BEYOND MULTIMODAL ALIGNMENT: SHARED PHYSICAL REPRESENTATIONS ACROSS SENSORS AND ACTION ORDERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal world models are often evaluated by whether different sensors produce similar representations. However, similar representations do not necessarily imply that the models make the same physical predictions, or that those representations can be reused when actions are combined in a new order. We study both questions through the physical responses predicted by a model. We first use the Cluster Haptic dataset to ask whether audio and acceleration can independently recover the behavior of the same surface from different observations. Predictions from the two sensors are substantially closer for the same surface than for different surfaces, with a $4.5\times$ gap on average, while both also outperform an average-surface prediction. We then show that this agreement alone does not determine how familiar actions should compose. In a controlled elastoplastic system, shared step dynamics fit observed programs less accurately than a whole-program predictor but generalize better to unseen action orders, with the ranking reversing on both held-out transitions across three independent initializations. Fusing free-decay and hysteresis observations further improves prediction, with diagonal Gaussian beliefs yielding the lowest errors. Together, these results distinguish cross-sensor consistency, multimodal fusion, and generalization to new action orders as separate questions in evaluating multimodal physical representations.

1 INTRODUCTION

Multimodal world models use shared representations to connect sensory observations with predictions of physical behavior (Shang et al., 2026; Tan et al., 2026; Jin et al., 2026). Different sensors can observe the same entity through very different signals. A surface scanned under a controlled contact, for example, produces both audio and acceleration, each varying with scan speed, direction, and force (Eguchi et al., 2026a). If these observations support a shared physical representation, their predictions should preserve the surface's behavior under additional scans. This motivates evaluating a representation through the physical responses it predicts when its sensory source changes.

Embedding alignment measures proximity between representations, while reconstruction and attribute probes assess decoding and recoverable physical distinctions. Response prediction adds a behavioral test: do different sensory observations lead to consistent predictions for the same entity? Predictive-state and function-representation methods motivate describing a system through its responses (Littman et al., 2001; Ingebrand et al., 2024). Predictions that agree only near a population average carry little information about a particular surface, and agreement between two predictions does not establish their accuracy. We therefore compare same-entity responses with wrong-entity and population controls, and separately evaluate error against measured responses.

We instantiate this approach on Cluster Haptic using audio and acceleration recorded under non-overlapping sets of scan conditions. A sensor-specific mapping, which we call a compiler, infers a coordinate in a common response chart. A shared response model predicts additional scans (Figure 1a). The chart and response model are learned from training surfaces and frozen before the four modality–panel compilers are fitted independently, without entity identifiers or a cross-modal matching loss. This common response supervision makes the predictions comparable while allowing each sensor to infer a new surface's behavior from its own 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.

