent geometry, training-fitted viscosity probes and a common history bank. SPRII has a higher between/within ratio and viscosity probe R^2 than released-default NOD. At direct horizon $H = 50$, the mean wrong-donor MSE changes are close to zero for both models. The additional-update analysis in panel B uses a distinct 11-system, relative-L_2 donor assay: both models incur positive wrong-donor penalties, and continued NOD has stronger geometry and viscosity accessibility. These measurements separate accessible viscosity from the response of a fixed predictor to donor substitution.

Table 15: **Burgers organization, viscosity accessibility and donor use.** Entries are mean $\pm$ SD across three sources. B/W is the latent between/within squared-distance ratio. A uses 45 ID recipients and a direct MSE donor assay; B uses training geometry and a separate 11-system relative-L_2 assay. Positive penalties indicate harm from wrong donors; 95% intervals resample paired sources and systems.

A. Primary comparison: released NOD and SPRII

	ID B/W	ID R^2	$H = 50$ MSE ($\times 10^{-3}$)	
Method			**Matched**	**Wrong−matched**
NOD	22.3 ± 19.09	0.651 ± 0.458	12.89 ± 0.992	0.014 ± 0.391
SPRII	$\mathbf{31.64} \pm 4.506$	$\mathbf{0.762} \pm 0.141$	$\mathbf{12.19} \pm 1.944$	$\mathbf{-0.054} \pm 0.597$

B. Additional-update comparison

Method	Train B/W	ID R^2	Viscous-OOD R^2	$H = 50$ relative-L_2 penalty [95% interval]
Continued NOD	42.07 ± 14.74	0.924 ± 0.006	0.975 ± 0.004	0.1331 [0.06729, 0.2127]
SPRII	24.58 ± 8.62	0.762 ± 0.141	0.895 ± 0.116	0.1275 [0.06538, 0.2041]

See Supplementary Evidence Package for other schedules and source-wise horizon/donor arrays.

Test-time-adaptive operator references. GEPS and CoDA use 50 support-code optimization steps for each query, while NOD and the SPRII endpoint use none. Table 16 retains those completed references under the released native metric. CoDA has lower mean error in all three distributions; GEPS

has lower ID error and higher OOD means, with substantial source dispersion. These comparisons do not establish equal inference cost or uniform dominance.

Table 16: **Burgers prediction with test-time adaptation.** Native full-trajectory MSE in units of 10^{-3}, mean $\pm$ sample SD across three sources. Both methods use 50 support-code updates per query; zero-step results are in Table 14.

Method	ID	Viscous OOD	Inviscid OOD	Code steps
GEPS	5.60 ± 7.05	12.37 ± 14.21	13.84 ± 17.41	50
CoDA	3.38 ± 0.57	6.48 ± 1.42	5.89 ± 1.55	50

C.5 Relation supervision on a frozen CoDA host

An amortized CoDA instance adds a history encoder to each of three fixed CoDA source decoders. The four arms in Table 17 share a teacher-fit initialization, a fixed encoder budget and final-checkpoint selection. Each source seed retains its own decoder and train-only teacher-code cache. Seeds 1234, 5678 and 9012 retain their completed 5,000-update decoders. All arms use the same 101-frame support histories and fixed evaluation manifest: 45 ID, 500 viscous-OOD and 100 inviscid-OOD pairs. The relation-trained arm uses Cross.

Table 17: **Objectives on the frozen CoDA decoder.**

Arm	Objective addition	Role
Plain amortized CoDA	Self prediction and common regularization	History-encoder baseline.
Matched Cross	A second prediction using the related donor	Relation-supervised arm.
Random Cross	The same second prediction using a wrong-system donor	Pairing control with matched donor marginals.
Duplicate Self	A second self-prediction forward/backward pass	Matched computation and reconstruction-loss scale.

A shared 500-update teacher fit precedes 1,000 encoder updates per arm. Within each seed, initialization and sampled histories are identical across arms. Plain uses one prediction branch; the other arms use two, so equal update counts do not imply equal training FLOPs. The primary endpoint uses the inferred context with no support-code optimization (code-0). The auxiliary code-50 endpoint initializes the code from the same encoder and applies fifty support-code optimization steps; the encoder and decoder remain frozen. The prespecified code-0 criterion compares Matched Cross with both Plain and Duplicate Self in each distribution: lower mean error and the same direction in a majority of source seeds. Random Cross measures sensitivity to the relation assignment. Paired contrasts use the same decoder seed, evaluation inputs and adaptation budget. Native MSE averages batch-8 errors equally and includes the initial frame. Viscosity ridge probes fit 360 training histories, with regularization selected by five-fold cross-validation over training cases.

Matched Cross meets this code-0 criterion in all three distributions: relative to Plain, mean error falls by 6.70%, 5.17% and 6.50%; relative to Duplicate Self, it falls by 4.98%, 6.74% and 7.39%, respectively. Each comparison favors Matched Cross in all three seeds. Its smaller advantages over Random Cross (1.74%, 0.38% and 1.46%) have directions 2/3, 1/3 and 2/3 across seeds. With only three source runs, all nine paired t intervals include zero (Table 19); these results do not establish a stable advantage of correct over random pairing.

After fifty support-code updates, Matched Cross has lower mean error than Duplicate Self in all three distributions, but lower mean error than Plain only on viscous OOD; Random Cross has slightly lower means than Matched Cross in all three distributions. All nine paired intervals include zero (Table 20). Each arm's mean query error is higher at code-50 than at code-0 in every distribution. Matched Cross retains the highest mean viscosity-probe R^2 (Table 21), separating parameter accessibility from forecasting value under this fixed adaptation budget. The predictive gains reported here concern amortized inference.

Table 18: **Amortized prediction with a frozen CoDA decoder.** Native full-trajectory MSE in units of 10^{-3}, mean $\pm$ sample SD across three sources. Code-0 is the primary endpoint.

Distribution	Arm	Code-0	Code-50
ID	Plain	3.4834 ± 0.5669	$\mathbf{3.5123} \pm 0.5353$
ID	Matched Cross	$\mathbf{3.2500} \pm 0.7286$	3.5566 ± 0.4909
ID	Random Cross	3.3076 ± 0.7055	3.5164 ± 0.4933
ID	Duplicate Self	3.4203 ± 0.7573	3.5899 ± 0.4809
Viscous OOD	Plain	6.3900 ± 1.4357	7.1890 ± 1.6605
Viscous OOD	Matched Cross	$\mathbf{6.0598} \pm 1.1847$	7.0845 ± 1.4797
Viscous OOD	Random Cross	6.0832 ± 1.2241	$\mathbf{7.0310} \pm 1.5672$
Viscous OOD	Duplicate Self	6.4976 ± 1.5413	7.4010 ± 1.6684
Inviscid OOD	Plain	5.8833 ± 1.6998	$\mathbf{6.7663} \pm 2.2725$
Inviscid OOD	Matched Cross	$\mathbf{5.5006} \pm 1.4956$	6.8105 ± 2.1187
Inviscid OOD	Random Cross	5.5822 ± 1.5558	6.8046 ± 2.2922
Inviscid OOD	Duplicate Self	5.9394 ± 1.8774	7.0893 ± 2.5458

Table 19: **Paired CoDA contrasts at code-0.** Differences are comparator minus Matched Cross in units of 10^{-3}; positive favors Matched Cross. Intervals are paired-source 95% t intervals ($n = 3$); n_+ counts positive source differences.

Distribution	Comparator	Seed 1	Seed 2	Seed 3	Mean [95% CI]	n_+
ID	Plain	0.1270	0.4228	0.1502	$0.2333\,[-0.1752, 0.6418]$	3
ID	Random Cross	-0.0113	0.0710	0.1129	$0.0576\,[-0.0994, 0.2145]$	2
ID	Duplicate Self	0.3036	0.1917	0.0155	$0.1703\,[-0.1904, 0.5310]$	3
Viscous OOD	Plain	0.1989	0.7125	0.0793	$0.3302\,[-0.5055, 1.1659]$	3
Viscous OOD	Random Cross	-0.0904	0.1676	-0.0071	$0.0234\,[-0.3038, 0.3506]$	1
Viscous OOD	Duplicate Self	0.4304	0.8332	0.0499	$0.4378\,[-0.5351, 1.4108]$	3
Inviscid OOD	Plain	0.2188	0.7455	0.1836	$0.3826\,[-0.3993, 1.1646]$	3
Inviscid OOD	Random Cross	-0.0237	0.2427	0.0257	$0.0816\,[-0.2704, 0.4335]$	2
Inviscid OOD	Duplicate Self	0.2938	0.9736	0.0488	$0.4387\,[-0.7515, 1.6290]$	3

Table 20: **Paired CoDA contrasts at code-50.** Differences are comparator minus Matched Cross in units of 10^{-3}; positive favors Matched Cross. Intervals are paired-source 95% t intervals ($n = 3$); n_+ counts positive source differences.

Distribution	Comparator	Seed 1	Seed 2	Seed 3	Mean [95% CI]	n_+
ID	Plain	0.0046	-0.0930	-0.0446	$-0.0443\,[-0.1655, 0.0768]$	1
ID	Random Cross	-0.0361	-0.0165	-0.0681	$-0.0402\,[-0.1049, 0.0244]$	0
ID	Duplicate Self	0.0232	0.0760	0.0006	$0.0333\,[-0.0629, 0.1294]$	3
Viscous OOD	Plain	0.1935	0.2277	-0.1076	$0.1045\,[-0.3537, 0.5628]$	2
Viscous OOD	Random Cross	-0.0976	0.1018	-0.1646	$-0.0534\,[-0.3977, 0.2908]$	1
Viscous OOD	Duplicate Self	0.3286	0.5325	0.0882	$0.3165\,[-0.2360, 0.8689]$	3
Inviscid OOD	Plain	0.0396	0.0626	-0.2347	$-0.0442\,[-0.4551, 0.3668]$	2
Inviscid OOD	Random Cross	-0.0369	0.1850	-0.1656	$-0.0059\,[-0.4464, 0.4347]$	1
Inviscid OOD	Duplicate Self	0.0376	0.8296	-0.0308	$0.2788\,[-0.9092, 1.4669]$	2

Table 21: **CoDA viscosity accessibility and decoder invariance.** Probe R^2 is mean $\pm$ sample SD across three sources on 45 ID histories. Decoder change is the maximum absolute change during encoder fitting, over all three decoders; all remain unchanged.

Arm	Code-0 viscosity R^2	Code-50 viscosity R^2	Max. decoder change
Plain	0.1325 ± 0.1388	0.2765 ± 0.2184	0
Matched Cross	$\mathbf{0.1752} \pm 0.1545$	$\mathbf{0.3010} \pm 0.1946$	0
Random Cross	0.1291 ± 0.1324	0.2845 ± 0.1945	0
Duplicate Self	0.1527 ± 0.1604	0.2878 ± 0.2334	0

Table 22: **CoDA fitting and evaluation costs.** Times are seconds, averaged over three sources: per fit for training and per query for evaluation. Code-0 includes encoder inference and forecasting; Code-50 rows exclude encoder inference. Evaluation times are amortized execution costs, excluding data transfer, decoder construction and scoring. Shared decoder and teacher costs are reported separately.

Quantity	Plain	Matched Cross	Random Cross	Duplicate Self
Trainable encoder parameters	89,282	89,282	89,282	89,282
Encoder fitting time	1267.92	2489.37	2517.80	2519.17
Code-0 encoder + forecast	0.03416	0.03500	0.03510	0.03353
Code-50 adaptation time	6.84052	6.74525	6.57095	6.47388
Code-50 forecasting time	0.03490	0.03604	0.03407	0.03370
Shared decoder fitting time		5372.08		
Shared teacher-code adaptation		2286.44		
Shared teacher-fit time		1.44		

A separate single-source development comparison used 45 support/query pairs from nine systems. At zero adaptation steps, Cross reduced mean error by 3.84% relative to Plain, with all nine systems in the same direction; it fell below that study's prespecified 5% improvement threshold. The four-arm analysis retains random-pairing and duplicate-self controls. Original CoDA with zero initialization and fifty support optimization steps is the external reference in Table 16. The report populations follow the released evaluation protocol; development pairs serve only the preceding single-source comparison. The completed four-arm evaluation extends the historical report populations and is not a new sealed test; its results did not change the fixed recipe or the earlier development-gate conclusion. The source package retains all three seeds, per-query errors, probes and timing reductions.

D SpringWorld Case Study

This section collects the questions needed to interpret the SpringWorld instance: what history alone provides, how Align and Cross contribute, and what training resources and objective weights accompany those effects. The existing comparisons use the visual input and training protocol in Appendix B.4. Development component measurements and sealed prediction results use separate evaluation populations.

D.1 History-conditioned baselines

Native and Structure both receive history. Introducing the separate channel alone does not reproduce the joint method's benefit: the development component means are 0.170 for Native, 0.172 for Structure and 0.110 for Align + Cross (Table 35). Appendix C.1 gives the primary comparison against Native and trajectory-decoupled prediction on the separate sealed systems.

D.2 Contributions of Align and Cross

Align accounts for most of the observed organization and task improvement in the component study. The joint recipe lowers development error to 0.110, compared with 0.123 for Align and 0.168 for Cross; its source-wise between/within ratio is 2.360, compared with 2.045 and 0.252, respectively. Table 35 gives the full component comparison and separate task and geometry populations.

D.3 COMPUTATIONAL COST

Both sources use 10,000 optimizer updates and their frozen-context readers use 10,000 updates. Align and Cross are added within source updates; this SpringWorld instance has no continuation stage. Table 23 separates optimizer budgets, parameter counts and measured running time.

Table 23: **SpringWorld training cost per fit.** Update times use a common RTX 6000D benchmark. Full-fit times average three original runs under different hardware and runtime conditions and are not directly comparable to the update benchmark.

Quantity	Native	Align + Cross
Source optimizer updates	10,000	10,000
Additional continuation updates	0	0
Reader updates per fitted reader	10,000	10,000
Trainable source parameters	4,509,120	4,525,632
Compute-only time per source update (s)	0.1756	0.1948
Original full-fit wall time (s)	34,270.26	34,324.20

D.4 BUDGETS AND CONFIGURATION SELECTION

Table 24 separates the source budget, reader budget and selection rule. Three fitted readers on one source are not three independent source runs. The initial comparisons fix the relation weights at $\lambda_{\mathrm{p}} = 1$ and $\lambda_{\mathrm{x}} = 0.1$; the sensitivity study below is a separate development analysis.

Table 24: **SpringWorld budgets and configuration selection.** Updates are per fit.

Comparison	Source	Reader	Configuration and selection
Native / Structure / Align / Cross / joint	10,000	10,000	Fixed component weights; final source step. Component table uses reader 0.
Sealed Native / TDS / joint	10,000	10,000	Three sources $\times$ three readers; sources and readers frozen before the primary test.
Full / compact TDS	10,000	10,000	$d \in \{0, \ldots, 6\}$ for compact TDS; separate selection-development bank.
Temporal FCRL-style adaptation	10,000	10,000	Fixed temperature 0.07; three sources $\times$ three common-adapter readers.

The TDS and FCRL-style inputs and matched-reader restrictions are specified in Appendix C.3. Configuration counts describe distinct recipes in the reported comparison and its declared selection stage. They exclude seed replicas, the separate relation-weight sensitivity study and earlier prototypes. In particular, the current fixed-temperature FCRL adaptation is distinct from the earlier temperature-selected FCRL experiment. Native, Structure, Align, Cross, the joint recipe, full TDS and temporal FCRL each use one fixed source configuration. Compact TDS declares seven candidate dimensions, $d \in \{0, \ldots, 6\}$, for its selection stage.

D.5 SENSITIVITY TO RELATION WEIGHTS

The one-at-a-time development analysis varies $\lambda_{\mathrm{p}} \in \{0.25, 1, 4\}$ at $\lambda_{\mathrm{x}} = 0.1$, and $\lambda_{\mathrm{x}} \in \{0.025, 0.1, 0.4\}$ at $\lambda_{\mathrm{p}} = 1$. The shared default gives five distinct settings, with three source seeds per setting. Architecture, data, update count and reader protocol are fixed. Prediction and training-fitted factor probes use the development population; the sealed primary test is excluded from configuration selection. All fifteen sources and forty-five readers are included. The prediction measure is cold-start $q = 0$, $H = 16$ error on 100 development systems (600 cases); the probes are evaluated on 178 development systems.

Relative to the default $(1, 0.1)$, changing Align to 0.25 or 4, or Cross to 0.4, increases mean cold-start error by 9.38%, 46.23% and 15.83%, respectively, with the same direction in all three sources. Reducing Cross to 0.025 lowers the mean by 4.69%, with lower error in two of three sources. Probe scores and prediction error do not vary in lockstep: for example, $\lambda_{\mathrm{p}} = 4$ raises mean drag and stiffness R^2 while increasing prediction error.

Table 25: **Sensitivity to Align and Cross weights.** Mean ± SD over three sources; MSE first averages three readers per source. Probes predict $\log m$, $\log \gamma$ and $\log k$.

λ_p	λ_x	**Dev. MSE**	**Mass** R^2	**Drag** R^2	**Stiffness** R^2
0.25	0.1	0.1203 ± 0.0025	0.2837 ± 0.0494	0.1441 ± 0.1210	-0.0204 ± 0.0218
1	0.1	0.1100 ± 0.0081	0.3630 ± 0.0408	0.1598 ± 0.0220	0.1430 ± 0.0284
4	0.1	0.1609 ± 0.0360	0.3498 ± 0.0989	0.2694 ± 0.1131	0.3166 ± 0.2892
1	0.025	0.1049 ± 0.0112	0.4162 ± 0.0846	0.1645 ± 0.0447	0.3578 ± 0.2626
1	0.4	0.1274 ± 0.0027	0.3751 ± 0.0322	0.1331 ± 0.0277	0.2403 ± 0.1694

D.6 Gradient relationships

Every 100 optimizer steps, the diagnostic computes the cosine between $\nabla\mathcal{L}_{\text{align}}$ and $\nabla\mathcal{L}_{\text{cross}}$ on the shared persistent encoder g_p, using the same diagnostic batch and parameter subset. It records both gradient norms and zero-norm cases, restores RNG state and training mode, and makes no parameter update. Cosine summaries exclude zero-norm observations and retain negative values. Early, middle and late phases use update intervals $(0, 10000/3]$, $(10000/3, 20000/3]$ and $(20000/3, 10000]$, respectively. These local gradient measurements describe objective interaction; task performance is measured separately in Table 25.

Table 26: **Distribution of Align–Cross gradient cosines.** Quantiles and negative frequencies use valid observations pooled across three sources; zero-norm frequencies use all observations.

(λ_p, λ_x)	**Valid** n	**Mean**	**Q1**	**Median**	**Q3**	**Negative (%)**	**Zero norm (%)**
$(0.25, 0.1)$	300	0.0263	−0.0152	0.0177	0.0720	35.0	0.0
$(1, 0.1)$	300	0.0214	−0.0387	0.0181	0.0770	40.7	0.0
$(4, 0.1)$	300	0.0246	−0.0355	0.0173	0.0842	40.3	0.0
$(1, 0.025)$	300	0.0308	−0.0194	0.0214	0.0765	34.3	0.0
$(1, 0.4)$	300	0.0379	−0.0383	0.0317	0.1028	37.3	0.0

Table 27: **Gradient norms and cosines across sources and training phases.** Norms are unweighted medians; other columns show mean valid cosines. Repeated diagnostic observations are not independent source runs.