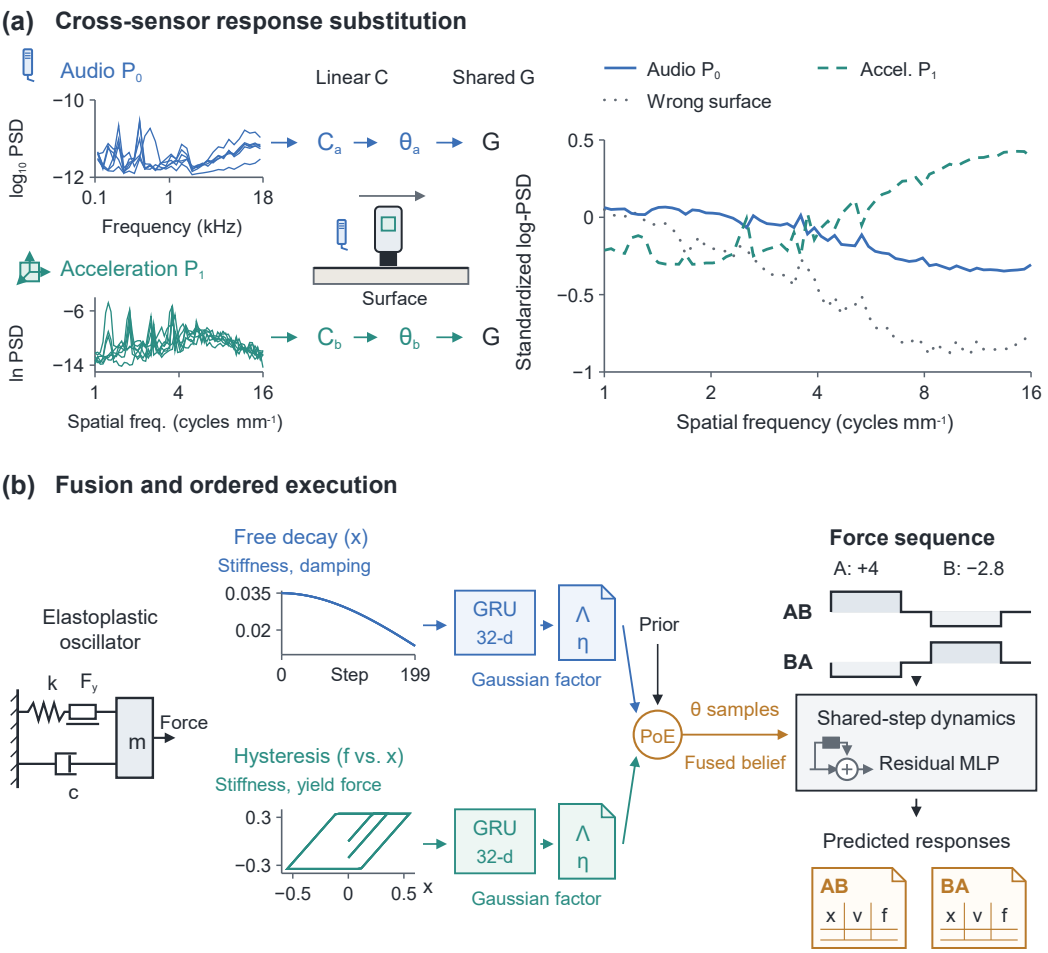


Figure 1: **Shared response coordinates across sensors and action orders.** (a) Independently fitted audio and acceleration compilers infer separate coordinates, compared through a shared frozen response model. The curves illustrate one query's normal acceleration spectrum. Four branches cover both cross-panel orientations. (b) Free decay is invariant to yield force F_y, and settled hysteresis to damping c. GRUs infer Gaussian factors (Λ, η), combined with the prior by a product of experts (PoE). Fused coordinate samples and force inputs enter shared step dynamics. Response sheets depict the post-action output format (x, v, f); physical drawings and parameter sheets are schematic.

Using a physical representation for sequential prediction requires a model of how each input changes the state encountered by the next input. We examine this requirement in an elastoplastic system, where familiar force pulses occur in new orders (Figure 1b). An ambiguity construction separates recovery of entity information from identification of an unseen composition law. We compare shared step dynamics with whole-program prediction and evaluate fusion of complementary observations. Each study uses its own chart and executor to test a different aspect of shared physical representation.

We evaluate whether shared physical representations preserve entity-specific responses across sensors and support new action orders. The real-data study changes the sensory source while holding the response model fixed. The controlled study compares executors with fixed entity coordinates, then evaluates complementary observations through a frozen executor. On Cluster Haptic, all four branches predict measured responses better than a population baseline, and every test surface has a smaller same-surface distance than its mean wrong-surface distance. In the controlled system, the whole-program predictor fits observed programs better, but shared step dynamics generalize better to both unseen transitions in all three initializations. Complementary fusion further improves prediction, with the diagonal belief performing best among the tested formulations.

2 Related Work

Physical world models learn representations for transition prediction, mechanical-property recovery, and tactile actions (Shang et al., 2026; Tan et al., 2026; Jin et al., 2026). Cross-modal interchangeability asks whether different sensors support a common task. Connect–Collapse–Corrupt learns cross-modal tasks from unimodal data, and function-contrastive learning connects views of a common function (Zhang et al., 2024; Gondal et al., 2021). Partial-view causal representation learning examines shared content under partial observability (Yao et al., 2024). We fit sensor mappings separately to a fixed response model and compare predictions beyond their input observations. Wrong-entity controls and measured responses test specificity and accuracy.

Describing a physical entity by its responses follows a predictive view of representation. Predictive-state models and causal states characterize systems through possible future observations (Littman et al., 2001; Downey et al., 2017; Shalizi & Crutchfield, 2001). Function Encoders, neural processes, and DeepONet learn function representations from input–output observations (Ingebrand et al., 2024; Garnelo et al., 2018; Lu et al., 2021). We build on this view by comparing the physical responses predicted from different sensory observations.

Gaussian products of experts fuse evidence (Hinton, 2002; Wu & Goodman, 2018; Sutter et al., 2021), while hidden-parameter state-space models represent uncertainty over task properties (Shaj et al., 2022). Compositional-generalization studies evaluate new combinations of familiar components (Lake & Baroni, 2018; Hupkes et al., 2020; Keysers et al., 2020). Identifiability results characterize assumptions for recovering latent structure from observations (Locatello et al., 2019; Khemakhem et al., 2020). Our controlled study evaluates whether complementary observations improve response prediction and whether learned dynamics transfer to new action orders.

3 Shared Physical Representations Through Responses

3.1 Response Coordinates and Prediction

We summarize each entity's responses to fitting queries by a low-dimensional coordinate. A sensor-specific compiler estimates this coordinate from observations, and a shared response model uses it to predict the response to a requested query.

We describe an entity by its responses under a specified apparatus and query family. Let $\omega \in \Omega$ denote an entity, $\mathcal{A}$ the set of basic interventions, and $q \in \mathcal{A}^*$ a finite intervention program. An apparatus α fixes the response sensor, contact geometry, initial condition, and preprocessing. The response law is

$$\mathcal{R}^{\alpha}_{\omega}(q) := \mathcal{L}\big(Y_{0:T(q)} \mid \mathrm{do}(q), \omega, \alpha\big)\,. \tag{1}$$

Here $\mathcal{L}$ denotes the response distribution, and $Y_{0:T(q)}$ the observations produced by q. In Cluster, q specifies a scan condition; in the controlled system, it specifies a sequence of force pulses. We represent these responses by preprocessed spectra or trajectory vectors. A response chart assigns coordinates θ to training-entity response profiles, and a response model G maps a coordinate and query to a prediction. We call G an executor because it produces the response to the requested intervention.

Given observations $e_{m,P}$ from modality m on evidence panel P, a compiler $C_{m,P}$ predicts the coordinate:

$$\widehat{R}_{m,P}(q) = G(C_{m,P}(e_{m,P}), q)\,. \tag{2}$$

Sharing means that the sensor-specific estimates are interpreted through the same response chart and executor. We assess this relationship through their predictions, rather than requiring equality of the inferred coordinates. The coordinates describe response variation and need not equal physical parameters such as stiffness or damping. The executor receives neither modality nor panel identifiers. Appendix A gives the associated response-equivalence relation.

3.2 Cross-Sensor Consistency and Entity Specificity

Response substitution changes the sensor and evidence panel while keeping the entity and prediction queries fixed. We construct the chart and executor from training entities and freeze them before

fitting each compiler separately. Two modalities a and b observe disjoint panels P_0 and P_1. Their predictions are compared in both orientations, $(a, P_0; b, P_1)$ and $(a, P_1; b, P_0)$, over queries $\mathcal{Q}_{\text{bridge}}$ outside both panels. Thus agreement cannot be explained by presenting the same query instances to both branches.

For stacked response predictions R, R', we use a squared distance normalized by training-entity variation:

$$D_{\mathcal{Q}_{\text{bridge}}}(R, R') = \frac{\|R - R'\|_F^2}{Z}, \qquad Z = \frac{1}{N_{\text{tr}}} \sum_{j=1}^{N_{\text{tr}}} \|G_{\mathcal{Q}_{\text{bridge}}}(\theta_j) - G_{\mathcal{Q}_{\text{bridge}}}(0)\|_F^2.$$

Here $G_{\mathcal{Q}_{\text{bridge}}}(0)$ is the population chart-center prediction. We average the two cross-panel distances over matching entities to obtain D_{same}, and over different entities to obtain D_{wrong}. The average distance from each branch to the population prediction is D_{pop}. The response-substitution criterion is

$$D_{\text{same}} < \min\{D_{\text{wrong}}, D_{\text{pop}}\}. \tag{3}$$

The two controls distinguish entity-specific agreement from agreement near a common center. Prediction error against withheld targets supplies the separate accuracy measure, normalized by population prediction error. Appendix B specifies the finite estimators and evidence allocation.