Weights	$\|\nabla\mathcal{L}_{\text{align}}\|$	$\|\nabla\mathcal{L}_{\text{cross}}\|$	**Source 0**	**Source 1**	**Source 2**	**Early**	**Middle**	**Late**
$(0.25, 0.1)$	5.9381	0.0086	0.0327	0.0328	0.0135	0.0378	0.0326	0.0092
$(1, 0.1)$	6.3315	0.0428	0.0208	0.0343	0.0092	0.0270	0.0286	0.0089
$(4, 0.1)$	5.4435	0.0800	0.0299	0.0182	0.0256	0.0273	0.0260	0.0205
$(1, 0.025)$	6.9317	0.0501	0.0393	0.0169	0.0363	0.0469	0.0307	0.0153
$(1, 0.4)$	6.6684	0.0334	0.0297	0.0455	0.0385	0.0152	0.0577	0.0408

Mean gradient cosines range from 0.0214 to 0.0379, while 34.3–40.7% of observations are negative. No diagnostic observation has a zero gradient norm. Thus the small positive mean does not imply uniformly aligned objectives, and these local measurements do not establish task-level synergy.

D.7 Relation reliability: PokeWorld comparison

Appendix E.1 and Figure 3 vary the correct-pair mixture weight α in PokeWorld. Proposition 2 in Appendix I.2 isolates the corresponding alignment pressure.

E FORMATION: RELATION DESIGN AND REPRESENTATION ORGANIZATION

PokeWorld, SpringWorld and Baxter test relation design, objective components and tactile selectivity. Fixed-route Use is measured separately in Appendix F.

E.1 RELATION RELIABILITY, SHARED FACTORS, AND OBJECTIVES: POKEWORLD

Takeaway. More reliable relations strengthen persistent organization; changing the pairing rule redirects frozen-code accessibility.

The experiments vary pair reliability at fixed R8, shared factors and objective composition. Within each study, sources 0/1/2 share architecture and budget; checkpoints and endpoints are fixed after validation and evaluated on 400 confirmation systems. Figure 11 shows the two formation maps.

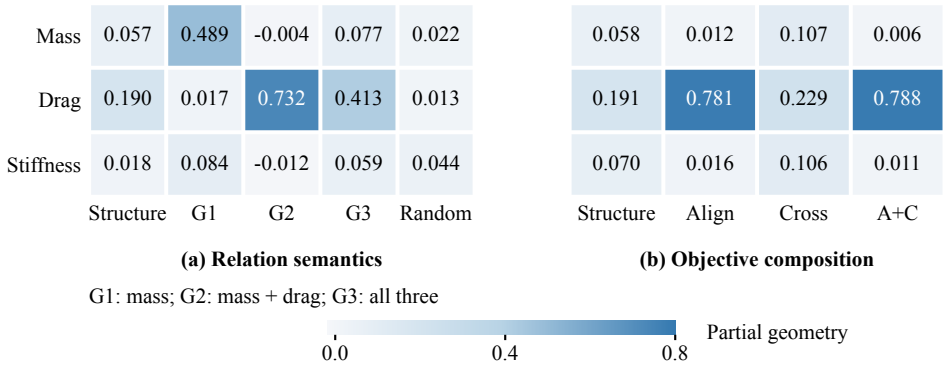


Figure 11: **Relation semantics and objective composition organize different factors.** Mean partial geometry across three sources on 400 confirmation systems. (a) Shared factors $G_1 = m$, $G_2 = (m, \gamma)$ and $G_3 = (m, \gamma, k)$; (b) objective components. Blocks use separate Structure references and a common color scale. Source SDs are in Tables 32 and 33.

Input evidence and interaction diversity. Rendered histories support factor readout, with higher accuracy for drag than for mass and stiffness. A separate privileged-dynamics estimator quantifies recoverability when numerical state is available. Replacing eligible same-rollout donors with independent rollouts raises drag readout from 0.2647 to 0.3469 while native h16 self loss changes from 0.3641 to 0.3869. Diversity and reliability answer separate questions: the R2/R4/R8 comparison changes available realizations, whereas α changes pair quality at fixed R8.

Table 28: **Factor readout from rendered histories and privileged dynamics.** Rendered-history entries show means and source SD.

Input readout (R^2)	Mass	Drag	Stiffness
Rendered history	0.270 ± 0.026	0.574 ± 0.009	0.149 ± 0.034
Privileged dynamics	0.762	0.937	0.760

Shared factors and frozen accessibility. Changing from G_1 to G_2 shifts accessibility from mass toward drag. Training-only ridge and MLP probes of frozen code centroids both favor G_1 for mass and G_2 for drag in every source replica on 400 validation systems. The spectral construction in Figure 26 shows how a wider sharing relation can change the selected predictive direction without lowering the original factor's score.

Table 29: **Both linear and nonlinear readouts show the G_1/G_2 accessibility shift.** Frozen centroid probes on validation systems. Error contrasts are G_2 minus G_1 with paired-system 95% intervals; positive favors G_1.

Probe	Target	G_1 R^2	G_2 R^2	MSE contrast	95% interval
Ridge	$\log m$	0.5271	0.0709	0.1492	$[0.1231, 0.1758]$
Ridge	γ	0.3075	0.9249	-0.7704	$[-0.8572, -0.6869]$
MLP	$\log m$	0.5272	0.0273	0.1635	$[0.1303, 0.1974]$
MLP	γ	0.1291	0.9644	-1.042	$[-1.143, -0.9438]$

Visualizing the frozen representation. Figure 12 displays all three replicas under both relations. Its factor coloring complements the probes; apparent spacing or orientation across independently projected sources is not a quantitative comparison.

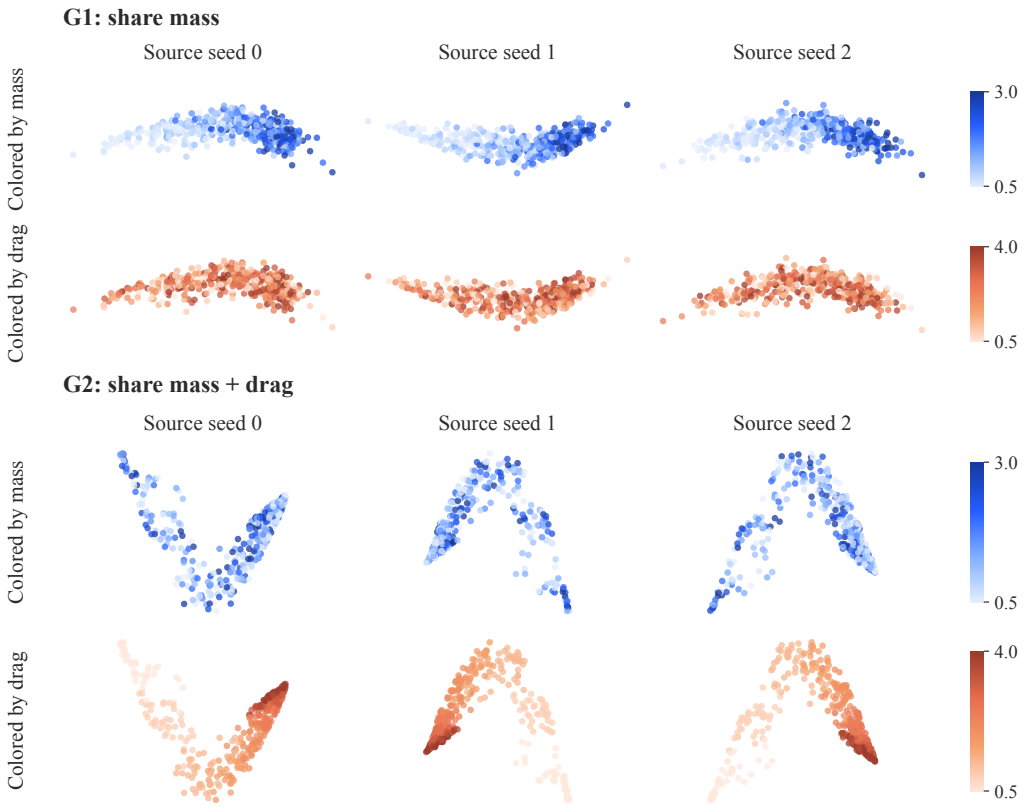


Figure 12: **Persistent-context atlas under G_1 and G_2.** Each panel shows the same 400 validation systems. Each frozen source has an independent PCA projection; mass and drag rows recolor identical coordinates using common physical-value ranges. Axes are not aligned across sources. The atlas illustrates the representations; quantitative comparisons use the geometry and probes in Appendix E.1.

Persistent-context atlas. The atlas uses the same six step-20,000 G1/G2 checkpoints and existing 400-system validation bank as the frozen factor probes. The registered training systems are the sorted first 400 indices from the training-bank permutation under seed 20260903, followed by the validation-bank permutation; this selects all 400 validation systems. Each code averages rollouts 0–3 and anchors $24, 32, 40, 47$. As in the geometry analysis, we first average anchors within each rollout, fit coordinatewise moments on the 400 training systems' four rollout means (standard-deviation floor 10^{-6}), normalize validation rollout means, and then form system centroids. No source update or confirmation/test access occurs. We center each source's validation centroids and compute a full-SVD PCA without whitening; the largest-magnitude loading fixes each component's sign, independently of physical labels. No axes are aligned across sources, and no systems, sources, components or

checkpoints are selected for visual appearance. The frozen coordinates, loading vectors, explained-variance ratios, normalization moments and system identifiers are included in the evidence package.

Objective controls and native references. The factorial keeps the two-view batch, predictor, dimensions and budget fixed. The relational InfoNCE alternative uses normalized persistent codes, one positive and 94 negatives per anchor, with all three sources in the $\tau \in \{0.07, 0.10, 0.20\} \times \lambda \in \{0.1, 1\}$ validation search. Confirmation geometry is 0.730 ± 0.019 and ratio 1.736 ± 0.104; the validation configuration record is in the Supplementary Evidence Package. Native embedding losses retain each learner's own target space. See Table 33 for the complete objective factorial.

The acquisition studies use separate D-Clean and PokeWorld banks: Poke's raw 24-step drag estimator scores 0.921, versus 0.320 ± 0.025 for monolithic JEPA and 0.481 ± 0.092 for Align + Cross, with 7.1% lower h16 object-state error on the original acquisition bank.

Actual-input versus privileged input estimates remain in Table 28; D-Clean acquisition uses its own one-shot sealed population. Native self prediction has no donor route.

Table 30: **D-Clean results on 200 one-shot sealed-test systems.** Means and sample SD over three sources. Errors use each source's own embedding target. Correct uses a same-system donor from another rollout; Gap is shuffled-minus-correct error. Native has no donor route.

Method	Drag R^2 ↑	Self h16 ↓	Correct ↓	Gap ↑
Native	0.9932 ± 0.0007	0.0789 ± 0.0356	n.a.	n.a.
Structure	0.9935 ± 0.0007	0.0736 ± 0.0404	0.5512 ± 0.0681	0.0797 ± 0.0079
Same-rollout alignment	0.4434 ± 0.0367	0.0520 ± 0.0078	0.0542 ± 0.0090	0.00635 ± 0.00104
Align	0.9984 ± 0.0001	0.0140 ± 0.0040	0.0141 ± 0.0041	0.0948 ± 0.0037
Align + Cross	0.9986 ± 0.0002	0.0155 ± 0.0073	0.0155 ± 0.0072	0.0941 ± 0.0031
Align–Random	0.5299 ± 0.0400	0.0563 ± 0.0100	0.0591 ± 0.0100	0.00034 ± 0.00009
Supervised γ	0.9985 ± 0.0002	0.0119 ± 0.0024	0.0124 ± 0.0024	0.0969 ± 0.0044

Table 31: **PokeWorld geometry under history diversity and relation fidelity.** Means and source SD over three seeds. A: validation geometry with train-standardized codes; partial drag correlation controls mass and stiffness. B: validation and frozen-confirmation results.

A. Independent-history budget

Relation	Budget	Within ↓	Between ↑	Ratio ↑	Partial ρ_γ ↑
Correct	R2	78.35 ± 4.67	29.38 ± 1.78	0.375 ± 0.017	0.095 ± 0.022
Correct	R4	55.66 ± 5.17	43.32 ± 2.82	0.781 ± 0.062	0.361 ± 0.036
Correct	R8	28.09 ± 1.93	104.30 ± 2.01	3.729 ± 0.313	0.773 ± 0.009
Random	R2	70.80 ± 8.74	17.28 ± 2.03	0.244 ± 0.002	-0.001 ± 0.017
Random	R4	83.68 ± 16.91	20.39 ± 3.64	0.245 ± 0.008	-0.010 ± 0.016
Random	R8	126.90 ± 0.89	34.18 ± 0.39	0.269 ± 0.004	-0.006 ± 0.004

B. Relation fidelity

	Validation		Confirmation	
α	Ratio ↑	Drag ↑	Ratio ↑	Drag ↑
0	0.272 ± 0.006	0.061 ± 0.031	0.260 ± 0.006	0.031 ± 0.026
0.5	0.976 ± 0.051	0.243 ± 0.056	0.880 ± 0.047	0.264 ± 0.063
1	3.749 ± 0.058	0.781 ± 0.027	4.391 ± 0.156	0.796 ± 0.023

Donor support and candidate composition. G_1 and G_2 use the same frozen donor-bank file. Each recipient has three candidate donor systems and four rollouts per candidate, giving twelve supported donor histories under either relation. At each update, the sampler selects one of the three systems uniformly, then selects one of its four rollouts uniformly. This fixed training support is distinct from the full set of semantically eligible systems (Table 34).

G_1 candidate construction uses fixed drag-index offsets $(1, 3, 7)$ and a different-stiffness system at each offset. G_2 selects three candidates from the same-mass, same-drag set with different stiffness.

Table 32: **Factor-selective geometry under relation refinement.** Means and sample SD over three sources. Exact factor tuples are split-disjoint; individual factor levels are shared.

	Validation			Confirmation		
Condition	**Mass**	**Drag**	**Stiffness**	**Mass**	**Drag**	**Stiffness**
Structure	0.069 ± 0.008	0.194 ± 0.007	0.037 ± 0.012	0.057 ± 0.010	0.190 ± 0.002	0.018 ± 0.008
G_1	0.419 ± 0.006	0.013 ± 0.009	0.103 ± 0.004	0.489 ± 0.009	0.017 ± 0.006	0.084 ± 0.003
G_2	0.015 ± 0.000	0.720 ± 0.009	-0.013 ± 0.001	-0.004 ± 0.001	0.732 ± 0.012	-0.012 ± 0.001
G_3	0.064 ± 0.013	0.427 ± 0.087	0.067 ± 0.035	0.077 ± 0.035	0.413 ± 0.082	0.059 ± 0.023
Random	0.030 ± 0.007	0.036 ± 0.014	0.054 ± 0.008	0.022 ± 0.009	0.013 ± 0.007	0.044 ± 0.006

Table 33: **Objective components on 400 confirmation systems.** Means and sample SD over three sources. Factor columns give partial geometry; Ratio is between/within separation. Native h16 errors use each source's own embedding target.

Condition	**Mass ↑**	**Drag ↑**	**Stiff. ↑**	**Ratio ↑**	**h16 ↓**
Structure	0.058 ± 0.005	0.191 ± 0.005	0.070 ± 0.011	0.299 ± 0.006	0.404 ± 0.033
Align	0.012 ± 0.009	0.781 ± 0.004	0.016 ± 0.005	4.203 ± 0.289	0.386 ± 0.019
Cross	0.107 ± 0.009	0.229 ± 0.009	0.106 ± 0.002	0.362 ± 0.009	0.373 ± 0.005
Align + Cross	0.006 ± 0.004	0.788 ± 0.001	0.011 ± 0.002	4.411 ± 0.142	0.381 ± 0.010

The observed G_1/G_2 accessibility contrast therefore compares equal-sized training supports with different candidate composition and preselection. A literal reduction of G_1 to G_2's training support size leaves three candidates unchanged. Candidate composition and donor-exposure frequencies follow the pairing rule; equal support does not separate them from shared-factor semantics.

Table 34: **Semantic eligibility and the fixed training support.** Counts are per recipient. Eligible systems satisfy the relation; training uses donors from only three preselected systems.

Split	**Relation**	**Eligible systems**	**Selected systems**	**Rollouts/system**	**Supported histories**
Training	G_1	173–174	3	4	12
Training	G_2	19	3	4	12
Validation	G_1	35–36	3	4	12
Validation	G_2	3	3	4	12

E.2 Objective components and reference learners: SpringWorld

Alignment-containing recipes improve cold-start prediction and cross-history organization. Task error uses one separately fitted reader per source recipe (reader seed 0) on 100 systems and 600 cases. Geometry uses frozen P_{64} before reader fitting on 178 systems with four histories each.

Align accounts for most of the original cold-start gain. Adding Cross to Align lowers error in all three sources, by 23.04%, 8.27% and 0.10%. Geometry uses train-standardized codes and partial Spearman factor-distance correlations, controlling the other two factors (k/m controls drag). Larger ratios indicate greater between-system separation relative to within-system variability. Full source-level grids are in the Supplementary Evidence Package.

Supervised reference encoders use factor-related supervision and serve as references rather than competitors. The question for the predictive learners is whether they organize this information with relation assignments but without regressing the numerical factor labels.

Supervised reference encoders. Table 36 compares GRU, Transformer, TCN and DeepSets using one source and reader per recipe. The supervised TCN has the lowest development error in this reference grid. The three-source objective comparison is in Table 35.

Table 35 reports task error and source geometry for the two cohorts.

The pooled Cross addition to Align is 10.37%; adding Align to Cross gives 34.10%. Align + Cross has greater source separation than Structure in all three geometry replicas.

Table 35: **Align accounts for most of the SpringWorld gain; Cross lowers error further.** Three-source means from separate cohorts: A, cold-start error on 100 development systems with one reader per source; B, frozen-source geometry on 178 systems. Ratio averages the source-wise between/within ratios.

A. Three-source component mean (reader 0)

	Native	Structure	Align	Cross	Align + Cross
Mean	0.1703	0.1716	0.1232	0.1675	0.1104

B. Frozen-source geometry

Recipe	Within	Between	Ratio	ρ_m	ρ_γ	ρ_k	$\rho_{k/m}$
Structure	131.6	31.64	0.240	0.001	0.005	-0.014	-0.025
Align	47.25	90.9	2.045	0.361	0.224	0.132	0.414
Cross	115.7	29.21	0.252	0.200	0.184	0.056	0.066
Align + Cross	40.58	93.63	2.360	0.354	0.193	0.141	0.421

Table 36: **SpringWorld development references and physical readout.** A: single-source standardized MSE on 100 cold-start and 178 query-mixture systems; supervised references use their own training recipes. B: training-fitted ridge R^2 for log factors on 100 validation systems. These are separate from the sealed and multi-reader evaluations.

A. Single-source reference (source 0, reader 0)

Source	Cold start	Query mixture
Controlled JEPA objectives		
Native	0.1716	0.8511
Structure	0.1711	0.8468
Align	0.1195	0.8033
Cross	0.1646	0.8443
Align + Cross	**0.0920**	**0.7892**
Other reference learners		
CPC	0.2218	0.9858
RSSM	0.1653	0.7882
Supervised split	0.1841	0.8864
Supervised reference encoders		
GRU	0.0934	0.6328
Transformer	0.0887	0.6631
TCN	0.0593	0.5948
DeepSets	0.0892	0.6597

B. Frozen-source physical probes

Source	m	γ	k	k/m
Native	0.0076	-0.0234	-0.0265	-0.0085
Align + Cross	0.3806	0.1529	0.2023	0.4945

E.3 Relation-selective tactile organization: Baxter

Hardness and shape relations selectively organize real tactile histories. The crossed bank contains cube/cylinder objects at three common hardness levels; confirmation holds out grasps from these known configurations.