3.3 Complementary Information and New Action Orders

For complementary observations, each compiler represents uncertainty over response coordinates with a probability distribution. Combining these beliefs can reduce uncertainty and improve response prediction. We evaluate this fusion benefit separately from cross-sensor substitution: complementary observations need not be individually sufficient to recover the same response function. Even when observations identify an entity's response coordinates, they may leave the effect of a new action order undetermined.

Proposition 1 (Ambiguity of unseen action orders)**.** *Low-rank response compression on observed queries, strict squared-loss Bayes-risk reduction from each modality, and strict posterior-trace reduction under fusion do not suffice to identify an unseen ordered query.*

Let observed primitive responses be $\mathcal{R}(A) = U$ and $\mathcal{R}(B) = V$, with independent Gaussian coordinates. Two modalities measure U and V separately with additive Gaussian noise. The observed responses have an exact two-dimensional chart; each measurement reduces Bayes risk, and fusion reduces posterior trace. Two systems can agree on these observations while assigning opposite composition laws:

$$\mathcal{R}_\sigma(AB) = U + V + \sigma\gamma UV, \qquad \mathcal{R}_\sigma(BA) = U + V - \sigma\gamma UV, \quad \sigma \in \{-1, +1\},\ \gamma > 0.$$

The systems induce identical evidence and primitive-response distributions but opposite order effects. A learner using the common observed law produces the same prediction for both systems and must err on at least one. Appendix A.2 gives the risk calculation. The observations leave the composition rule undetermined even if the entity coordinates (U, V) are known. The controlled experiment examines whether reusing a learned state-update rule supports prediction under a new action order.

4 Shared Physical Responses Across Sensors

4.1 Surface Observations and Test Responses

Cluster Haptic records synchronized acceleration, audio, force, and position while a controlled three-axis machine scans 118 surfaces under varied direction, velocity, normal force, and repeat conditions (Eguchi et al., 2026a;b). We split surface identities into 80 training, 19 development, and 19 test surfaces. The response target is a three-axis log spatial acceleration spectrum. Audio and acceleration provide different observations from which to infer this common target response.

Of 80 scan queries, 13 contain the reference condition or a change in one scan coordinate. Their fixed allocation gives disjoint evidence panels P_0 and P_1 with six and seven queries (Figure 2a). The remaining 67 multi-factor queries form $\mathcal{Q}_{\text{bridge}}$. A velocity change alone is an axis query, for example, whereas a joint velocity and normal-force change belongs to the bridge set. Compilers