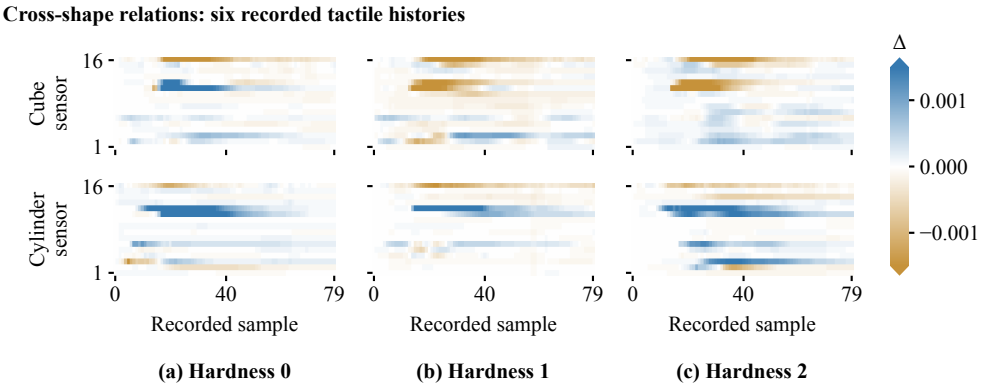


Figure 13: **Shared hardness connects tactile histories across object shapes.** Columns share hardness levels 0–2; rows differ in shape. Sensor changes are relative to the initial sample, with a common symmetric scale saturated at the pooled 97th percentile of absolute change.

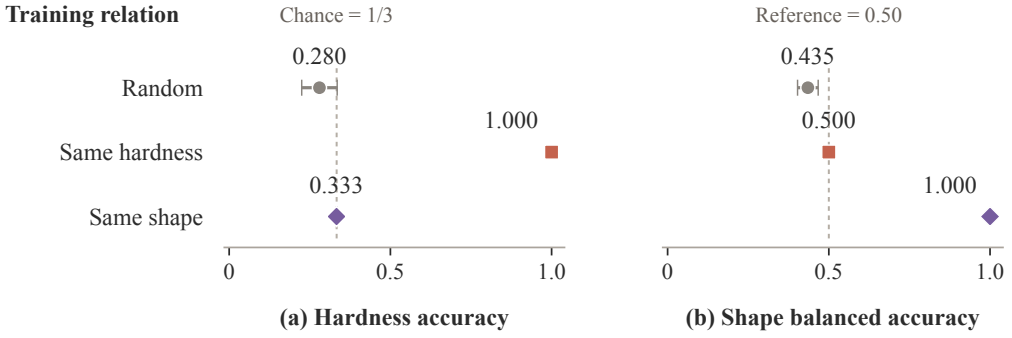


Figure 14: **Relation choice selects which tactile property becomes accessible.** Means and sample SD over three sources on held-out grasps from six known configurations. (a) Cross-shape hardness accuracy; (b) shape balanced accuracy with one hardness level held out. Dashed lines mark the respective chance references, $1/3$ and 0.50.

The input recoverability test reaches hardness accuracy 0.954 ± 0.003 using eight samples and 1.000 ± 0.000 using the full peak; shape is already perfect at both lengths. Thus, the input already supports both classifications.

The key comparison is between the two relation-trained models: hardness accuracy is 1.000 under $\mathcal{R}_H$ versus 0.333 under $\mathcal{R}_S$; shape balanced accuracy is 1.000 under $\mathcal{R}_S$ versus 0.500 under $\mathcal{R}_H$. The corresponding contrasts $C_H = 0.667$ and $C_S = 0.500$ are positive in every source on validation and confirmation. This reversal shows that the pairing rule selects which property is accessible within the six known configurations; Random provides an additional unstructured-pairing reference.

Forty-sample files are alternate windows from the same acquisitions.

E.4 Structured relation errors

This three-source PokeWorld comparison separates correct-pair frequency from the structure of pairing errors. The conditions are correct pairing; random mismatch with $\alpha = 0.5$; drag-only mismatch with $\alpha = 0.5$; and mass-only mismatch with $\alpha = 0.5$. Structured errors preserve the other two physical factors. All four conditions use a common training bank with legal exact single-factor mismatches, and share initialization, recipient draws, correct/error masks, source budgets

and evaluation protocol within seed. All twelve Align-only sources ($\lambda_p = 1$, $\lambda_x = 0$) were trained for 20,000 updates on this new 2,000-system bank, with source seeds 0–2; no source from the earlier fidelity bank is reused. Probe fitting uses 200 training systems, and evaluation uses 200 common validation systems with four rollouts and four history anchors each. All results below are development-only. The finite-bank mismatch kernel selects a distinct legal donor; it differs from independent factor resampling in Appendix I.2.

Partial geometry and training-fitted probe R^2 quantify each factor's organization and accessibility. The between/within ratio measures system separation. Differences from correct pairing are computed within source. The corollary characterizes alignment pressure for a fixed representation; the factor-specific response of a trained encoder is measured by these comparisons.

Table 37: **Factor measurements under structured relation errors.** Means and sample SD across three sources. Probe targets and partial correlations follow Appendix E.1's PokeWorld protocol.

Pairing	**Factor**	**Probe R^2**	**Partial geometry ρ**
Correct	Mass	0.1630 ± 0.0138	0.0796 ± 0.0105
Correct	Drag	0.6087 ± 0.0881	0.4665 ± 0.0461
Correct	Stiffness	0.0589 ± 0.0432	0.0168 ± 0.0186
Random mismatch	Mass	0.2229 ± 0.0274	0.1954 ± 0.0013
Random mismatch	Drag	0.3987 ± 0.0305	0.3092 ± 0.0043
Random mismatch	Stiffness	-0.0938 ± 0.1124	0.0806 ± 0.0024
Drag-only mismatch	Mass	0.3377 ± 0.0457	0.2978 ± 0.0152
Drag-only mismatch	Drag	0.3608 ± 0.0682	0.1162 ± 0.0224
Drag-only mismatch	Stiffness	0.0970 ± 0.0451	0.1312 ± 0.0320
Mass-only mismatch	Mass	0.1766 ± 0.1025	0.0180 ± 0.0064
Mass-only mismatch	Drag	0.9105 ± 0.0435	0.7013 ± 0.0186
Mass-only mismatch	Stiffness	0.1166 ± 0.0127	-0.0129 ± 0.0026

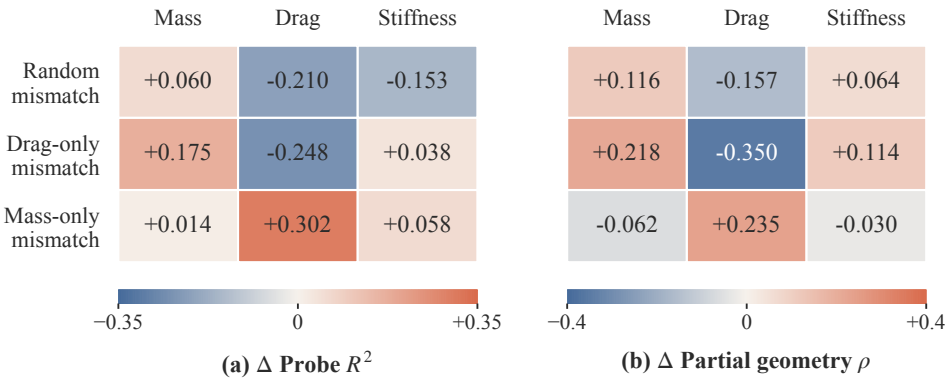


Figure 15: **Structured mismatches redistribute factor organization.** Mean paired-source changes from correct pairing in (a) probe R^2 and (b) partial geometry ρ; positive indicates an increase. All mismatch rules use $\alpha = 0.5$ and three Align-only sources on 200 development validation systems. Mass-only mismatch lowers mass geometry without lowering its mean probe score. Source SDs are in Table 37.

Drag-only mismatch changes drag geometry from 0.4665 to 0.1162 and mass geometry from 0.0796 to 0.2978. Mass-only mismatch changes mass geometry to 0.0180 and drag geometry to 0.7013. All four directions hold in each of the three source seeds. Accessibility is less uniform: mass-only mismatch does not lower mass probe R^2 (0.1766 versus 0.1630 for correct pairing). These results support factor-directed organization under structured errors. Correct pairing does not minimize native latent MSE at any evaluated horizon; the prediction panel reports each frozen encoder's own target space and scale, and does not measure common physical-state error or control return.

Table 38: **Pairing quality, system separation and native prediction.** Coverage is the fraction of recipients with a legal donor; factor columns give match rates, and separation is the between/within ratio. Native latent MSE uses each encoder's own target space, with target variance in brackets. Errors and ratios show means and sample SD across three sources.

Pairing	Coverage (%)	Mass (%)	Drag (%)	Stiffness (%)	Separation
Correct	100	100	100	100	1.0601 ± 0.1093
Random mismatch	100	54.97	54.99	51.76	0.9694 ± 0.0236
Drag-only mismatch	100	100	50	100	1.1618 ± 0.1712
Mass-only mismatch	100	50	100	100	3.8995 ± 1.3415

Pairing	$H = 1$	$H = 4$	$H = 16$
Correct	0.0719 ± 0.0071 [0.6226]	0.1103 ± 0.0054 [0.6267]	0.2312 ± 0.0040 [0.6285]
Random mismatch	0.0108 ± 0.0008 [0.7416]	0.0506 ± 0.0010 [0.7439]	0.2126 ± 0.0015 [0.7457]
Drag-only mismatch	0.0265 ± 0.0034 [0.7061]	0.0688 ± 0.0045 [0.7075]	0.2216 ± 0.0038 [0.7083]
Mass-only mismatch	0.0223 ± 0.0004 [0.6841]	0.0675 ± 0.0043 [0.6885]	0.2157 ± 0.0041 [0.6966]

E.5 Relations inferred from learned histories

A Structure teacher trained without relation supervision encodes one 24-frame history per training rollout. Each source seed defines a frozen graph over 8,000 rollouts from the 2,000-system bank. After per-coordinate standardization on training codes, each rollout receives three Euclidean nearest neighbors, excluding itself. Physical labels and system grouping are not used to construct or select the graph. RandomHistory draws three neighbors from the same candidate universe. The graphs are fixed before source training; physical labels are used only for subsequent diagnostics.

Inferred and RandomHistory each use three newly trained Align-only sources ($\lambda_p = 1$, $\lambda_x = 0$), with 20,000 updates per source. True reuses three G_3 sources from the same bank with the same source hyperparameters.[1] All nine models use the common evaluation protocol: probes are fitted on 200 training systems, and evaluation uses 200 validation systems with 3,200 windows per model. No sealed data enter this analysis. The earlier α curve uses a different bank and is a descriptive reference, not an interpolation rule for these graphs.

Table 39: **Agreement and coverage of history-inferred relations.** Match rates and coverage are percentages, averaged over three source-specific graphs. Donor usage is donor-rollout entropy normalized by $\log 8000$. True uses exact same-system relations.

Relation	Tuple p	Mass	Drag	Stiffness	Coverage	Donor usage
True	100	100	100	100	100	1.000
Inferred	0.0639	11.15	13.30	3.64	100	0.974
RandomHistory	0.0431	10.04	10.01	3.50	100	0.980

Table 40: **Factor accessibility and geometry with inferred relations.** Entries are means and sample SD across three sources.

Relation	Factor	Probe R^2	Partial geometry ρ
True	Mass	0.1700 ± 0.0154	0.0737 ± 0.0092
True	Drag	0.6556 ± 0.0791	0.4807 ± 0.0877
True	Stiffness	0.0670 ± 0.0020	0.0314 ± 0.0453
Inferred	Mass	0.0538 ± 0.0423	0.0360 ± 0.0069
Inferred	Drag	0.1563 ± 0.0316	0.0061 ± 0.0111
Inferred	Stiffness	0.0314 ± 0.0111	0.0666 ± 0.0028
RandomHistory	Mass	0.0911 ± 0.0318	0.0256 ± 0.0117
RandomHistory	Drag	0.1065 ± 0.0870	0.0342 ± 0.0043
RandomHistory	Stiffness	-0.0235 ± 0.0202	0.0323 ± 0.0255

[1] The reused G_3 sources were trained on L20 GPUs with PyTorch 2.6; the new graph-based sources used RTX PRO 6000 GPUs with PyTorch 2.8.

Table 41: **System separation and native prediction with inferred relations.** Means and sample SD across three sources. The bracketed second line gives mean target variance. Each source defines its own latent prediction target.

Relation	Between/within	Native $H = 1$	Native $H = 4$	Native $H = 16$
True	1.0683 ± 0.2664	0.0721 ± 0.0103	0.1124 ± 0.0042	0.2327 ± 0.0099
		[0.6195]	[0.6225]	[0.6249]
Inferred	0.2577 ± 0.0013	0.0326 ± 0.0031	0.0743 ± 0.0043	0.2357 ± 0.0051
		[0.7309]	[0.7366]	[0.7482]
RandomHistory	0.2499 ± 0.0117	0.0121 ± 0.0076	0.0542 ± 0.0144	0.2268 ± 0.0051
		[0.7405]	[0.7419]	[0.7436]

For a shared-space score T, recovery denotes the fraction of the True–RandomHistory gap attained by Inferred: $(T_{\text{Inferred}} - T_{\text{RandomHistory}})/(T_{\text{True}} - T_{\text{RandomHistory}})$. Ratios use source means and are not clipped. We report them only where the True reference exceeds RandomHistory in every paired-source resample. Native latent errors are retained as diagnostics, not common physical-task errors or recovery scores.

Table 42: **Inferred-relation changes and recovery relative to True.** Δ is Inferred minus RandomHistory. Recovery intervals are 95% paired-source bootstrap intervals over three seeds; they do not measure cross-task stability. Dashes denote inapplicable ratios.

Measurement	Factor	Δ	Recovery (%)
Probe R^2	Mass	-0.0373	$-47.3\,[-135.5, 1.0]$
Probe R^2	Drag	$+0.0498$	$9.1\,[-2.7, 30.5]$
Probe R^2	Stiffness	$+0.0549$	$60.7\,[55.3, 70.0]$
Geometry ρ	Mass	$+0.0104$	$21.6\,[-27.6, 53.4]$
Geometry ρ	Drag	-0.0281	$-6.3\,[-8.6, -4.2]$
Geometry ρ	Stiffness	$+0.0343$	—[a]
Between/within ratio	—	$+0.0077$	$0.9\,[-0.5, 2.4]$
Native h16 error	—	$+0.0090$	—[b]

[a] The True–RandomHistory stiffness-geometry gap crosses zero under source resampling; no consistent True advantage defines a recovery scale.
[b] The native models use different latent target spaces.

The inferred graph modestly enriches drag matches (13.30% versus 10.01%) and mass matches (11.15% versus 10.04%). Probe effects remain factor-dependent: mean stiffness R^2 rises from -0.0235 to 0.0314, with increases in all three sources; drag rises by 0.0498 on average but improves in only one source, while mass falls by 0.0373 on average. Drag geometry falls in all three sources. True has higher probe R^2 for every factor and source. Thus, this fixed history-based rule changes selected accessible factors but does not recover the effect of the supplied physical relations.

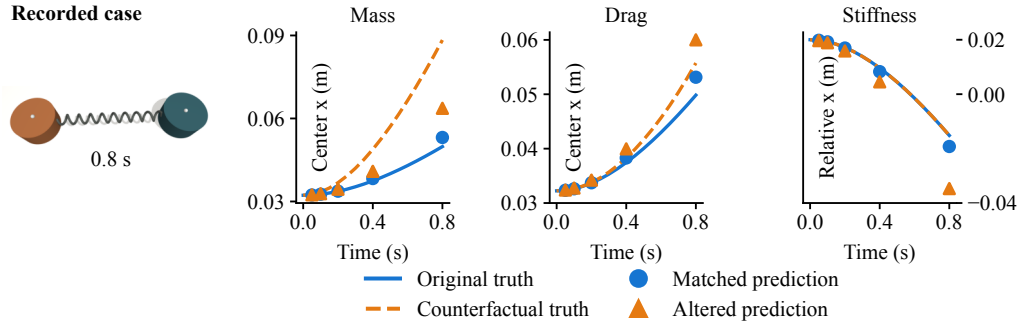
F USE: FIXED-ROUTE CONTEXT INTERVENTIONS

This section expands Figure 4: SpringWorld provides physical donor landscapes, PokeWorld tests factor-specific direction, and RH20T and Balls examine recorded task context and actual reader memory. Swimmer tests recipient-specific updates through frozen predictor weights. Probes measure accessibility; fixed-route interventions measure dependence on supplied context.

F.1 PHYSICAL DONOR LANDSCAPES: SPRINGWORLD

Changing donor physics redirects the fixed reader toward the corresponding target dynamics. The assay uses 64 continuous development systems, three frozen sources and reader seed 0. For each factor, five donor levels and five simulator target levels share initial states and actions; the two other physical parameters remain fixed. A fixed rule selects the query without using model error. All 5×5 grids are complete; h16 is primary.

Valley depth is mean off-diagonal error minus mean diagonal error, averaging 20 and five cells respectively. Models use the original training-only target normalization. Systems then sources receive equal weight; intervals use 10,000 paired-system draws conditional on these fitted routes.



(a) Physical substitutions at a fixed recipient state

Recipe	Mass	Drag	Stiffness
Structure	11.79 [5.73, 19.00]	3.10 [1.56, 4.85]	0.16 [-0.14, 0.55]
Align	192.24 [140.24, 247.06]	31.24 [21.98, 41.52]	23.83 [15.86, 33.53]
Cross	31.97 [18.37, 48.07]	8.86 [5.08, 13.44]	0.33 [-0.66, 1.42]
A+C	182.05 [132.00, 234.29]	30.33 [21.58, 39.97]	23.95 [15.63, 33.83]

(b) Population matching advantage: mean [95% CI], MSE × 10^{-3}

Figure 16: **Physical donor substitutions redirect predictions and produce error valleys.** (a) One preselected case: lines show simulator truth and markers show predictions at horizons 1/2/4/8/16. (b) Valley depths for four separately fitted recipes on 64 systems, with paired-system 95% intervals. Align + Cross uses the pooled intervals in Table 43.

Align + Cross has positive valley depth for all three factors in every source. The balanced factorial main effect of Align, $(\text{Align} - \text{Structure} + \text{Align} + \text{Cross} - \text{Cross})/2$, is 0.16526, 0.02480 and 0.02365 for mass, drag and stiffness, with respective reported intervals [0.12092,0.21175], [0.01720,0.03286] and [0.01573,0.03306]; the corresponding Cross main effects 0.00499, 0.00243 and 0.00015 have intervals spanning zero. Each contrast compares complete source/reader recipes; donor substitutions within each recipe keep its fitted reader fixed.

Output-coordinate decomposition. The equal-mass center/relative transform decomposes the frozen reader's physical output response. Simulator relative shares are 0.7143/0.7149/1.0000 for mass/drag/stiffness; reader shares span 0.610–0.700/0.650–0.689/0.637–0.696 across sources. The stiffness relative-minus-center statistic is 0.00155107238, quantifying the distribution of the response across physical output coordinates.

The center/relative transform acts on the position and velocity coordinates $(x_1, y_1, x_2, y_2, v_{x1}, v_{y1}, v_{x2}, v_{y2})$.

Table 43: **Align + Cross yields positive donor-matching valleys in every source.** Depth is off-diagonal minus diagonal h16 error on 64 development systems. Source results and recipe comparisons use three frozen sources and one reader per source. Paired-system 95% intervals are conditional on fitted models.

Factor	Source	Valley depth	95% interval	ρ
Mass	Pooled	0.1820	$[0.1320, 0.2343]$	0.591
	Seed 0	0.1950	$[0.1379, 0.2558]$	0.606
	Seed 1	0.1612	$[0.1131, 0.2134]$	0.592
	Seed 2	0.1899	$[0.1364, 0.2462]$	0.601
Drag	Pooled	0.0303	$[0.0216, 0.0400]$	0.585
	Seed 0	0.0334	$[0.0238, 0.0439]$	0.600
	Seed 1	0.0335	$[0.0229, 0.0458]$	0.578
	Seed 2	0.0240	$[0.0159, 0.0327]$	0.542
Stiffness	Pooled	0.0240	$[0.0156, 0.0338]$	0.672
	Seed 0	0.0285	$[0.0184, 0.0407]$	0.776
	Seed 1	0.0229	$[0.0143, 0.0328]$	0.401
	Seed 2	0.0205	$[0.0124, 0.0299]$	0.628

Table 43 (continued). Recipe comparison.

Recipe	Mass: depth [95% CI]	Drag: depth [95% CI]	Stiffness: depth [95% CI]
Structure	0.01179 $[0.00573, 0.01900]$	0.00310 $[0.00156, 0.00485]$	0.00016 $[-0.00014, 0.00055]$
Align	0.1922 $[0.1402, 0.2471]$	0.03124 $[0.02198, 0.04152]$	0.02383 $[0.01586, 0.03353]$
Cross	0.03197 $[0.01837, 0.04807]$	0.00886 $[0.00508, 0.01344]$	0.00033 $[-0.00066, 0.00142]$
Align + Cross	0.182 $[0.132, 0.2343]$	0.03033 $[0.02148, 0.03983]$	0.02395 $[0.01539, 0.03398]$

The landscape figure uses the upper pooled intervals; this panel uses the four-recipe intervals.

F.2 Factor-specific responses and direction: PokeWorld

Takeaway. Shared-factor relations redirect frozen predictive responses toward the corresponding mass or drag counterfactual direction.

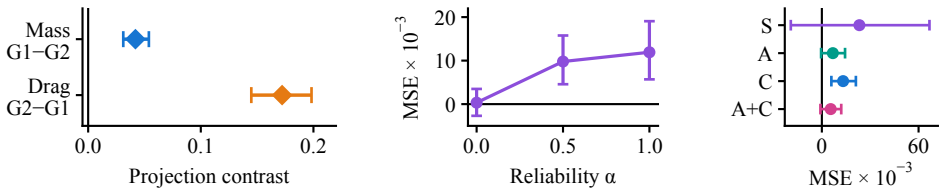
Physical-risk and native-embedding assays measure this response in different spaces. The 96-system physical-risk assay in Figure 17 holds the native predictor and training-only ridge state decoder fixed. Matched histories come from an independent recipient interaction; a wrong-factor donor changes exactly one physical parameter. Three hash-fixed windows per system retain current observations, actions, target and h16. All three source seeds use common training-target normalization; Δ_f is wrong-factor minus matched four-coordinate object-state MSE. The principal direction assay evaluates step-20,000 Align G_1/G_2 checkpoints (three sources per relation) on 144 eligible validation systems with three queries each. It uses each source's native 128-dimensional observation embedding without fitting a new encoder, reader or decoder. For factor f and relation G, let $\Delta\hat{z}_{f,s}$ be the wrong-donor minus matched prediction displacement and $\Delta z^{\mathrm{cf}}_{f,s}$ the paired counterfactual target displacement in the native embedding space. We define

$$P(f, G) = \frac{\sum_s \langle \Delta\hat{z}_{f,s}, \Delta z^{\mathrm{cf}}_{f,s} \rangle}{\sum_s \|\Delta z^{\mathrm{cf}}_{f,s}\|_2^2},$$

pooling within a physical system and then giving source seeds equal weight. The reported contrasts are

$$\Delta_m = P(m, G_1) - P(m, G_2) = 0.041969, \qquad 95\%\ \mathrm{CI} = [0.031176, 0.053899],$$
$$\Delta_\gamma = P(\gamma, G_2) - P(\gamma, G_1) = 0.172192, \qquad 95\%\ \mathrm{CI} = [0.144846, 0.198290].$$

Reported intervals use paired physical-system bootstrap. Their sum is 0.214161 $[0.184608, 0.243162]$. The already fitted state-decoder version reports the same directions at a smaller scale: $\Delta_m = 0.004986$ and $\Delta_\gamma = 0.020592$. The primary statistic is a dimensionless projection in each model's native 128-D embedding. Both contrasts show relation-specific functional redirection on this development bank; the state-decoder assay expresses that response in state coordinates.



(a) Native direction · 144 systems **(b) Physical use · 96 systems**

Relation	Mass	Drag	Stiffness
G1	5.27 [-0.79, 12.63]	3.36 [-1.30, 8.32]	2.84 [-1.60, 8.29]
G2	1.35 [-1.11, 4.30]	2.75 [-5.07, 10.89]	0.23 [-1.33, 1.73]
G3	0.94 [-1.69, 3.66]	7.20 [1.67, 13.54]	2.59 [-0.34, 5.99]

(c) Complete physical-risk contrasts · mean [95% CI] × 10^{-3}

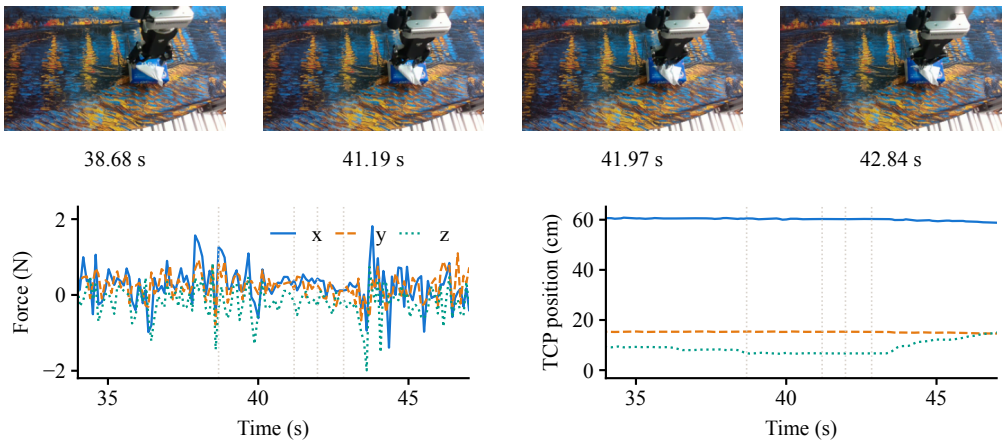
S: Structure; A: Align; C: Cross.

Figure 17: **Shared-factor relations redirect predictive responses.** (a) Mean G_1-minus-G_2 mass and G_2-minus-G_1 drag projections on 144 systems. (b)–(c) Wrong-factor minus matched h16 object-state MSE on a separate 96-system bank. Pointwise 95% intervals resample systems with three sources fixed (and state decoders fixed in (b)–(c)). Yellow marks an interval above zero. Native projections and physical MSE have distinct units.

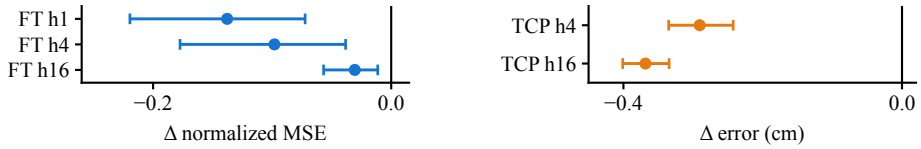
F.3 Task-specific donor use in recorded interactions: RH20T

RH20T relations join different episodes sharing task metadata. Three evaluations test different aspects of the resulting context: a 25-task forecasting cohort compares matched-donor prediction; a 24-task hard-negative bank compares donor specificity and absolute risk; and a separate 25-task validation bank tests action use and donor-count sensitivity.

Matched-donor forecasting. Cross lowers all five endpoint means relative to Structure under matched-donor evaluation. The models jointly predict force/torque and TCP from synchronized visual, state and action histories. Table 44 reports all 14 conditions and three fitted seeds.



(a) Recorded interaction and synchronized measurements



(b) Matched forecasting: Cross − Structure (25 held-out tasks)

Figure 18: **Cross reduces all five matched-donor forecasting endpoint means.** (a) RH20T frames and synchronized force/TCP observations. (b) Cross-Indep minus Structure on 25 held-out tasks; negative favors Cross. Force/torque uses normalized MSE and TCP uses cm. Task-paired 95% intervals follow averaging over three sources. Absolute risks on the separate 24-task bank are in Table 45.

Table 44: **Self-context and matched-donor forecasting on RH20T.** Equal-weight means over 25 held-out tasks and three sources. Force/torque uses normalized MSE; TCP uses cm. LD denotes LowDim; A+C denotes Align + Cross. Dashes indicate unavailable matched inputs. Shading highlights the Structure/Cross comparison.

A. Self-context

	Force/torque MSE			TCP (cm)	
Condition	**h1**	**h4**	**h16**	**h4**	**h16**
Native	0.6346	1.117	1.497	1.962	4.063
Structure	**0.6199**	**1.099**	**1.498**	**1.944**	**3.965**
Align–Indep	0.6633	1.132	1.503	2.262	4.429
Align–Random	0.6232	1.106	1.510	2.045	4.257
Cross–SameEp	0.6145	1.090	1.495	1.963	3.965
Cross–Indep	**0.6230**	**1.093**	**1.497**	**1.901**	**3.902**
Cross–Random	0.6307	1.096	1.495	1.948	3.950
A+C–SameEp	0.6407	1.120	1.500	2.098	4.371
A+C–Indep	0.6602	1.130	1.505	2.268	4.435
A+C–Random	0.6225	1.101	1.507	2.036	4.258
LD–Native	0.5909	1.089	1.489	1.730	3.887
LD–Cross–Indep	0.5760	1.069	1.488	1.598	3.748
LD–A+C–SameEp	0.6162	1.118	1.504	2.032	4.302
LD–A+C–Random	0.6090	1.122	1.505	1.866	4.191

B. Matched donor

	Force/torque MSE			TCP (cm)	
Condition	**h1**	**h4**	**h16**	**h4**	**h16**
Native	–	–	–	–	–
Structure	**0.7991**	**1.226**	**1.551**	**2.429**	**4.952**
Align–Indep	0.6667	1.135	1.504	2.536	4.644
Align–Random	0.6255	1.108	1.510	2.062	4.274
Cross–SameEp	0.6709	1.140	1.527	2.254	4.745
Cross–Indep	**0.6612**	**1.128**	**1.521**	**2.139**	**4.584**
Cross–Random	0.6665	1.127	1.516	2.158	4.604
A+C–SameEp	0.6498	1.127	1.506	2.407	4.578
A+C–Indep	0.6633	1.133	1.506	2.573	4.670
A+C–Random	0.6250	1.103	1.507	2.045	4.269
LD–Native	–	–	–	–	–
LD–Cross–Indep	0.6039	1.131	1.531	1.893	4.471
LD–A+C–SameEp	0.6187	1.121	1.506	2.078	4.339
LD–A+C–Random	0.6108	1.124	1.506	1.905	4.230

Donor specificity and absolute risk. The 24-task bank contains 10,681 windows with matched, action-metadata-matched/different-task, calibration-matched/different-task and random donor eligibility. Align + Cross and Monolithic Q+D share the manifest, 48-pair batch, 20,000-step checkpoint and three seeds; parameter counts differ by 0.309%. Define

$$S_f = E_f(\text{hard}) - E_f(\text{matched}), \qquad D_{\text{spec}} = S_{\text{A+C}} - S_{\text{Mono}}. \tag{13}$$

Align + Cross has greater FT donor specificity, while Monolithic has lower matched error. Table 45 reports both comparisons and their paired intervals; the TCP matched-error interval favors Monolithic.

Table 45: **Donor specificity and absolute risk on the 24-task bank.** Means weight tasks equally after averaging three sources, with task-paired 95% intervals. Specificity is hard-minus-matched error; contrasts are A+C minus Monolithic. Positive risk differences favor Monolithic. FT h4 is primary; TCP h16 is secondary.

A. Donor specificity

Endpoint	$S_{\text{A+C}}$	S_{Mono}	D_{spec}
FT h4 MSE	0.00260 [0.00107, 0.00429]	0.00030 [−0.00050, 0.00134]	**0.00230** [0.00073, 0.00404]
TCP h16 (cm)	0.0891 [0.0209, 0.1614]	0.0176 [−0.0025, 0.0414]	0.0715 [−0.0042, 0.1480]

B. Absolute matched and hard-donor risk

Endpoint	Model	Matched	95% interval	Hard	95% interval
FT h4	Align+Cross	0.3343	[0.2501, 0.4262]	0.3369	[0.2518, 0.4300]
FT h4	Monolithic	0.3205	[0.2390, 0.4104]	0.3208	[0.2391, 0.4105]
TCP h16 (cm)	Align+Cross	4.899	[4.355, 5.482]	4.989	[4.446, 5.561]
TCP h16 (cm)	Monolithic	4.349	[3.925, 4.786]	4.367	[3.948, 4.800]

C. A+C minus Monolithic risk

Endpoint	Donor	Difference	95% paired interval
FT h4	matched	0.0138	[−0.0002, 0.0345]
FT h4	hard	0.0161	[0.0028, 0.0363]
TCP h16 (cm)	matched	0.5503	[0.3699, 0.7485]
TCP h16 (cm)	hard	0.6218	[0.4302, 0.8282]

Action use and donor-count controls. The additional 25-task validation bank holds checkpoints fixed while zeroing actions, replacing donors, or averaging $K = 1, 2, 4$ histories. Task-paired differences first average three seeds, then use 10,000 task draws. Align + Cross has positive donor penalties for FT and TCP; Cross has positive FT but unresolved TCP penalties. The Align + Cross TCP donor effect decreases with donor count, while Random retains a small positive FT effect. Ablation establishes action use. Table 46 lists action, donor and history-count effects.

Table 46: **Action use and donor-count sensitivity on a separate 25-task bank.** A: action-zero minus observed, and shuffled-minus-correct donor effects. B: random-minus-correct effects at fixed donor count K and fixed weights. Pointwise 95% task-bootstrap intervals follow averaging three sources. FT uses MSE; TCP uses cm.

A. Action and donor interventions

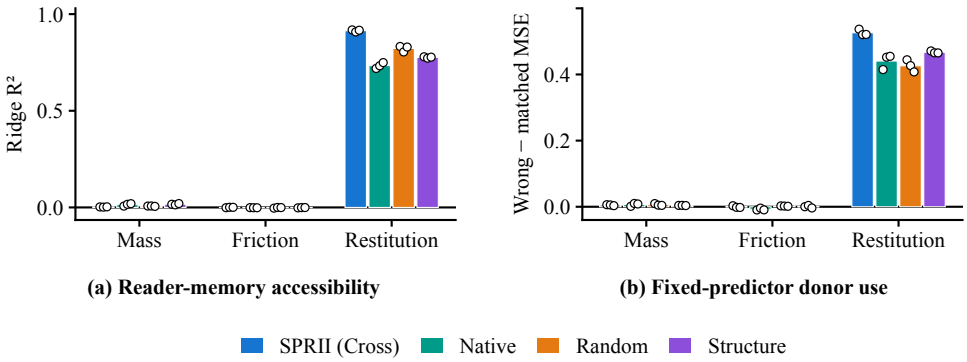
Intervention/model	Metric	Difference	95% task interval
A+C: action zero	FT h4	0.002074	[0.000740, 0.003650]
A+C: action zero	TCP h4	0.06439	[0.04507, 0.08274]
A+C: action zero	TCP h16	0.1399	[0.07687, 0.2025]
Align + Cross	FT h4	0.009692	[0.004828, 0.01624]
Align + Cross	TCP h4	0.4705	[0.2801, 0.6948]
Align + Cross	TCP h16	0.3385	[0.1768, 0.5304]
Cross	FT h4	0.01711	[0.009042, 0.02876]
Cross	TCP h4	0.004189	[−0.02059, 0.02922]
Cross	TCP h16	0.01571	[−0.0472, 0.07335]

Table 46 (continued). B. Donor-count sensitivity at fixed weights

Model	K	FT h4 MSE	TCP h4 (cm)	TCP h16 (cm)
Align + Cross	1	0.01019 [0.00476, 0.01786]	0.4686 [0.2741, 0.6968]	0.3368 [0.1719, 0.5327]
Align + Cross	2	0.00874 [0.00325, 0.01648]	0.3652 [0.1675, 0.6004]	0.2614 [0.09066, 0.4618]
Align + Cross	4	0.00794 [0.00240, 0.01566]	0.2978 [0.08930, 0.5383]	0.2191 [0.04070, 0.4249]
Cross	1	0.01606 [0.00836, 0.02703]	0.00488 [−0.01979, 0.03002]	0.01573 [−0.04830, 0.07455]
Cross	2	0.01593 [0.00634, 0.02960]	0.00691 [−0.01343, 0.02892]	0.01599 [−0.02912, 0.05778]
Cross	4	0.01602 [0.00531, 0.03114]	0.00869 [−0.01117, 0.02953]	0.01166 [−0.02378, 0.04752]
Random	1	0.00071 [0.00017, 0.00133]	0.00162 [−0.00353, 0.00672]	0.00113 [−0.01151, 0.01315]
Random	2	0.00067 [0.00017, 0.00126]	0.00110 [−0.00355, 0.00570]	−0.00018 [−0.01234, 0.01100]
Random	4	0.00066 [0.00016, 0.00122]	0.00078 [−0.00368, 0.00521]	−0.00091 [−0.01283, 0.00794]

F.4 Physical information in reader memory: Balls

RSSM reader memory selectively exposes restitution, and fixed-predictor substitutions show corresponding donor use. The measured memory includes the trained support projection and is distinct from complete source state U and independent-support code S. Twelve RSSM fits (four recipes by three joint source/reader seeds) reproduce stored matched error before one-factor donor substitution.



(a) Reader-memory accessibility **(b) Fixed-predictor donor use**

Figure 19: **Restitution dominates reader-memory accessibility and donor effects.** (a) Training-fitted ridge R^2 on RSSM reader memory. (b) Wrong-one-factor minus matched MSE on 2,000 recipients through fixed weights. Bars average three source/reader fits; symbols show individual fits. SPRII uses Cross.

For the third SPRII (Cross) replica, wrong restitution changes MSE from 1.1823 to 1.7033; mass and friction changes are 0.00331 and −0.00209. Its within/between ratio is 1.059: within-system code variation remains slightly larger than between-system variation. Repeated-history and S3/S5/S8 inputs pass through the same fixed S3-trained head, measuring its sensitivity to available histories.

JEPA and CoPhyNet factor substitutions also favor restitution; their cohorts and replication counts are given with the corresponding results.

Actual-memory channels and fixed-reader substitutions. The CoPhyNet probes below measure actual reader memory M; source U and independent-support S are separate channels. Development-bank coverage is specified in Appendix B.6.

Table 47: **Distinct histories lower error through the fixed S3-trained RSSM reader.** MSE averages three fits; Repeat averages repetitions of each individual history. Reader weights remain fixed.

Recipe	Repeat	S3	S5	S8
SPRII (Cross)	1.29	1.171	1.156	1.147
Native	1.532	1.286	1.241	1.206
Random	1.497	1.356	1.333	1.316
Structure	1.517	1.288	1.246	1.218

Table 48: **Reader-memory probes, RSSM geometry and separate donor assays.** A: training-fitted CoPhyNet memory probes, including the learned projection. B: RSSM same-physics/wrong-physics code-distance ratios. C: separate single-source evaluations, reporting wrong-factor minus matched MSE. These panels measure distinct channels and cohorts.