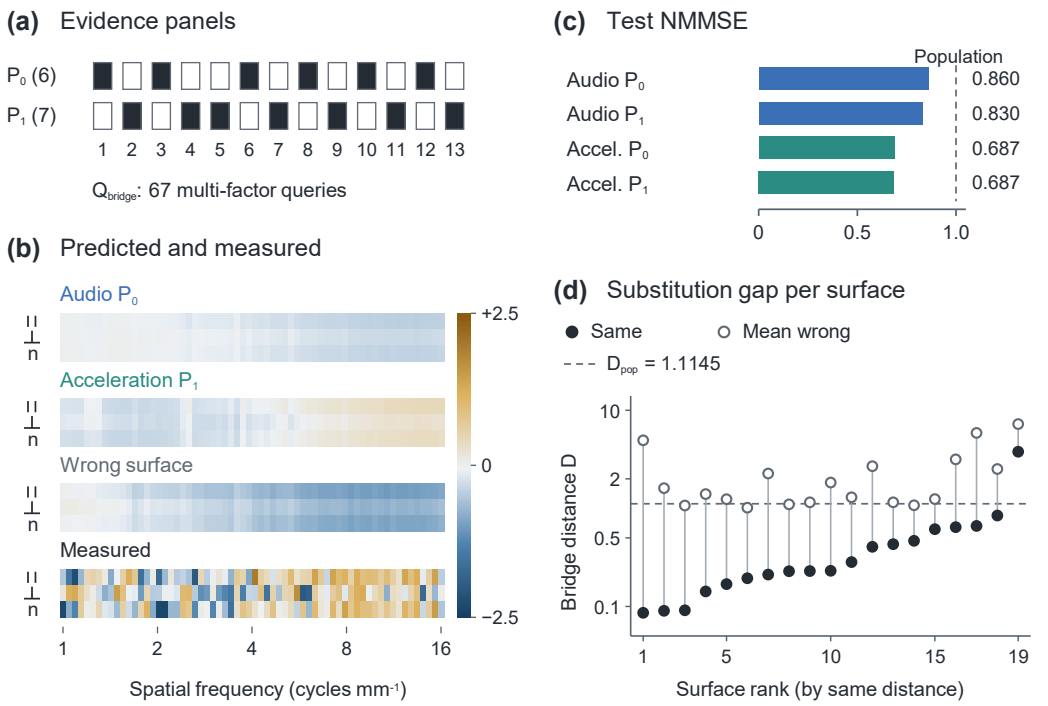


Figure 2: **Surface-specific responses inferred from audio and acceleration.** (a) The 13 axis queries are split into disjoint evidence panels; the 67 remaining queries form $\mathcal{Q}_{\text{bridge}}$. (b) Predicted and measured standardized log spatial spectra for surface 20 at $(180°, 20\,\text{mm/s}, 500\,\text{mN})$, with wrong surface 56. Measured responses use repeat 1. Rows denote parallel ($\parallel$), perpendicular ($\perp$), and normal (n) acceleration. All four groups share the color range $[-2.5, 2.5]$ and a logarithmic frequency axis. Predictions are constrained by the frozen low-rank response chart; measured targets are shown without chart compression. (c) Test NMMSE on the same 19 surfaces, with population error normalized to one. (d) Same-surface and mean wrong-surface distances for 19 test surfaces, sorted by same-surface distance. Thin lines connect paired values. Same-surface distance is lower for all 19 surfaces. The population response-distance reference $D_{\text{pop}} = 1.1145$ uses a different normalization from the NMMSE baseline.

receive repeat-0 evidence; repeat-1 measurements supply response targets. Test bridge responses are withheld until the chart, compilers, and panel assignment are fixed.

A training-only rank-four basis compresses each scan, and a rank-three SVD represents surface profiles concatenated across all 80 queries. These fixed linear maps form the shared chart and executor. Four linear-kernel compilers map audio or acceleration from either evidence panel to this chart. The training surfaces cover every query, so the experiment tests recovery of a new surface's responses from limited observations. The bridge queries are absent from the test surface's evidence, but their scan conditions are known to the response model. Both cross-panel orientations contribute equally to evaluation; Appendix B gives signal processing, exact estimators, and development results.

4.2 Cross-Sensor Consistency and Predictive Accuracy

Figure 2b compares predictions from audio and acceleration with a wrong-surface control and the measured response for one test surface and bridge query. All four branches predict measured responses better than the population baseline (Figure 2c). Test NMMSE is 0.8602 and 0.8298 for audio P_0 and P_1, and 0.6873 and 0.6870 for acceleration. Acceleration is more accurate under both panel assignments. Appendix B reports the corresponding development results.

The cross-sensor predictions also preserve surface identity. Mean same-surface distance is $D_{\text{same}} = 0.5161$, compared with $D_{\text{wrong}} = 2.3104$ and $D_{\text{pop}} = 1.1145$. The overall mean wrong-surface

distance is approximately 4.5 times the same-surface distance. Each of the 19 test surfaces has a smaller same-surface distance than its mean wrong-surface distance, with a median per-surface ratio of 5.2 (Figure 2d). Cross-sensor consistency preserves surface-specific behavior while allowing the modalities to differ in predictive accuracy. Section 5 uses a separate controlled system to examine how response coordinates support prediction when familiar actions occur in a new order.