A. CoPhyNet reader-memory probes (R^2)

Recipe	Factor	Source 1	Source 2	Source 3
SPRII (A+C)	Mass	0.0509	0.0546	0.0678
	Friction	0.0027	−0.0020	0.0002
	Restitution	0.8297	0.8266	0.8210
Native	Mass	0.0283	0.0271	0.0411
	Friction	−0.0014	0.0024	0.0009
	Restitution	0.7581	0.7599	0.7684
Random	Mass	0.0306	0.0262	0.0295
	Friction	−0.0025	−0.0043	0.0007
	Restitution	0.7611	0.7681	0.7598

B. RSSM within/between geometry

Recipe	Source 1	Source 2	Source 3
SPRII (Cross)	1.182	1.373	1.059
Native	3.904	1.769	2.493
Random	2.370	2.429	2.334
Structure	2.479	2.415	2.496

C. Separate source100 donor assay

Learner	Matched MSE	Δm	Δ friction	Δ restitution
JEPA	1.240	+0.000490	+0.001173	0.5140
CoPhyNet	1.006	+0.004422	−0.001568	0.5184

Source U, independent support S and actual reader memory M are distinct channels. A+C denotes Align + Cross. Per-fit history substitutions are indexed in the Supplementary Evidence Package; Table 47 gives history-count means.

F.5 Recipient-specific updates through frozen weights: Swimmer

Recipient-specific persistent updates reduce error for three frozen predictors. The 512 prospective systems are disjoint from training and selection; averages cover three adapter checkpoints, then 36 query/candidate rows per system. Four thousand paired-system draws quantify uncertainty.

Table 49: **Recipient-specific updates improve all three frozen Swimmer predictors.** Gain is wrong-system minus own-system update error in 10^{-5} units. Intervals are paired-system 95% bootstrap intervals on 512 new systems.

Frozen base	Gain	95% interval	Positive systems
Persistent-JEPA	5.329	[3.930, 6.724]	65.04%
Masked-GRU (64101)	6.930	[5.415, 8.621]	67.97%
Masked-GRU (64103)	6.661	[4.878, 8.486]	68.36%

Each base retains its architecture and training normalization. The within-base comparison measures the value of its recipient-specific update through fixed predictor weights.

Task benefit under different targets and fitted readouts is examined in Appendix G.

G VALUE: CONDITIONAL TASK BENEFIT AND READER REALIZATION

This section expands Figure 5: SpringWorld tests horizon and reader effects, D-Clean contrasts direct context with decoded parameters, and PokeWorld separates aggregate from component-level benefit. Population and replication details follow Appendix B.6.

G.1 PREDICTION HORIZON AND RECIPIENT EVIDENCE: SPRINGWORLD

History benefit grows toward longer horizons for both cold and moving queries in the 178-system common-adapter grid. Both query types have zero observed recipient transitions and differ in initial motion. The comparison spans four strata, three sources and three readers per source.

Table 50: **History benefit increases with prediction horizon.** Percentage error reduction of separately fitted Persistent versus Null readers, with pointwise stratified system-bootstrap 95% intervals.

Horizon	Cold: gain [95% interval]	Moving: gain [95% interval]
1	1.95 [−8.47, 11.01]	1.15 [−0.83, 3.01]
2	13.39 [3.93, 21.49]	1.98 [−0.22, 4.02]
4	23.19 [14.37, 30.35]	3.23 [0.91, 5.77]
8	35.20 [26.74, 41.92]	11.62 [7.17, 16.57]
16	35.68 [28.22, 42.27]	29.73 [22.68, 36.45]

The 36.10% cold-start and 7.11% full-mixture summaries use different populations of 100 and 178 systems, respectively. The following temporal-reader comparison varies recipient evidence under a separate protocol.

Recipient-prefix manipulation. A separate temporal reader with $r \in \{0, 1, 2, 4, 8\}$ does not materially change the recipient-only baseline. At h16, r0/r8 errors are 0.469305/.467214 for Null, 0.334363/.333367 for Persistent and 0.164647/.156172 for Oracle. The observed endpoint changes are small under this temporal-reader protocol.

The recipient-prefix reader uses 100 validation systems and three sources by three readers. Its endpoint intervals use 4,000 system draws and generally include zero.

G.2 DIRECT CONTEXT VERSUS PHYSICAL BOTTLENECKS

Takeaway. Direct context outperforms a decoded-parameter interface in D-Clean; the same comparison favors decoded parameters in SpringWorld.

Four fresh readers share the frozen Align + Cross query encoder: Null, Persistent, Decode and Oracle. Persistent and Decode receive exactly the same independent donor history; the train-fitted parameter decoder freezes before reader fitting. Each arm uses the same 10,000-step horizon-conditioned recipe, three sources and three readers, with 32 windows from each of 200 validation systems. This population is separate from the 100-system TDS comparison.

At h32, Persistent MSE is 0.000590019 versus 0.000763955 for Decode and 0.000223496 for Oracle. The paired Decode-minus-Persistent gap is 0.000173937 [0.000068237,0.000307406], positive in all nine cells. Independent validation of the frozen decoder gives $R^2 = 0.996715, 0.997340, 0.996993$ on systems 1000–1199. Accurate parameter estimation therefore need not supply a predictively equivalent learned interface.

SpringWorld's opposite ordering. The common-adapter comparison projects the same donor code either directly or through training-only ridge estimates of three log-parameters. All arms share query encoder, donor identities, sample order and target normalization; Decode/Oracle add only the

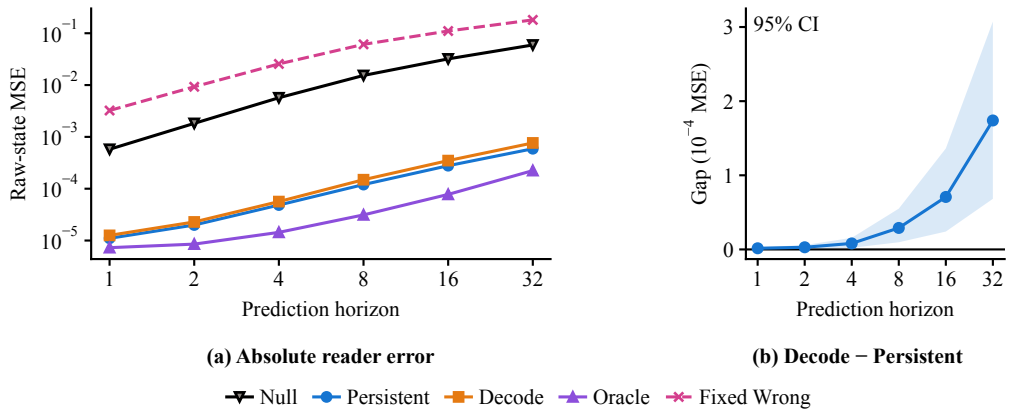


Figure 20: **Direct context outperforms the physical bottleneck across D-Clean horizons.** (a) Raw-state MSE (log scale); Fixed Wrong substitutes donors through the Persistent reader. (b) Decode-minus-Persistent MSE with pointwise system-bootstrap 95% intervals, conditional on fitted sources and readers. The primary endpoint is h32.

256-parameter input projection to the common reader. On the 100-system cold-start bank, Decode-minus-Persistent is -0.0023623 $[-0.0045732, -0.0004762]$: explicit physical parameters preserve and slightly improve realized value here. Persistent reduces Null error by 36.10%, and Persistent error is 2.12% higher than Decode error, using the same Decode-denominator convention as the D-Clean comparison. Table 52 gives actual-donor probes on 178 systems and separate full-mixture results.

Table 51: **Direct-context and bottleneck readers use the same D-Clean donors.** (A) Raw-state MSE; (B) pointwise paired-system 95% intervals conditional on the fitted source–reader grid; (C) independent validation of the frozen drag decoder. The primary endpoint is h32.

A. Absolute reader and intervention error

Horizon	Null	Persistent	Decode	Oracle	Fixed Wrong
1	0.000577	0.000011	0.000013	0.000007	0.003227
2	0.001821	0.000020	0.000023	0.000009	0.009263
4	0.005689	0.000048	0.000056	0.000014	0.025560
8	0.015320	0.000120	0.000149	0.000031	0.061030
16	0.031970	0.000278	0.000349	0.000078	0.110300
32	0.059360	0.000590	0.000764	0.000223	0.180900

B. Paired contrasts and intervention uncertainty

Horizon	Fixed Wrong 95% CI	D−P (10^{-4})	D−P 95% CI (10^{-4})
1	[0.002261, 0.004323]	0.0154	[0.0059, 0.0281]
2	[0.006822, 0.01199]	0.0291	[0.0076, 0.0586]
4	[0.01949, 0.03224]	0.0809	[0.0218, 0.1565]
8	[0.04752, 0.07559]	0.2898	[0.0965, 0.5504]
16	[0.08612, 0.1362]	0.7078	[0.2409, 1.3620]
32	[0.1433, 0.2213]	1.7390	[0.6824, 3.0740]

C. Independent frozen-decoder validation

Sampling seed	Validation R^2	γ MSE	95% MSE interval
0	0.9967	0.003653	[0.0027, 0.0052]
1	0.9973	0.002957	[0.0022, 0.0043]
2	0.9970	0.003344	[0.0025, 0.0046]

D−P is Decode minus Persistent. Fixed Wrong changes the Persistent input, without fitting a fifth reader. The grid contains 27 Null/Persistent/Oracle fits and nine Decode fits; no per-arm source SD is reconstructed.

Table 52: **SpringWorld reader error and donor-factor accessibility.** (A) Common-adapter MSE across three sources and three readers. (B) Training-fitted log-parameter probes on 712 donor histories from 178 validation systems, weighted equally by system.

A. Reader risk: cold-start and full-mixture populations

Population	Null	Persistent	Decode	Oracle
Cold start: 100 systems / 600 cases	0.17780	0.11360	0.11130	0.06188
Full mixture: 178 systems	0.87170	0.80970	0.80870	0.76590

B. Actual-donor accessibility: 178 systems

Source recipe	Seed	R^2_m	R^2_γ	R^2_k	$R^2_{k/m}$
Align + Cross	0	0.3918	0.1754	0.2050	0.4842
	1	0.3112	0.1537	0.1421	0.3774
	2	0.3569	0.1618	0.2345	0.5115
FCRL-style temporal	0	−0.0000	−0.0019	−0.0180	−0.0091
	1	0.0107	−0.0073	−0.0349	−0.0154
	2	0.0030	0.0013	−0.0134	−0.0056

Cold-start Decode-minus-Persistent is -0.0023623 $[-0.0045732, -0.0004762]$. The k/m estimate derives from fitted mass/stiffness coordinates. The nine fitted-reader rows remain in the Supplementary Evidence Package. The probe bank in B is distinct from the 100-system cold-start risk population in A.

G.3 Aggregate and component-level benefit: PokeWorld

Persistent context improves object-position and velocity prediction, with benefits that vary across output components. The R8 source study fits separate Null, Persistent, Shuffled and Oracle readers with the same architecture and training schedule. All 36 combinations use three sources by three reader seeds, 10,000 updates, and the reused 400-system development bank with 16 windows per system. Shuffled fits a separate reader using wrong-context inputs.

At the prespecified h16 endpoint, object-position and velocity errors fall, while finger-position error rises. The aggregate Null-minus-Persistent interval spans zero; Persistent improves on the separately trained Shuffled reader.

Actual donor-window drag readout is strong ($R^2 = 0.747$–0.783), compared with 0.131–0.164 for query-only inputs. The Oracle's 0.200% and Persistent's 0.057% aggregate improvements describe this learned reader family.

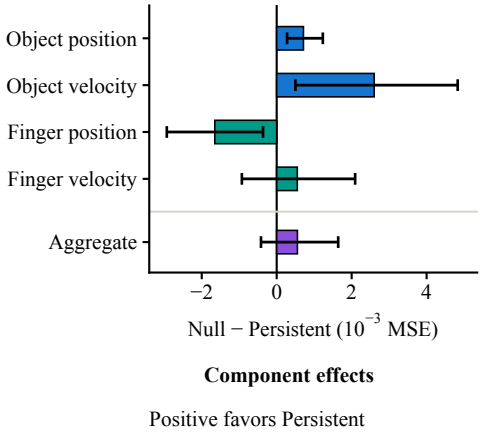


Figure 21: **Persistent context improves object-state prediction, with mixed effects on finger state.** Null-minus-Persistent MSE on the development bank; positive favors Persistent. Pointwise paired-system 95% intervals condition on the fitted source–reader grid; the aggregate interval adjusts for three contrasts. Components are not additive.

Factor-targeted utility. The 36-reader G_1/G_2 development grid pairs task gains with actual-donor physical probes. Its utility-versus-simulator-sensitivity slope is 0.002810 $[-0.004443, 0.009203]$, with positive slopes in each source. G_1 retains higher mass readout and G_2 higher drag readout in 5,913 actual donor windows. Table 54 gives the slopes and donor probes; bin details remain in the Supplementary Evidence Package. Additional population checks are grouped in Appendix G.4.

Aggregate, component and factor-targeted value. For targeted utility, $\Delta U = (E_N - E_P)_{G_1} - (E_N - E_P)_{G_2}$ is regressed against $d = z_{\text{train}}(S_m) - z_{\text{train}}(S_\gamma)$. Simulator sensitivities use central

Table 53: **Aggregate PokeWorld gains are small despite component-level benefits.** Separately fitted readers on 400 development systems; aggregate 95% intervals adjust for three contrasts. Oracle receives true parameters.

A. Aggregate h16 MSE

Arm	MSE
Null	0.9629
Persistent	0.9624
Shuffled	0.9637
Oracle	0.961

B. Prespecified aggregate contrasts

Contrast	Difference	Adjusted 95% interval
Null–Persistent	0.000553	$[-0.000419, 0.001641]$
Null–Oracle	0.001927	$[0.000427, 0.003615]$
Shuffled–Persistent	0.001367	$[0.000274, 0.002498]$

C. Oracle component reference: Null–Oracle

Component	Position	Velocity
Object	0.002809	0.006168
Finger	−0.003009	0.001740

$\pm 1\%$ perturbations, fixed query/actions and training factor/target scales. Actual-donor probes cover 5,913 distinct windows, equally weighting systems, with ridge coefficient 1.

Table 54: **PokeWorld factor-targeted utility and donor accessibility.** (A) The G_1-minus-G_2 history-benefit slope against simulator sensitivity has a system-bootstrap interval spanning zero. (B) Probe R^2 on actual donor windows, averaged across source seeds. The development grid includes the initial fits.

A. Continuous slope of ΔU against sensitivity dominance

Source seed	0	1	2
Expanded-grid slope	0.003838	0.003307	0.001284
Mean: 0.002810; 95% system-bootstrap interval: $[-0.004443, 0.009203]$			

B. Train-fitted donor-code probe, validation R^2

Recipe	$\log m$	$\log \gamma$	$\log k$	$\log(k/m)$
G_1	0.20700	0.12460	0.08187	0.24440
G_2	0.07075	0.78890	0.07491	0.14640

G.4 Realizing value through frozen-source readouts

Changing the consuming route recovers additional value from frozen SpringWorld context, while the PokeWorld comparisons show that this recovery is not automatic.

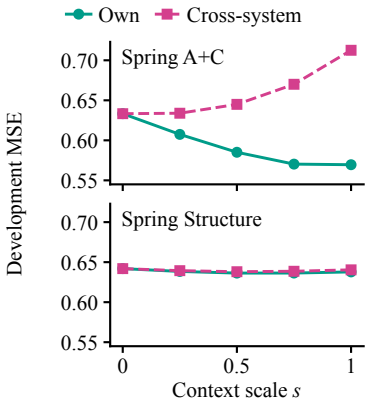
Reader recovery across source families. The M_1/M_2 formulas are defined in Appendix B.5. Nine Align + Cross pairs cover three sources by three readers; Structure has one source and three readers. Poke G_1/G_2 use source 2 with two/three readers. Checkpoints are selected on development data. Align + Cross favors M_2 in all nine standard and physically weighted pairs; Poke favors M_1 in all five. Table 55 reports descriptive fitted-grid means. The routes share their architecture, but their active capacity is not exactly matched (Appendix B.5).

Matched-architecture donor control. Matched context lowers error relative to a separately fitted mismatched reader with the same M_2 architecture. The comparison reuses the complete three-source by three-reader M_1/M_2 grid. Each mismatched reader receives 1,000 AdamW updates with batch size 256 and learning rate 10^{-3}; checkpoints are selected every 100 updates on the original development population. The source and its fitted recipient-only base R_0 remain frozen. A fixed different-system donor map applies during training, checkpoint selection and evaluation, with recipient inputs and targets unchanged. The original M_1/M_2 predictions are recomputed on the same records. The physical probes in Table 52 describe these same frozen source checkpoints.

Table 56 separates the full development mixture from its cold-start h16 endpoint. Matched M_2 has lower error than mismatched M_2 in all nine source–reader pairs at both endpoints, by 8.7% and 33.7%

Source	Seeds × readers	Standard	Weighted
Spring A+C	s0/1/2 × 3	+7.94%	+8.22%
Spring Structure	s0 × 3	-0.99%	-1.11%
Poke G1	s2 × 2	-0.35%	-0.18%
Poke G2	s2 × 3	-0.86%	-0.81%

(a) Mean within-fit error reduction



(b) Fixed M2 response

Figure 22: **Reader recovery depends on source family and donor context.** (a) Mean within-fit error reduction, $100[E(M_1) - E(M_2)]/E(M_1)$; positive favors M_2. Reader percentages are averaged within source, then across sources. (b) SpringWorld raw MSE under own- and cross-system donors through fixed M_2 readers; $s = 0$ gives the recipient-only base R_0.

Table 55: **Reader recovery differs across frozen source families.** Descriptive mean errors and gains $E(M_1) - E(M_2)$; positive favors M_2. Weighted gains use the separate physically weighted route. These fitted-run means do not estimate uncertainty.

Family	Sources × readers	M_1	M_2	Gain	Weighted gain
Align + Cross	3 × 3	0.6185	0.5695	0.04901	0.05059
Structure	1 × 3	0.6316	0.6378	−0.006213	−0.006971
Poke G_1	1 × 2	0.8099	0.8127	−0.002805	−0.001495
Poke G_2	1 × 3	0.7919	0.7987	−0.006841	−0.006443

All nine Align + Cross pairs favor M_2 in both route variants; all five Poke pairs favor M_1. Figure 22 shows within-fit percentages, while this table retains absolute differences. Full fitted rows, R_0 and Oracle results remain in the Supplementary Evidence Package.

in the respective means. Mismatched M_2 improves on M_1 in none of the nine pairs. Figure 23 retains both source-paired and hierarchical intervals: for mismatched $M_2 - M_1$, the source-paired intervals include zero at both endpoints, whereas the cold-start hierarchical interval is positive. These results identify a benefit of correctly associated context within this architecture and development protocol.

Table 56: **Matched context lowers error with identical M_2 architecture.** Task MSE, mean ± sample SD across three sources, each averaging three readers. Both populations are development data: 178 systems for the full mixture and 100 for cold-start h16.

Endpoint	M_1	M_2 matched	M_2 mismatched
Full development	0.6185 ± 0.009762	0.5695 ± 0.01796	0.6240 ± 0.007561
Cold start, h16	0.1799 ± 0.006272	0.1216 ± 0.01436	0.1835 ± 0.008084

Stronger formation does not ensure reader gains. Increasing relation reliability strengthens organization and drag accessibility, but it does not produce a positive mean M_2 gain at the 5,000-update h16 endpoint under this reader protocol (Table 57). Increasing the fitting budget reduces absolute error. Each $\alpha \in \{0, 0.5, 1\}$ uses three frozen 20,000-update PokeWorld sources and three readers per source. Unlike post-fitting scale s, α changes source training relations. Each paired contrast compares readers within a fixed source; query encoders vary across α.