5 Generalization Across Action Orders

5.1 Controlled Dynamics and Complementary Observations

Cluster tests inference of a new entity within a response family whose scan conditions were covered during training. The controlled study asks whether a supplied entity coordinate supports prediction when familiar actions form new transitions. The governing equations stay fixed, so this tests reuse of learned dynamics across action orders. We use an elastoplastic system whose stiffness k, damping c, and yield force F_y vary across entities. For unit mass and plastic displacement p, restoring force and motion satisfy

$$f = \mathrm{clip}(k(x-p), -F_y, F_y), \qquad \dot{v} = u - cv - f, \qquad \dot{x} = v. \tag{4}$$

The simulator updates plastic displacement after each velocity-first Euler step; Appendix C.1 gives the exact recursion and initial conditions. Free decay starts from a small release that remains elastic, so its evidence is exactly invariant to F_y. Hysteresis records fully settled quasistatic responses, for which velocity is zero and damping has no effect. Both observations retain stiffness information. Training, development, and test entities occupy disjoint Cartesian grids of sizes 5^3, 4^3, and 4^3.

Three force pulses A, B, and C form finite programs. The 17 fitting programs contain all three primitives in all positions but only self transitions and $B \leftrightarrow C$. Development introduces $A \leftrightarrow C$. Test programs are AB and BA; neither adjacency appears in fitting or development, including in longer programs. After each program, we simulate 60 additional steps with zero input and record displacement, velocity, and restoring force. The driven segment is excluded from scoring. Plastic memory makes the post-action response depend on which pulse occurs first, measured by the response difference

$$\Delta_\omega^{A,B} = r_\omega(AB) - r_\omega(BA). \tag{5}$$

Here $r_\omega(q)$ is the normalized post-action trajectory, and Δ is the response commutator. Its RMS across test entities, time, and channels is 0.62016. Both parameters hidden from the single modalities affect this post-action response, providing a setting in which complementary observations can improve prediction.

A rank-three chart retains 98.95% of centered fitting-response variation. To assess execution separately from sensory inference, an oracle supplies an entity's coordinate from its fitting-query response profile. For the executor comparison in Section 5.2, NMSE divides prediction MSE by the error of a common query-only predictor within each split. This evaluation asks whether the dynamics model can use the response coordinate before introducing uncertainty from sensor observations.

5.2 Reusing Step Dynamics on Unseen Transitions

This diagnostic compares two executors using the same response coordinate. The shared-step, or step-factorized, model applies a residual MLP repeatedly,

$$s_{t+1} = s_t + E_\phi(s_t, u_t/4, \theta),$$

using normalized observables $s_t = (x_t, v_t, f_t)$, the rescaled force input, and the coordinate. Two hidden layers of 128 SiLU units map these seven inputs to three state increments, giving 17,923 trainable parameters. The coordinate stays fixed throughout a program, while the state carries its action history.

Let $\mathcal{T}_{\mathrm{fit}}$ denote the empirical distribution of simulator transitions from fitting entities and programs. Training minimizes

$$\mathcal{L}_{\mathrm{step}}(\phi) = \mathbb{E}_{(s,u,s',\theta)\sim\mathcal{T}_{\mathrm{fit}}}\left[\frac{1}{3}\left\|E_\phi(s, u/4, \theta) - (s'-s)\right\|_2^2\right]. \tag{6}$$

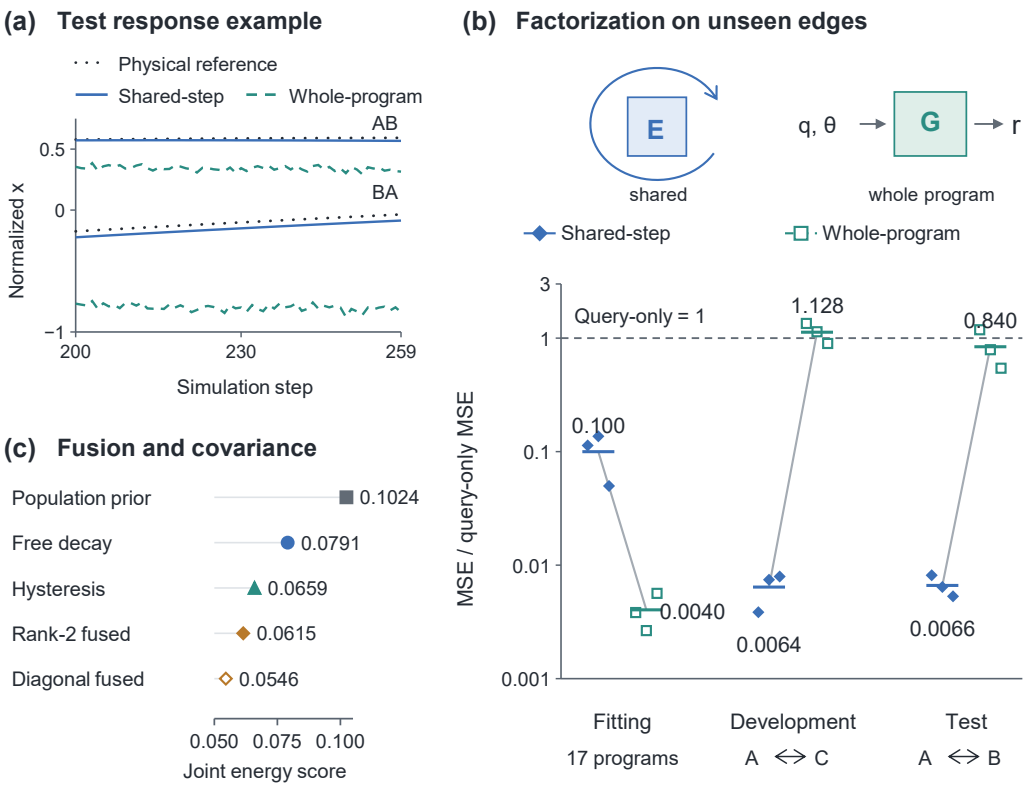


Figure 3: **Action-order generalization and complementary fusion.** (a) Simulator displacement and oracle-coordinate predictions for test entity 32, seed 0, on both action orders. The entity index was fixed before training; seed 0 is the first initialization. All curves share training-based normalization. (b) Three independent initializations per executor at a 30,000-update budget. Small points show individual runs; horizontal bars and labels show their means. Both executors use the same query-only baseline within each split. Checkpoints are selected by development post-action response MSE. The example in (a) uses these selected models. (c) Joint energy score for fusion through a separately trained frozen executor, with lower values better. The diagonal compiler is trained separately from the rank-two-factor compiler.

Minibatches sample 128 transitions uniformly with replacement. The targets include intermediate states during the driven segment as well as the post-action response. Thus the shared-step executor receives local transition supervision throughout each fitting trajectory. At evaluation, it starts from the normalized zero physical state and recursively uses its own predictions for the full force sequence and post-action response. No measured intermediate state is supplied during this rollout.

The whole-program executor embeds each action token into 32 dimensions and encodes the ordered word with a 32-dimensional GRU. A $35 \to 52 \to 180$ SiLU head combines its final hidden state with the three response coordinates to predict the 60×3 post-action trajectory. This model has 17,844 parameters and minimizes post-action response MSE over batches of 128 entity–program pairs. Both executors use the same chart and fitting programs; the shared-step model receives intermediate transition targets, while the whole-program model receives post-action trajectory targets.

Figure 3a shows both models predicting the same test entity. The shared-step predictions follow the simulator more closely; the whole-program predictor underestimates displacement under both orders.

For the architecture comparison in Figure 3b, we train each executor from three independent initializations for 30,000 updates. Both use AdamW with learning rate 10^{-3} and weight decay 10^{-5}, selecting between 15,000 and 30,000 updates by development post-action response MSE. A separately trained query-only GRU supplies one common denominator per split. It receives the action word without an entity coordinate and predicts the complete post-action response. All checkpoint selections are

finalized before test responses are evaluated. Appendix C.4 specifies this comparison and reports each run.

The whole-program executor fits the 17 observed programs more accurately, with mean NMSE 0.00404 compared with 0.09990 for shared-step execution. The ranking reverses on both unseen edges in every initialization. Development NMSE is 1.128 versus 0.00641; test NMSE is 0.840 versus 0.00663, in the same model order. Test MSE before division by the common baseline is 0.76948 versus 0.00607. The whole-program mean improves on the query-only test baseline, but its errors remain much larger than those of shared-step execution. This reversal therefore holds for the original MSE as well as the normalized comparison.