The reader predicts train-standardized eight-state displacements on 400 systems from the existing validation bank. The recipient-only base R_0 and each route fit for 1,000 or 5,000 updates, with batch size 128 and learning rate 0.001, using the same 5,913 windows and development selection rule. Sources and cached observations are identical between budgets; geometry is unchanged and refitted probe differences are below 10^{-9}. Uncertainty uses the paired source/system bootstrap in

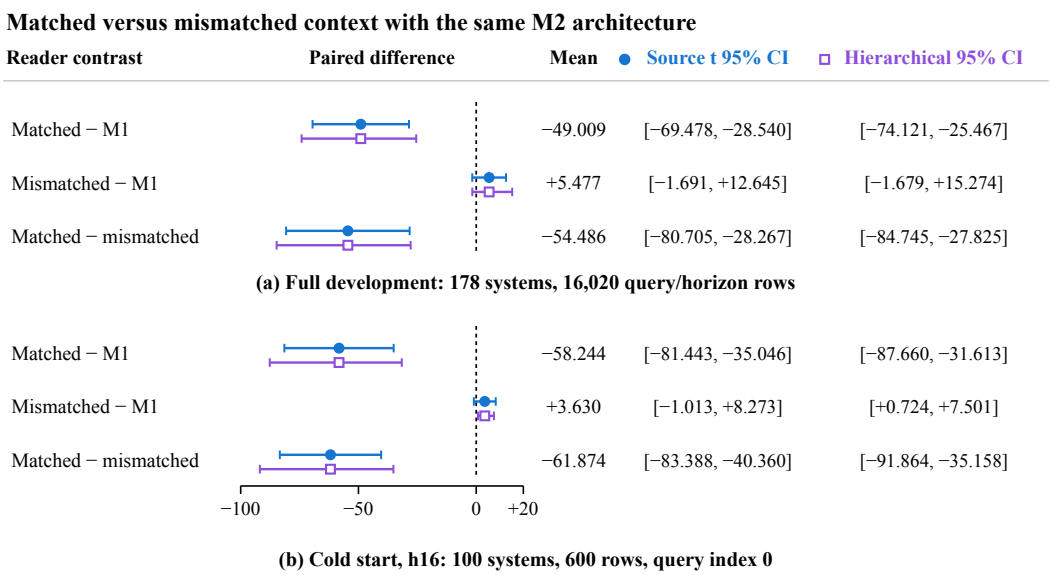


Figure 23: **Paired error differences for the matched-architecture control.** (a) Full development mixture; (b) cold-start h16. Negative values favor the first reader (10^{-3} MSE). Source-paired 95% t intervals use three source means; hierarchical 95% intervals resample sources, readers and systems. Both condition on development-selected fits; matched and mismatched M_2 readers are fitted separately.

Table 57. At 5,000 updates, all three wrong-minus-matched intervals include zero. Zero-context and wrong-donor interventions measure channel dependence and content specificity, respectively. Full source results at both budgets and all horizons are in the Supplementary Evidence Package.

Table 57: **Stronger formation does not ensure greater reader value.** h16 gain is $E(M_1) - E(M_2)$ in 10^{-3} normalized MSE; positive favors M_2. Budgets share frozen sources and development systems, averaging three readers per source. B/W uses training-standardized latent statistics.

A. Organization, accessibility and pooled reader value

α	B/W	$R^2_{\log\gamma}$	1k gain	5k gain	5k 95% interval
0	0.320	0.101	−2.512	−1.890	[−8.366, +3.116]
0.5	1.042	0.254	+0.061	−2.942	[−8.255, +1.948]
1	3.796	0.797	−4.469	−6.291	[−12.010, −0.328]

B. Source-specific reader value

α	Reader updates	Source 0	Source 1	Source 2
0	1000	−3.463	−3.017	−1.057
0.5	1000	+1.222	−0.015	−1.023
1	1000	−3.310	−4.186	−5.910
0	5000	−4.739	−1.617	+0.686
0.5	5000	−2.123	−3.436	−3.267
1	5000	−5.288	−11.270	−2.314

Intervals use 5,000 paired source/system bootstrap draws and are descriptive with three source seeds. Full horizon and per-route errors remain in the Supplementary Evidence Package.

Donor-specific responses through fixed readers. SpringWorld donor substitutions keep the fitted source and reader weights fixed while varying post-fitting context scale s. Own-system context lowers Align + Cross error as its scale increases, whereas whole-system cross donors raise it (Figure 22(b); Table 58). This difference supports content-specific use through the fixed reader. The single-source

Structure comparison has smaller, nonmonotone own-context effects. At $s = 0$, M_2 equals the recipient-only base R_0, not M_1; standard and physically weighted routes are evaluated separately.

Table 58: **Donor identity changes the response to context scale.** Development MSE as persistent inputs are multiplied by s, with donor substitutions through fixed readers. At $s = 0$, the route equals the recipient-only base R_0, not M_1.

A. Standard M_2; displayed in Figure 22

Source	Donor	$s = 0$	0.25	0.5	0.75	1
Align + Cross	Own	0.6333	0.6074	0.5851	0.5703	0.5695
	Cross-system	0.6333	0.6339	0.6450	0.6700	0.7125
Structure	Own	0.6419	0.6384	0.6363	0.6364	0.6378
	Cross-system	0.6419	0.6394	0.6381	0.6386	0.6404

B. Physically weighted $M_{2,\mathrm{phys}}$

Source	Donor	$s = 0$	0.25	0.5	0.75	1
Align + Cross	Own	0.6333	0.6090	0.5859	0.5684	0.5660
	Cross-system	0.6333	0.6337	0.6434	0.6671	0.7102
Structure	Own	0.6419	0.6387	0.6365	0.6361	0.6372
	Cross-system	0.6419	0.6394	0.6376	0.6376	0.6388

Route-selection and fixed-intervention evaluators remain separate.

Fresh-population consistency check. The preliminary Poke study uses 100 new systems with 16 windows each, six frozen source exports and twelve paired readers. Source slopes $+0.009202, -0.168619, +0.131394$ vary in sign. The study therefore ended at the prespecified consistency check, before the subsequent intervention. These fresh systems form a separate population from the earlier reader comparisons.

Object-only PokeWorld readouts. A separate single-source comparison restricts the target to four-dimensional object state. Its R8 $M_2 - M_1$ interval spans zero, while G_2 has higher M_2 mean error (Table 59). These independently selected readers do not establish additional recovery on this development population.

Table 59: **Object-only PokeWorld reader comparisons.** h16 object-state MSE on 400 development systems for one source–reader pair. M_1 and M_2 are selected independently; the paired interval is conditional on these fits and evaluation systems.

Source	M_2	M_1	Recipient-only
R8	0.9011	0.9026	0.9118
G_2	0.9132	0.9085	0.9080

R8 $M_2 - M_1$: -0.001497 $[-0.011855, 0.008400]$.

G.5 Closed-loop control with a development-selected recipe: Pendulum

The complete R7 recipe is selected on ID control development with relation weight 0.003 and variance target 0.1, then held fixed across ID and the four OOD groups. Three training seeds each contribute ten 200-step episodes per group. Table 60 reports the return and success rate from those same trajectories. Success means that all of the final 100 post-action states lie within 60° of upright; this criterion is unchanged.

Table 60: **Pendulum control with one ID-selected R7 recipe.** Mean return ± sample SD across three seeds, with success percentage in parentheses; higher is better. The same recipe is used for all OOD groups.

Group	CaDM	Random relation	SPRII (R7)
ID	-326.5 ± 46.5 (96.7%)	-321.4 ± 33.2 (100.0%)	-323.9 ± 26.2 (100.0%)
OOD_c0	-376.6 ± 63.4 (93.3%)	-346.8 ± 30.6 (100.0%)	-322.2 ± 29.4 (100.0%)
OOD_c1	-387.0 ± 11.9 (93.3%)	-400.1 ± 24.6 (93.3%)	-381.0 ± 13.6 (100.0%)
OOD_c2	-355.5 ± 33.9 (90.0%)	-479.6 ± 126.7 (76.7%)	-370.2 ± 51.8 (90.0%)
OOD_c3	-1100 ± 273.5 (20.0%)	-1185 ± 131.2 (6.7%)	-1060 ± 384.4 (30.0%)

R7 has higher mean return than CaDM in four of five groups, but lower return in OOD c2; the direction is not shared by all seeds in every group. The return and success endpoints remain separate. The earlier complete relation recipe is retained in the source package and is not combined columnwise with R7; R7 did not improve every group over that earlier recipe.

Same-checkpoint ID probes give mass R^2 of 0.027, 0.012 and 0.050 for CaDM, Random and SPRII, respectively, and length R^2 of 0.670, 0.683 and 0.669. The corresponding common-bank own-donor H10 MSEs are 0.0181, 0.0226 and 0.0190; wrong-minus-matched errors are 0.1132, 0.0956 and 0.1018. These logged-action, fixed-model interventions test context use separately from online control. The weak mass probes do not establish recovery of all physical parameters. Full seed values and the single-recipe selection record are preserved in the Supplementary Evidence Package.

H Generality Across Learners, Partners, and Fields

CoPhy varies learner and scene; Overcooked separates behavioral readout from cooperation outcomes; and FHN varies the number of independent histories for field prediction. The suite map in Appendix A.4 links each application to its implementation. The corresponding Formation, Use and Value analyses appear in Appendices E, F and G.

H.1 Learner and scene transfer: CoPhy

The CoPhy comparisons show both gains and adverse effects across learner families and scenes. We distinguish the source-0 development matrix, additional-source replication, and sensitivity to the fitted reader. The three-seed held-out Collision comparison is reported in Appendix C.2.

Breadth across learners and scenes. The source-0 matrix contains 45 evaluated learner–scene–recipe cells across JEPA, CPC, RSSM and supervised CoPhyNet in Balls, Collision and Blocktower. Figure 24 gives relative error reductions against each available control, and Table 61 gives absolute errors. CoPhyNet has Native and Random controls; its Structure arm was not evaluated.

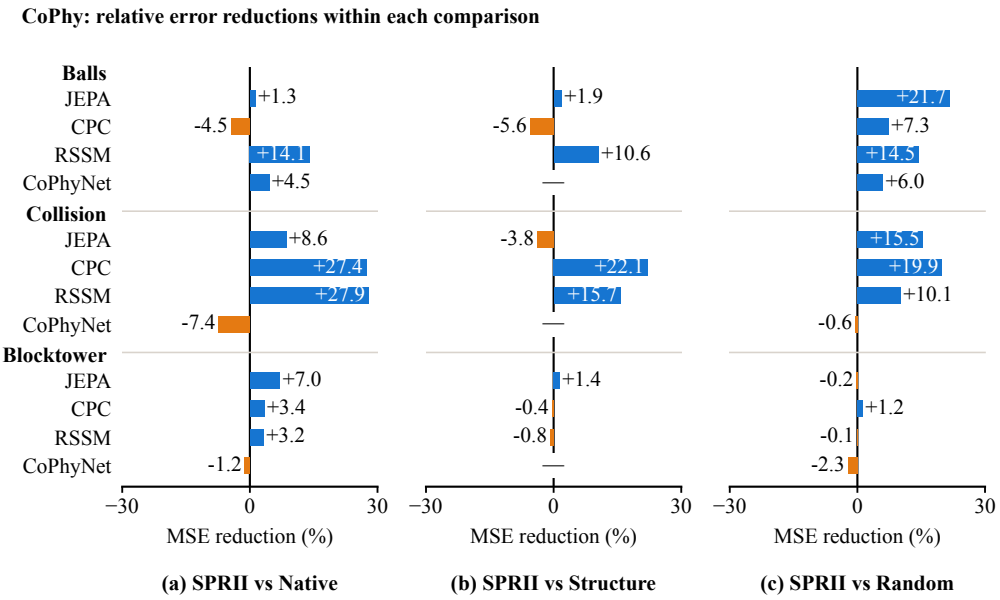


Figure 24: **CoPhy gains depend on learner, scene and comparator.** (a)–(c) Percentage MSE reduction against Native, Structure and Random: $100(\text{control} - \text{SPRII})/\text{control}$; positive favors SPRII. These are single-source development comparisons. CoPhyNet Structure was not evaluated in this matrix; held-out Collision results appear separately in Table 10.

Table 61: **CoPhy prediction error by learner and scene.** Source-0 development MSE under S3/query3 and source100/head100. Dashes mark unevaluated Structure arms; bold marks the lowest MSE per row.

Scene	Family	Native	Structure	SPRII	Random
Cross: self-supervised dynamics objectives					
Balls	JEPA	1.2563	1.2630	**1.2396**	1.5827
	CPC	1.4399	**1.4239**	1.5042	1.6228
	RSSM	1.3105	1.2582	**1.1254**	1.3155
Collision	JEPA	0.2512	**0.2212**	0.2297	0.2717
	CPC	0.2448	0.2283	**0.1778**	0.2220
	RSSM	0.2731	0.2338	**0.1970**	0.2192
Blocktower	JEPA	0.0931	0.0878	0.0865	**0.0863**
	CPC	0.1030	**0.0991**	0.0995	0.1007
	RSSM	0.0974	**0.0935**	0.0943	0.0942
Align + Cross: supervised CoPhyNet objective					
Balls	CoPhyNet	1.0534	–	**1.0056**	1.0698
Collision	CoPhyNet	**0.2033**	–	0.2184	0.2171
Blocktower	CoPhyNet	0.0871	–	0.0882	**0.0862**

Source information and consumed memory. Source U, independent support S and trained reader memory M are separate measurements (Appendices B.3 and B.5). The complete source-U grid covers all 45 source roles at source100, seed 0; it uses train-fitted ridge probes before the supervised reader projection. In Balls, SPRII restitution R^2_U is 0.7875/0.7672/0.8133/0.6170 for JEPA/CPC/RSSM/CoPhyNet, while mass/friction accessibility stays weak. Accessibility and task error dissociate here as well: CPC Balls has $R^2_U = 0.7672$ versus Native 0.5959 but MSE 1.5042 versus 1.4399. Collision mass access is lower for SPRII than Native in JEPA (0.3683 versus 0.3799) and CoPhyNet (0.1757 versus 0.2004). All factors and controls, including the Blocktower grid, are in the Supplementary Evidence Package, with U, S and M reported separately.

Replication across sources and in Balls. Additional sources preserve the comparison grid: JEPA Blocktower has higher error than Random at both sources, while JEPA Collision changes ordering against Structure. Appendix B.6 specifies source coverage and development selection.

Table 62: **CoPhy prediction and accessibility across source seeds.** Development results include reader-selection examples. S denotes independent support and M fitted reader memory. Probe targets are restitution (Balls), mass (Collision) and vertical gravity (Blocktower). Dashes mark unevaluated cells.

	Prediction MSE			Support R^2_S			Reader memory R^2_M		
Arm	**Src 0**	**Src 1**	**Src 2**	**Src 0**	**Src 1**	**Src 2**	**Src 0**	**Src 1**	**Src 2**
JEPA / Balls									
Native	1.256	1.319	—	0.7977	0.7981	—	0.7865	0.7778	—
Structure	1.263	1.294	—	0.7855	0.7866	—	0.7832	0.7788	—
Cross	1.240	1.305	—	0.9162	0.9185	—	0.9051	0.9116	—
Random	1.583	1.644	—	0.6777	0.6918	—	0.6884	0.6823	—
JEPA / Collision									
Native	0.2512	0.2444	—	0.4431	0.4366	—	0.5657	0.5619	—
Structure	0.2212	0.2238	—	0.5976	0.6246	—	0.6659	0.6706	—
Cross	0.2297	0.2154	—	0.6422	0.6718	—	0.6451	0.6713	—
Random	0.2717	0.2645	—	0.5005	0.5250	—	0.5286	0.5502	—
JEPA / Blocktower									
Native	0.0931	0.0895	—	0.1028	0.0902	—	0.0430	0.0467	—
Structure	0.0878	0.0876	—	0.1252	0.1301	—	0.0714	0.0879	—
Cross	0.0865	0.0876	—	0.1761	0.1564	—	0.1242	0.0860	—
Random	0.0863	0.0853	—	0.1673	0.1678	—	0.1256	0.1154	—
CPC / Collision									
Native	0.2448	0.2517	—	0.4372	0.4213	—	0.5604	0.5484	—
Structure	0.2283	0.2323	—	0.6126	0.5929	—	0.6476	0.6367	—
Cross	0.1778	0.1841	—	0.7495	0.7411	—	0.7541	0.7419	—
Random	0.2220	0.2212	—	0.6657	0.6588	—	0.6717	0.6711	—
RSSM / Collision									
Native	0.2731	0.2674	0.2729	0.334	0.240	0.344	0.497	0.494	0.477
Structure	0.2338	0.2332	0.2273	0.564	0.565	0.570	0.627	0.642	0.634
Cross	0.1970	0.1997	0.2112	0.738	0.723	0.733	0.736	0.719	0.724
Random	0.2192	0.2199	0.2284	0.633	0.643	0.634	0.678	0.669	0.664

RSSM Collision: three-source summary

Arm	**MSE: mean ± SD**	**Support R^2_S**	**Reader memory R^2_M**
Native	0.2711 ± 0.0032	0.306	0.490
Structure	0.2314 ± 0.0036	0.566	0.635
Cross	0.2026 ± 0.0075	0.731	0.727
Random	0.2225 ± 0.0051	0.636	0.671

The three-source Balls extension uses 2,000 development recipients with exact pairing. Its relation-trained arm improves Matched error against Native/Random/Structure by 8.95%/13.68%/9.13% for RSSM, and against Native/Random by 3.91%/3.19% for CoPhyNet. All five recipient-bootstrap family-adjusted intervals are positive. Table 63 reports Null, Matched and Wrong inputs and all five contrasts; these intervals condition on the fitted models and exclude selection uncertainty. Actual-memory use is analyzed in Appendix F.4; the source-U probes above measure a different channel.

Sensitivity to the fitted reader. With frozen sources, the CoPhyNet reader recipe changes comparison sign and magnitude across scenes. The source-50 factorial in Table 64 varies initialization, dropout and delivery; the source-100 endpoint below uses a separate budget.

At source-100, the every-step reader gives Native/SPRII/Random MSE 1.037115/0.956198/1.124851 on Balls and 0.208457/0.222786/0.218758 on Collision. These use the source-100 budget.

Operation and configuration comparisons. Table 65 reports the CoPhy configuration comparisons with their evaluation populations and fitted-model coverage. The single-source development

Table 63: **All five Balls contrasts favor relation training.** Three-source means on 2,000 development recipients. Recipient-bootstrap 95% intervals are Bonferroni-adjusted over five contrasts, conditional on fitted models and excluding selection uncertainty.

A. Absolute risks under recorded context conditions				
Family	**Arm**	**Matched**	**Null**	**Wrong**
RSSM	SPRII (C)	1.171	1.892	2.296
	Native	1.286	2.485	2.293
	Random	1.356	2.004	2.300
	Structure	1.288	1.937	2.335
CoPhyNet	SPRII (A+C)	1.044	1.791	2.228
	Native	1.087	1.773	2.244
	Random	1.079	1.759	2.216

B. Comparator minus SPRII, paired before source averaging			
Family	**Comparator**	**Difference**	**Adjusted 95% interval**
RSSM	Native	0.1151	$[0.0902, 0.1395]$
	Random	0.1856	$[0.1607, 0.2096]$
	Structure	0.1177	$[0.0927, 0.1433]$
CoPhyNet	Native	0.0424	$[0.0276, 0.0574]$
	Random	0.0344	$[0.0193, 0.0497]$

Table 64: **CoPhyNet gains depend on the reader recipe.** SPRII versus Native error reduction (%) for one frozen source on validation data (source-50/head-100).

Reader recipe	**Balls**	**Collision**	**Blocktower**
Main initialization, dropout 0.1, initial	-5.448	-4.089	-1.224
Alternative initialization, dropout 0	$+7.506$	-2.104	-1.477
Alternative initialization, dropout 0.1	$+6.074$	-4.426	$+2.090$
New initialization, dropout 0	$+0.617$	-7.591	$+0.621$
New initialization, dropout 0.1, every step	$+6.646$	-2.925	$+1.134$

comparisons complement the replicated results; their paired intervals condition on the fitted models and evaluation units.

Table 65: **CoPhy operation and configuration comparisons.** Errors are lower-is-better MSE. Development rows use one fitted source/reader seed; paired intervals condition on the reported models and evaluation units.

Configuration	Coverage	Numerical comparison	Finding / scope
JEPA Collision: J2	1,994 test episodes; 3 seeds per arm	J2 0.184 ± 0.005; Structure 0.215 ± 0.007	14.2% lower than Structure (all arms source-50); all four arms in Table 10.
CoPhyNet Collision: Cross	4,000 development recipients; seed 0	Cross 0.2032; SPRII S1 reference 0.2107	Near Native; paired interval against Native includes zero.
CPC Balls: Align	2,000 development recipients; seed 0	Align 1.393; SPRII C2 reference 1.415; Structure 1.388	Lower than the SPRII C2 reference, higher than Structure. Full contrasts are indexed in the package.
CoPhyNet Blocktower	8,088 development recipients; seed 0	SPRII 0.088180; Native 0.087114; later, separately evaluated Structure 0.087114; Random 0.086230. SPRII−Random: 0.001951 $[0.000668, 0.003184]$	Paired interval versus the later Structure control includes zero; Random has lower mean error.

Configuration and source identities are indexed in the Supplementary Evidence Package. Each row uses its stated population and fitted-model coverage.

H.2 Recurring-partner behavior and cooperation: Overcooked

Relative to H+VC, relation alignment lowers behavioral-probe error across the three history budgets. At full history, familiar-partner returns also improve in each of three training seeds. All comparisons use one layout and fixed 20,000-update endpoints; Appendix B.4 gives the shared evaluation protocol.

Table 66: **Full-history alignment improves familiar-partner returns in all three seeds.** Paired Align − H+VC differences, mean ± sample SD across three training seeds. Lower probe MSE and higher return are better. Familiar and held-out development partners are reported separately.

History	Familiar probe MSE	Familiar return	Held-out dev. return
25%	-0.00567 ± 0.00810	1.489 ± 2.335	-2.222 ± 15.21
50%	-0.01344 ± 0.00728	0.389 ± 1.110	18.44 ± 20.66
100%	$\mathbf{-0.02120} \pm 0.00610$	$\mathbf{1.889} \pm 2.175$	$\mathbf{-1.111} \pm 11.44$

For full history, the familiar return differences are $+4.400$, $+0.667$ and $+0.600$ across seeds, whereas held-out development differences are $+9.333$, $+0.667$ and -13.333. The latter population therefore shows mixed return effects. At 25% and 50% history, seeds also differ in familiar-return signs.

Table 67 compares behavioral readout, representation distances and cooperation returns at the same checkpoints. At full history, Align lowers probe MSE relative to Random in all three seeds (paired difference -0.02583 ± 0.00342), and the Align–H+VC probe difference is -0.02120 ± 0.00610. The Align mean, 0.05395, remains above the constant-mean reference, 0.04333; held-out development probe means also remain above their reference, 0.07062.

Table 67: **Overcooked behavioral readout and cooperation returns.** Mean ± sample SD across three training seeds. A denotes Align; VC denotes H+VC. The distance ratio is different-/same-partner distance on the familiar behavioral-signature bank, computed per seed before averaging. Random has no return evaluation.

A. Behavioral readout and representation

History	Recipe	Familiar MSE ↓	Held-out dev. MSE ↓	Distance ratio
25%	VC	0.06788 ± 0.00667	0.13880 ± 0.04230	1.343 ± 0.061
25%	A	0.06220 ± 0.00974	0.09733 ± 0.01384	1.917 ± 0.071
50%	VC	0.06934 ± 0.00811	0.14090 ± 0.02324	1.408 ± 0.019
50%	A	0.05590 ± 0.00736	0.08966 ± 0.01240	2.329 ± 0.326
100%	VC	0.07516 ± 0.00791	0.12400 ± 0.02857	1.458 ± 0.036
100%	A	0.05395 ± 0.01106	0.10630 ± 0.01046	2.269 ± 0.206
100%	Random	0.07978 ± 0.01423	0.14840 ± 0.03203	1.460 ± 0.066

B. Cooperation returns

History	Recipe	Familiar partners	Held-out dev. partners
25%	VC	15.78 ± 1.262	−1.333 ± 4.372
25%	A	17.27 ± 1.768	−3.556 ± 14.850
50%	VC	15.20 ± 0.200	−19.56 ± 2.341
50%	A	15.59 ± 0.916	−1.111 ± 18.410
100%	VC	14.04 ± 1.500	−19.11 ± 8.572
100%	A	15.93 ± 0.742	−20.22 ± 3.672

H.3 History-count scaling in reaction-diffusion fields: FHN

Additional independent histories reduce h50 field-prediction error in both methods. SPRII has larger relative reductions on all three splits, while its mean error remains higher than NOD at matched $K = 1$ and $K = 4$. From $K = 1$ to $K = 4$, SPRII error decreases by 16.9%, 18.0% and 10.9% on ID, OOD-Intra and OOD-Extra, compared with 8.4%, 9.2% and 7.1% for NOD. The released NOD-Hier architecture runs on the audited Python-generated bank (Appendix B.4). Initial condition 5 selects configurations; reporting uses 15, 24 and 45, with three fixed sources 42/43/44. Source 42 is the development seed; 43/44 repeat the selected recipe. NOD uses 45,000 updates and SPRII 50,000; these history-count comparisons therefore do not isolate alignment at equal training compute. The training schedules are specified in Appendix B.4.

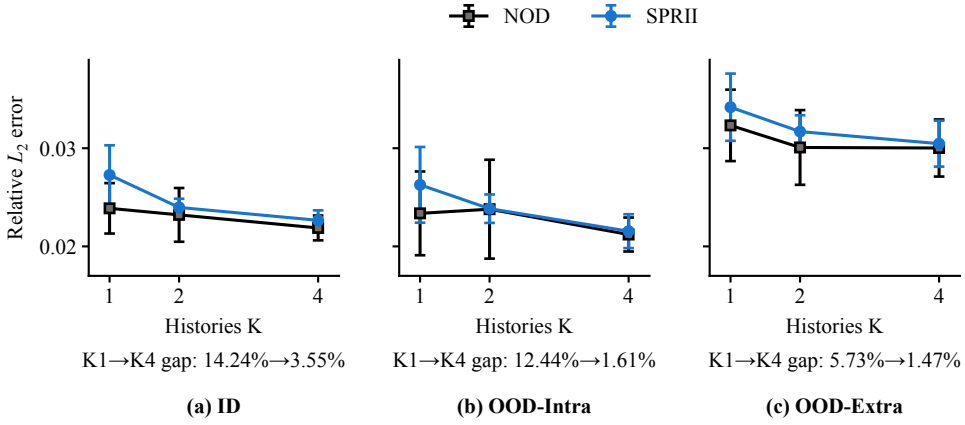


Figure 25: **Additional histories narrow the FHN endpoint gap on all three splits.** H50 relative-L_2 error: (a) ID, (b) OOD-Intra, (c) OOD-Extra. Points and whiskers show mean ± sample SD across three sources. Annotations give $100(\text{SPRII} - \text{NOD})/\text{NOD}$ at $K = 1, 4$.

Table 68: **Additional histories narrow the FHN prediction gap.** H50 relative-L_2 error averaged across three sources; lower is better. Both methods receive the same K independent histories, with different training budgets as described above.

	$K = 1 \to 4$ **reduction**		**NOD error**		**SPRII error**	
Split	**SPRII**	**NOD**	$K = 1$	$K = 4$	$K = 1$	$K = 4$
ID	16.9%	8.4%	0.02388	0.02188	0.02728	0.02266
OOD-Intra	18.0%	9.2%	0.02336	0.02121	0.02627	0.02155
OOD-Extra	10.9%	7.1%	0.03233	0.03002	0.03418	0.03046

At $K = 1$, SPRII's H50 mean error is 14.24%, 12.44% and 5.73% higher on ID, OOD-Intra and OOD-Extra. At the matched $K = 4$ observation budget, these gaps shrink to 3.55%, 1.61% and 1.47%. Factor accessibility varies by factor, split and source, as shown in the probe table below.

FHN factor accessibility. The bank contains nine training, four OOD-Intra and twelve OOD-Extra systems; $K = 1$ uses initial condition 36. All six models share 13,500 report-case keys.

Table 69: **FHN accessibility varies by factor, split and source.** Training-fitted probe R^2 with one history ($K = 1$).

A. k

Split	**Method**	**Src 42**	**Src 43**	**Src 44**
ID	NOD	0.9698	0.9301	0.9836
ID	SPRII	0.9444	0.3775	0.8952
Intra	NOD	0.9269	0.8098	0.9377
Intra	SPRII	0.8602	0.5513	0.6923
Extra	NOD	0.9351	0.9139	0.9284
Extra	SPRII	0.8587	0.0949	0.8730

B. β

Split	**Method**	**Src 42**	**Src 43**	**Src 44**
ID	NOD	0.9627	0.9496	0.9394
ID	SPRII	0.9766	0.9449	0.9097
Intra	NOD	0.9877	0.9829	0.9765
Intra	SPRII	0.9600	0.9269	0.8990
Extra	NOD	0.9521	0.9409	0.9299
Extra	SPRII	0.9848	0.9518	0.9154

The additional $K = 2, 4$ probes use their four-recipient population. The full source-by-history-by-horizon prediction grid, checkpoint identities and solver/equivalence records are in the Supplementary Evidence Package.

Additional-budget configuration comparison. A separate source-42 configuration is compared with extra-50k NOD on report initial conditions 15/24/45, excluding selection initial condition 5. At ID $K = 4, H = 50$, SPRII has lower mean error, but the paired-system interval includes zero: relative L_2 error is 0.02013 for SPRII and 0.02152 for extra-50k NOD. The interval conditions on the reported fitted models and evaluation systems; the other eight split/horizon cells have higher SPRII error.

I Theoretical Connections

We analyze four reference models connecting relations to persistent-context learning. Pairing determines which variations agreement penalizes; predictive relevance selects among stable directions; fixed-route interventions measure context dependence; and conditional risk characterizes the value available to a downstream reader. Each proposition states its required assumptions.

I.1 Setup and lifecycle

Setup. Let G be the grouping variable specified by a relation, H an interaction history, and $Z = \phi(H) \in \mathbb{R}^d$ a fixed representation with finite second moments. A group may denote a whole system, selected attributes, a partner, or a recorded task. Write $\mu_G = \mathbb{E}[Z \mid G]$, $\Sigma_{\text{between}}(G) = \text{Cov}(\mu_G)$, and $\Sigma_{\text{within}}(G) = \mathbb{E}[\text{Cov}(Z \mid G)]$. The sampling law is $\Pi_G(A, B) = P(B)Q_G(A \mid B)$, as in Eq. (5). Conditional independence is imposed explicitly for the iid results below; the general pair law permits dependent interaction histories.

Remark: available information, reliance and realized value. The lifecycle distinguishes information present in a context, reliance by a specified predictor, and task value realized by that predictor. These are operationally assessed with geometry and probes, fixed-predictor substitutions, and task readers, respectively; failure of a particular probe is not evidence that the information is absent. For a fixed joint law of recipient inputs X, donor code D and target Y, conditional independence of Y and D given X implies zero *Bayes-optimal incremental value* $\Delta_\ell^\star$ in Proposition 4. This statement concerns additional target information beyond X, not the difference between arbitrary finite fitted readers: their excess-risk terms can differ even when $\Delta_\ell^\star = 0$. Conversely, accessible information need not be relevant beyond X, and positive available value need not be realized by a fitted reader. Donor sensitivity establishes reliance under the specified substitution, while the residual cross term in Appendix I.3 determines whether that reliance reduces risk. A newly fitted reader's benefit concerns its own use of context; it does not establish use by the original predictor.

I.2 Formation: pairing, reliability and predictive selection

Pairing structure.

Proposition 1 (structure exposed by pairing). For a pair law with finite second moments, means μ_A, μ_B, marginal covariances Σ_A, Σ_B, and cross covariance Σ_{AB},

$$\mathbb{E}_{\Pi_G}\|Z^A - Z^B\|_2^2 = \operatorname{tr}\Sigma_A + \operatorname{tr}\Sigma_B - 2\operatorname{tr}\Sigma_{AB} + \|\mu_A - \mu_B\|_2^2. \tag{14}$$

If a matched pair is instead specified by drawing G from its population law and then drawing Z^A, Z^B conditionally iid from $P(Z \mid G)$, then

$$\operatorname{Cov}(Z^A, Z^B) = \Sigma_{\text{between}}(G), \qquad \mathbb{E}\|Z^A - Z^B\|_2^2 = 2\operatorname{tr}\Sigma_{\text{within}}(G). \tag{15}$$

Proof. Expand the squared difference and use $\mathbb{E}\|Z^A\|^2 = \operatorname{tr}\Sigma_A + \|\mu_A\|^2$ and $\mathbb{E}[(Z^A)^\top Z^B] = \operatorname{tr}\Sigma_{AB} + \mu_A^\top\mu_B$. This proves Eq. (14). For the ideal matched pair, write $Z^i = \mu_G + \epsilon_i$. Conditional centering gives $\mathbb{E}[\epsilon_i \mid G] = 0$, and conditional independence gives $\mathbb{E}[\epsilon_A\epsilon_B^\top \mid G] = 0$. Thus the cross covariance is $\operatorname{Cov}(\mu_G)$, while the covariance of $\epsilon_A - \epsilon_B$ given G is $2\operatorname{Cov}(Z \mid G)$. Taking its trace and averaging proves Eq. (15). The same centering argument also yields the total-covariance identity $\operatorname{Cov}(Z) = \Sigma_{\text{between}}(G) + \Sigma_{\text{within}}(G)$. □

What changes under an actual sampler? Conditional centering gives, without the iid premise,

$$\Sigma_{AB} = \operatorname{Cov}(\mathbb{E}[Z^A \mid G], \mathbb{E}[Z^B \mid G]) + \mathbb{E}[\operatorname{Cov}(Z^A, Z^B \mid G)].$$

The two mixed mean–residual terms vanish by iterated expectation. Overlap, shared episode dynamics, balancing, and forced mismatches can leave the second term nonzero or alter the conditional means. The actual pairing law therefore determines which dependence alignment preserves. Under ideal pairing, alignment penalizes within-group differences and leaves group-mean variation as a candidate for retention. A constant code also has zero alignment loss; predictive and non-collapse terms provide the additional constraints needed to learn informative codes.

Reliability and independent repetitions. Reliability changes whether shared variation is exempt from the disagreement penalty. Let Q_c select correct pairs and Q_r random pairs with the same donor and recipient marginals. For $\alpha \in [0,1]$, the mixture $Q_\alpha = \alpha Q_c + (1-\alpha)Q_r$ implies $\mathbb{E}_{Q_\alpha}\ell = \alpha\mathbb{E}_{Q_c}\ell + (1-\alpha)\mathbb{E}_{Q_r}\ell$ for any integrable pair loss at fixed parameters.

Proposition 2 (alignment pressure under corrupted relations). Let $H = L\theta + N$, where θ has covariance Σ_θ and independent zero-mean noise N has covariance Σ_N. Correct pairs share θ and independently resample N; random pairs independently resample both. For a fixed encoder $Z = WH$,

$$\mathbb{E}_{Q_\alpha}\|W(H^A - H^B)\|_2^2 = 2\operatorname{tr}(W\Sigma_N W^\top) + 2(1-\alpha)\operatorname{tr}(WL\Sigma_\theta L^\top W^\top). \tag{16}$$

Proof. Under Q_c, $H^A - H^B = N^A - N^B$ has zero mean and covariance $2\Sigma_N$. Under Q_r, independence gives covariance $2L\Sigma_\theta L^\top + 2\Sigma_N$ for the difference. In each case, the expected

squared norm after applying W is the trace of the transformed covariance. Mixing these two expressions proves the result. □

Correct pairing leaves persistent variation outside this penalty; random pairing contracts it. The coefficient $1 - \alpha$ describes the objective in this reference model; Figure 3(a) reports the learned response. If only alignment assignments change, per-view variance/covariance regularizers remain common. If Cross consumes the corrupted assignment, its task loss changes too. The calculation therefore isolates alignment pressure; the monotone geometry in Figure 3(a) is an empirical response of separately trained nonlinear models.

Corollary (factor-specific alignment pressure). In Proposition 2, let $\theta = (m, \gamma, k)$ have independent components, $L = [L_m, L_\gamma, L_k]$, and $\mathrm{Var}(\gamma) = \sigma_\gamma^2$. Correct pairs share all three factors. In the corrupted component, retain m, k and independently redraw only γ from its marginal; independently redraw the observation noise in both components. For a fixed W,

$$\mathbb{E}\|W(H^A - H^B)\|^2 = 2\,\mathrm{tr}(W\Sigma_N W^\top) + 2(1-\alpha)\sigma_\gamma^2\,\mathrm{tr}(WL_\gamma L_\gamma^\top W^\top). \tag{17}$$

Proof. The correct-pair difference is $N^A - N^B$. In the corrupted component it is $L_\gamma(\gamma^A - \gamma^B) + N^A - N^B$; independence and centering remove cross terms and give second moment $2\sigma_\gamma^2 L_\gamma L_\gamma^\top + 2\Sigma_N$. Taking the trace after applying W and mixing the two components yields Eq. (17). □

The additional penalty acts along L_γ; the shared m, k variations cancel from the pair difference. This fixed-encoder statement isolates the pressure exerted by alignment. It does not imply that retraining a nonlinear model leaves all other factors' geometry or probe accuracy unchanged: predictive losses, capacity and variance constraints also select the learned representation. Factor-selective degradation is therefore an empirical hypothesis for Appendix E.4. For a finite bank whose sampler *forces* unequal drag values, independent redrawing is not the sampling law. Replace $2\sigma_\gamma^2$ by the actual $\mathbb{E}_Q[(\gamma^A - \gamma^B)^2]$; for uniform distinct draws from n bank entries, this is $2n\sigma^2_{\gamma,\mathrm{bank}}/(n-1)$ when bank variance uses denominator n. More general balancing and unequal supports use the actual pair second moment in Proposition 1. The qualitative cancellation of exactly shared factors remains, while the coefficient must match the sampler.

Why separate realizations can add evidence. For conditionally iid codes $Z_1, \ldots, Z_R \mid G$ with mean μ_G and covariance $\Sigma(G)$, the average $\bar{Z}_R = R^{-1}\sum_r Z_r$ has conditional covariance $\Sigma(G)/R$: independence removes the off-diagonal covariance terms in the sum. Consequently, $\mathbb{E}\|\bar{Z}_R - \mu_G\|^2 = \mathrm{tr}\,\Sigma_{\mathrm{within}}(G)/R$. For dependent repetitions, the remaining cross-covariance terms determine the reduction. This estimation reference describes the information available from repeated histories; SPRII learns its encoder from those histories. The R2/R4/R8 bank comparison in Table 31 tests the learning effect of providing more interaction realizations.

Semantic refinement and predictive selection. A relation specifies which variation may persist across a pair; the training objective determines which eligible factor the representation emphasizes. We first analyze a fixed code, then give a learning example in which the dominant factor changes.

Semantic refinement for a fixed representation. Let G_2 refine G_1, so that G_1 is a measurable function of G_2. For the same fixed Z under the same population law, total covariance gives

$$\Sigma^Z_{\mathrm{between}}(G_2) - \Sigma^Z_{\mathrm{between}}(G_1) = \mathbb{E}[\mathrm{Cov}(\mathbb{E}[Z \mid G_2] \mid G_1)] \succeq 0.$$

To see this, write $M_i = \mathbb{E}[Z \mid G_i]$. The tower property gives $\mathbb{E}[M_2 \mid G_1] = M_1$; total covariance applied to M_2 yields the identity. Positive semidefiniteness follows because the last term's quadratic form is $\mathbb{E}[\mathrm{Var}(v^\top M_2 \mid G_1)] \geq 0$ for every v. Since the total covariance of Z is unchanged, within-group covariance decreases by the same matrix.

Thus $G_1 = m$, $G_2 = (m, \gamma)$, and $G_3 = (m, \gamma, k)$ successively make drag and stiffness eligible as group-stable variation. This identity compares *one representation under different groupings*. Experimental donor pools may include forced mismatches and have non-nested supports; their effects enter through the actual pair law.

Predictive selection under relation constraints. The preceding identities keep the representation fixed while changing its grouping. We now allow the representation to change and examine the joint influence of predictive reward and relation disagreement in a joint spectral selection problem.

A constrained predictive reference. Let U collect the inputs supplied outside the history code, such as current state, actions, and horizon. To model history information beyond U, assume finite second moments and consider the residuals $\widetilde{H} = H - \mathbb{E}[H \mid U]$ and $\widetilde{Y} = Y - \mathbb{E}[Y \mid U]$, with a centered linear model $\widetilde{Y} = K\widetilde{H} + \eta$ and $\mathbb{E}[\eta\widetilde{H}^\top] = 0$. Assume positive-definite history covariance on the modeled subspace and whiten that subspace. Below, H, Y denote these transformed variables, with $\mathrm{Cov}(H) = I_p$ and $Y = KH + \eta$ (absorbing the whitening into K). Residualization and whitening define this analytical reference model.

Proposition 3 (joint predictive and relation selection). For $Z = WH$, $W \in \mathbb{R}^{d\times p}$ with $1 \le d < p$, impose $WW^\top = I_d$ and let $\beta \ge 0$. Consider

$$\mathcal{J}_G(W) = \min_C \mathbb{E}\|Y - CWH\|_2^2 + \beta\mathbb{E}_{\Pi_G}\|W(H^A - H^B)\|_2^2. \tag{18}$$

Write $D_G = \mathbb{E}_{\Pi_G}[(H^A - H^B)(H^A - H^B)^\top]$ and $M_G = K^\top K - \beta D_G$. The optimal decoder is $C^\star = KW^\top$, and the optimal row spaces of W are top-d eigenspaces of M_G:

$$W_G^\star \in \arg\max_{WW^\top = I_d} \mathrm{tr}(WM_GW^\top). \tag{19}$$

The selected subspace is unique if $\lambda_d(M_G) > \lambda_{d+1}(M_G)$; its orthonormal basis need not be.

Proof. Since $\mathrm{Cov}(WH) = I_d$ and $\mathrm{Cov}(Y, WH) = KW^\top$, completing the square gives

$$\begin{aligned}\mathbb{E}\|Y - CWH\|_2^2 &= \mathrm{tr}\,\mathrm{Cov}(Y) - 2\,\mathrm{tr}(CWK^\top) + \mathrm{tr}(CC^\top)\\ &= \mathrm{tr}\,\mathrm{Cov}(Y) - \|KW^\top\|_F^2 + \|C - KW^\top\|_F^2.\end{aligned}$$

Minimization sets $C = KW^\top$ and leaves predictive reward $\mathrm{tr}(WK^\top KW^\top)$. The alignment term equals $\mathrm{tr}(WD_GW^\top)$, including when the pair difference has a nonzero mean. Removing the constant target variance therefore yields Eq. (19).

To characterize the maximum, diagonalize the symmetric matrix $M_G = V\,\mathrm{diag}(\lambda_1, \ldots, \lambda_p)V^\top$ with decreasing eigenvalues. Since $W^\top W$ is a rank-d orthogonal projector, $a_i = \|Wv_i\|_2^2$ satisfies $0 \le a_i \le 1$ and $\sum_i a_i = d$. Hence

$$\mathrm{tr}(WM_GW^\top) = \sum_{i=1}^{p} \lambda_i a_i \le \sum_{i=1}^{d} \lambda_i.$$

Choosing the first d eigenvectors as rows of W attains the bound. A strict eigengap requires projection onto precisely that subspace; ties permit alternative choices within the tied eigenspace. □

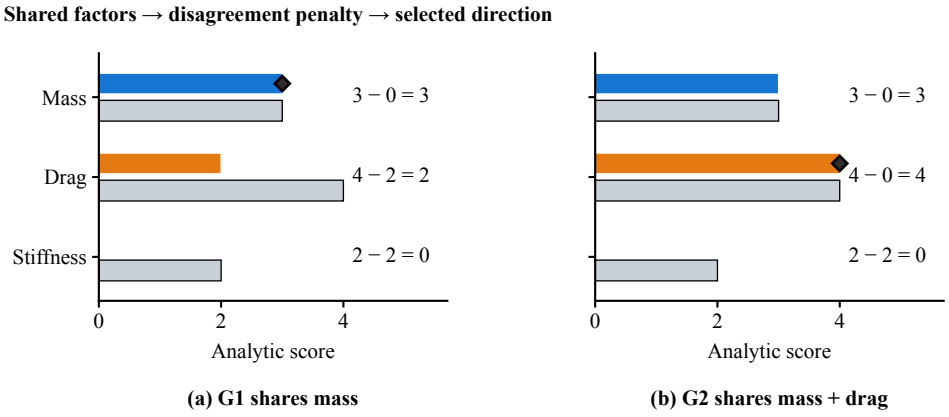


Figure 26: **A broader sharing relation can select a different predictive direction.** In the Proposition 3 example, broadening (a) G_1 to (b) G_2 changes the selected direction (diamond) from mass to drag. Gray bars show predictive scores; colored bars subtract relation disagreement.

A realizable three-factor crossing. Let $H = (M, \Gamma, K_0)$ have independent standard-normal coordinates, use $d = 1$, $\beta = 1$, and set $K^\top K = \mathrm{diag}(3, 4, 2)$. Under $G_1 = M$, pairs share mass and independently resample the other two coordinates. Under $G_2 = (M, \Gamma)$, they share mass and drag. Both laws have the same standard-normal marginals. A shared coordinate cancels in the pair difference; two independent standard normals have difference variance 2. Therefore

$$\begin{aligned} D_{G_1} &= \mathrm{diag}(0, 2, 2), \quad M_{G_1} = \mathrm{diag}(3, 2, 0), \\ D_{G_2} &= \mathrm{diag}(0, 0, 2), \quad M_{G_2} = \mathrm{diag}(3, 4, 0). \end{aligned} \tag{20}$$

Proposition 3 selects $W^\star_{G_1} = e_m^\top$ but $W^\star_{G_2} = e_\gamma^\top$. Mass remains shared, yet removing drag's penalty makes drag the leading direction. Broader semantic eligibility is therefore compatible with a different, non-nested learned subspace. This reproduces the qualitative factor crossover in Figure 3(b) and Appendix E.1: the preferred direction changes while the original factor remains shared.

The argument also presupposes available evidence: for a fixed encoder, a factor independent of its input remains independent of its code. The actual-input estimators in Table 28 therefore check for factor evidence in histories before asking which factors the representation emphasizes.

I.3 Use: fixed-predictor reliance

A fixed-weight intervention holds the predictor and recipient inputs fixed and changes supplied persistent context. It tests functional use without refitting readers for each donor condition.

Risk change under a donor substitution. Fix a predictor f and a joint evaluation law for (X, Y, D_m, D_s), where D_m is matched and D_s is the specified replacement. Assume finite second moments for the target and both predictions. Write $\hat{Y}_m = f(X, D_m)$, $\hat{Y}_s = f(X, D_s)$, and let $\Delta_{\mathrm{spec}} = \mathbb{E}\|Y - \hat{Y}_s\|^2 - \mathbb{E}\|Y - \hat{Y}_m\|^2$. Expanding $Y - \hat{Y}_s = (Y - \hat{Y}_m) + (\hat{Y}_m - \hat{Y}_s)$ yields

$$\Delta_{\mathrm{spec}} = \mathbb{E}\|\hat{Y}_s - \hat{Y}_m\|^2 + 2\mathbb{E}\langle Y - \hat{Y}_m, \hat{Y}_m - \hat{Y}_s\rangle. \tag{21}$$

The risk change includes both prediction sensitivity and its interaction with the original residual. If the matched prediction is $\mathbb{E}[Y \mid X, D_m]$ and the replacement adds no conditional target information, so $\mathbb{E}[Y \mid X, D_m, D_s] = \mathbb{E}[Y \mid X, D_m]$, conditioning makes the cross term vanish. Then specificity equals the mean-squared prediction change. The general decomposition applies to any trained predictor with finite risk.

Positive specificity establishes lower risk with the matched donor under the specified replacement law. Near-zero specificity can arise from cancellation of the two terms despite different predictions. The donor and recipient protocols in RH20T, SpringWorld and Swimmer specify that law. Zero-context replacement measures channel dependence, while wrong-donor replacement measures content specificity. Absolute-risk comparisons and fresh-reader evaluations quantify task benefit.

I.4 Value: conditional risk and reader realization

The same relation-conditioned interface supports numerical prediction and categorical action learning, but their native losses measure different quantities. We apply standard conditional-risk and information-theoretic identities (Cover & Thomas, 2006; Gneiting & Raftery, 2007) to separate available task value from its realization by a trained reader.

Common setup. Fix the encoders, target construction, and pairing law $\Pi_G(A, B) = P(B)Q_G(A \mid B)$. They induce a joint law of the permitted recipient inputs $X = X_B$, donor code $D = D_A$, and target $Y = Y_B$. Write $R_\ell(f) = \mathbb{E}\ell(f, Y)$. Risk is evaluated under this fixed joint law, allowing statistical dependence between donor and recipient. For Cross it is the recipient-conditioned risk in Eq. (3); the same analysis applies to a native history-conditioned reader in an Align-only instance.

Proposition 4 (task value and reader error). Let $\Delta^\star_\ell$ be the optimal recipient-only risk minus the optimal risk with (X, D), allowing unrestricted measurable readers.

(a) Numerical targets. For square-integrable Y and squared loss, the optimal readers are $m_0 = \mathbb{E}[Y \mid X]$ and $m_D = \mathbb{E}[Y \mid X, D]$, and

$$\Delta^\star_{\mathrm{sq}} = \mathbb{E}\|Y - m_0\|^2 - \mathbb{E}\|Y - m_D\|^2 = \mathbb{E}\|m_D - m_0\|^2 \geq 0. \tag{22}$$

(b) Categorical targets. For Y in a finite alphabet and log loss $\ell(p, Y) = -\log p(Y)$, write $p_0 = P(Y = \cdot \mid X)$ and $p_D = P(Y = \cdot \mid X, D)$. These distributions attain the respective optimal risks $H(Y \mid X)$ and $H(Y \mid X, D)$, so

$$\Delta^\star_{\log} = H(Y \mid X) - H(Y \mid X, D) = I(Y; D \mid X) = \mathbb{E}\,\mathrm{KL}(p_D \| p_0) \geq 0. \tag{23}$$

Logarithms are natural. We assume conditional distributions exist; X and D may be continuous.

For either loss, two learned readers f_0 and f_D with finite risk obey

$$R_\ell(f_0) - R_\ell(f_D) = \Delta^\star_\ell + \mathcal{E}^\ell_0 - \mathcal{E}^\ell_D, \tag{24}$$

where the excess risks are

$$\mathcal{E}^{\mathrm{sq}}_0 = \mathbb{E}\|f_0 - m_0\|^2, \quad \mathcal{E}^{\mathrm{sq}}_D = \mathbb{E}\|f_D - m_D\|^2,$$
$$\mathcal{E}^{\log}_0 = \mathbb{E}\,\mathrm{KL}(p_0 \| \widehat{p}_0), \quad \mathcal{E}^{\log}_D = \mathbb{E}\,\mathrm{KL}(p_D \| \widehat{p}_D).$$

Here the learned log-loss readers are probability vectors $\widehat{p}_0, \widehat{p}_D$. The KL comparisons use the same conditioning inputs; assigning zero probability to a positive-probability event is excluded by the finite-risk assumption.

Proof. For squared loss, $r = Y - m_D$ has zero conditional mean given (X, D) and is orthogonal to $v = m_D - m_0$. Expanding $Y - m_0 = r + v$ proves Eq. (22). More generally, for a reader f with its corresponding conditional mean m,

$$\mathbb{E}\|Y - f\|^2 = \mathbb{E}\|Y - m\|^2 + \mathbb{E}\|m - f\|^2,$$

by the same orthogonality, establishing optimality and the squared-loss excess risks.

For log loss, adding and subtracting the true log probabilities gives

$$\mathbb{E}[-\log \widehat{p}_D(Y) \mid X, D] = -\sum_y p_D(y) \log \widehat{p}_D(y) = H(p_D) + \mathrm{KL}(p_D \| \widehat{p}_D).$$

The nonnegativity of KL divergence shows that p_D is optimal. Average this equality over (X, D) and repeat it conditioned on X to obtain both optimal risks and excess-risk terms. Finally, iterated expectation gives

$$\mathbb{E}\,\mathrm{KL}(p_D \| p_0) = \mathbb{E} \log \frac{p_D(Y)}{p_0(Y)} = H(Y \mid X) - H(Y \mid X, D) = I(Y; D \mid X).$$

Subtracting the two reader-risk decompositions for each loss proves Eq. (24). □

What this separates. A factor can be perfectly accessible in D while irrelevant to Y: if F_1, F_2 are independent, $D = F_1$, and $Y = F_2$ with no recipient inputs, the code's task value is zero. Redundancy with X can also remove its incremental value. For squared loss, zero value means that the conditional *mean* is unchanged; for categorical log loss it means that the conditional *distribution* is unchanged. The criterion thus depends on the chosen loss. Equation (24) further separates attainable value from how well the particular readers realize it. The PokeWorld frozen-source study estimates realized risk for its fitted readers. Together with the accessibility probes, it separates readable information from the benefit those readers obtain.

An explicit cold-start example. Let $\theta \sim \mathcal{N}(0, \sigma^2_\theta)$, $X = \theta + \epsilon_x$, and $D = \theta + \epsilon_d$, with centered Gaussian noises independent of each other and of θ, and positive prior and noise variances. For $Y = a_h\theta + \epsilon_y$, assume centered target noise of finite variance independent of $\theta, \epsilon_x, \epsilon_d$. Completing the square in the Gaussian conditional density gives recipient posterior variance $v_x = (\sigma_\theta^{-2} + \sigma_x^{-2})^{-1}$ and, after observing D, variance $(v_x^{-1} + \sigma_d^{-2})^{-1}$. Target noise cancels from the risk difference, giving

$$\Delta^\star_{\mathrm{sq}} = a_h^2 \left[v_x - (v_x^{-1} + \sigma_d^{-2})^{-1} \right] = a_h^2 \frac{v_x^2}{v_x + \sigma_d^2}. \tag{25}$$

Within this model, larger remaining recipient uncertainty v_x makes the same history more valuable. Averaging R independent history measurements replaces σ_d^2 by σ_d^2/R. Target sensitivity enters through a_h; a longer horizon helps only insofar as it changes the target's dependence on persistent information or the available evidence. The model explains the role of recipient uncertainty in SpringWorld cold start. A controlled change in recipient evidence holds population, target and fitted route fixed; horizon effects additionally depend on a_h.

Connection to training and generalization beyond JEPA. For fixed encoders, targets, and pairing law, the isolated Cross objective asks the reader for the recipient target's conditional mean under squared loss, or its conditional distribution under categorical log loss. Sharing an object, partner, or task constructs potentially informative pairs; the donor still needs to add target-relevant information beyond X. A numerical target can be a future state or a latent embedding held fixed for this reference; a categorical target can be an observed action. The complete SPRII objective fits these components jointly through a shared predictor.

A training law and an evaluation law define separate risks; each identity holds within its specified law and fixed representation. Action log loss also differs from rollout return, which depends on the policy-induced state distribution and environment response. Thus the generalization is a common relation-conditioned learning interface, with native targets and losses specifying what the reader learns to use. Native-task evaluation tests its realized benefit.

Connection to the downstream intervention. The M_1/M_2 study changes the consuming interface while keeping the source and prescribed query/base path fixed. The gain $\Delta_{\text{realize}} = \text{Err}(M_1) - \text{Err}(M_2)$ measures a difference in realized risk, with positive values favoring M_2 (Section 3.4). Equation (24) attributes this difference to conditional value and reader error jointly. A positive adapter gain demonstrates additional realization under the compared routes; the unrestricted attainable value remains a separate quantity.

J Closest Methodological Connections

The closest precedents already use related experience, separate context from state, or test dependence on supplied context. Table 70 locates the additional questions studied here.

Table 70: **Closest methodological connections.**

Work	Relevant construction	Connection developed by SPRII
FCRL	Same-function observation sets; contrastive encoding; new decoders on a fixed encoder.	Relation training within predictive/action learners; selected shared factors; separate measurement of code access and predictor reliance.
NOD	A separate same-system trajectory supplies a low-dimensional code for state evolution.	Embedding, outcome, and action targets; Align/Cross components; controlled relation semantics and information conditions.
RIA	Predictive interventions estimate relations used to organize dynamics context.	Observed relations provide a controlled variable whose reliability and semantics can be changed independently.
GG-ODE	Initial-state/environment separation; same-environment contrastive regularization.	An interface instantiated across base objectives, with partial-factor sharing and task/use diagnostics.
Policy embeddings	Cross-episode imitation and identity objectives; downstream policies and wrong-embedding tests.	Relationship to physical and JEPA persistence; history-conditioned action applications and controlled information-use studies.

Relations and predictive context. FCRL learns from same-function observation sets and transfers fixed encoders (Gondal et al., 2021). NOD conditions an operator on a separate same-system trajectory, directly preceding the Cross-type construction (Chen et al., 2026). RIA uses an interventional prediction module to estimate whether two inferred contexts come from the same environment, then regularizes their similarity with a relational head. GG-ODE separates initial state from environmental context (Guo et al., 2022; Huang et al., 2023). Agent embeddings also use cross-episode imitation and incorrect-context tests (Grover et al., 2018); our Overcooked instance adds alignment to a history-conditioned action learner (Laskin et al., 2023; Jing et al., 2026). Here, controlled reliability and partial-factor relations make the selected shared content an experimental variable, while Align/Cross controls separate operations within each native predictive objective.

Context inference and test-time adaptation. CaDM learns dynamics context through forward/backward prediction (Lee et al., 2020); DALI learns latent dynamics context for adaptation (Röder et al., 2025). Our D-Clean experiment adapts the context-learning component to a common frozen-code reader and reports source-training budgets explicitly; it does not rank the original control agents. CoDA and GEPS adapt physical models with context codes (Kirchmeyer et al., 2022; Kassaï Koupaï et al., 2024). Their completed Burgers references use 50 support-code optimization steps per query, whereas NOD and the SPRII endpoint use amortized code inference. The frozen-decoder CoDA extension in Appendix C.5 asks the separate question of whether relation supervision improves an amortized history encoder on this host. Its four control arms distinguish encoder learning, relation assignments and additional reconstruction computation.

Prediction, accessibility and reliance. Latent world models and joint-embedding predictors provide the predictive setting (Hafner et al., 2019; 2020; Assran et al., 2023; Bardes et al., 2024). Physical-information studies distinguish input evidence from learned access (Tan et al., 2026), while paired-view learning constrains shared information (Bardes et al., 2022; Tian et al., 2020). Our grouping supervision supplies relation assignments between histories. A frozen-code probe tests accessibility; a fitted reader tests realized task value; substitution through fixed weights tests model reliance under the stated donor law (Fisher et al., 2019; Molnar et al., 2023; Watson & Wright, 2021). These distinctions motivate the separate Formation, Use and Value analyses.