Shared step dynamics carry the effect of the first action into the execution of the second through the evolving state. In AB, pulse B acts on the displacement, velocity, and restoring force left by A; reversing the order changes that state. The same update rule processes both sequences. Reusing this rule provides a mechanism for applying learned dynamics to new action orders. The comparison tests this recursive structure together with its local supervision; matching the chart and update budget does not separate those two design choices.

5.3 Combining Complementary Evidence

For the fusion analysis, we freeze a separately trained shared-step executor selected by development oracle error under a 30,000-update budget. Two 32-dimensional GRU compilers map free-decay and hysteresis observations to Gaussian information factors over the response coordinates (Figure 1b). Appendix C reports the executor's training-budget analysis.

A standard-normal prior combines these factors by adding their natural parameters under the Gaussian product-of-experts rule (Hinton, 2002; Wu & Goodman, 2018; Sutter et al., 2021):

$$\Sigma_S = (I + \textstyle\sum_{m \in S} \Lambda_m)^{-1}, \qquad \mu_S = \Sigma_S \textstyle\sum_{m \in S} \eta_m. \tag{7}$$

Each factor uses $\Lambda_m = L_m L_m^\top$ with $L_m \in \mathbb{R}^{3\times 2}$. Both compilers are trained jointly on true fitting responses through the frozen executor, with single-modality and fusion losses.

Joint energy score evaluates the predicted AB/BA response distribution (Gneiting & Raftery, 2007). Each coordinate sample is executed under both orders, yielding a joint vector of $2 \times 60 \times 3 = 360$ response values. The same sampled entity therefore links the two predicted trajectories. Mean-response NMSE evaluates their predictive means relative to the executed population prior. Order-effect error compares the difference of those means with the true $AB - BA$ response, using the true order-effect RMS as its scale. A zero predicted order effect has error one. These definitions retain both response accuracy and the difference induced by changing the action order (Appendix C).

Combining observations improves prediction over either source alone (Figure 3c; Table 1a). Response NMSE decreases from 0.5492 for free decay and 0.4066 for hysteresis to 0.3193 after fusion; joint energy score also improves. Replacing evidence with an observably different sequence from another entity matched in stiffness raises NMSE to 0.8398. This control supports the contribution of entity-specific complementary information to the fusion improvement.

The controlled blind spots make this gain physically interpretable. Free decay supplies damping information that settled hysteresis lacks, while hysteresis supplies yield information absent from the elastic release. Both quantities affect the post-action response. Fusion reduces posterior trace by 68% relative to the more informative single modality. The information sum is full rank, with minimum eigenvalue 0.5867, so the learned factors jointly constrain all three chart coordinates.

The separately trained diagonal model gives the lowest errors on all three reported metrics, including order-effect prediction (Table 1b). Both fused distributions also outperform the point-valued control on energy score and NMSE. The diagonal factors allow rank three, whereas the model with off-diagonal information uses rank-two factors. This comparison favors the diagonal formulation while changing both the per-modality rank constraint and the allowed off-diagonal structure.

Table 1: **Fusion and belief parameterization for test AB/BA responses.** Results on 64 test entities with the frozen executor used for fusion. NMSE is relative to the executed population prior; order-effect error is relative RMS error. Lower is better. Single-modality order-effect errors were not reported. Diagnostic controls are reported in Appendix C.5.

(a) Response prediction

Model	Joint ES ↓	NMSE ↓
Population prior	0.10239	1.0000
Free decay	0.07909	0.5492
Hysteresis	0.06594	0.4066
Rank-2 fused	0.06147	0.3193
Diagonal fused	**0.05456**	**0.2873**
Point fused	0.08484	0.3736

(b) Order-effect prediction

Model	Relative RMS error ↓
Population prior	0.37388
Rank-2 fused	0.18165
Diagonal fused	**0.17396**
Point fused	0.20158

6 Discussion and Limitations

In Cluster, sharing means recovering surface-specific acceleration responses from either sensory source. This evaluation is asymmetric: acceleration evidence shares its sensing modality with the response target, while audio evidence does not. This may contribute to acceleration's lower prediction error. Using audio responses as the target would test whether the same-versus-wrong distance gap persists under the reciprocal choice of response apparatus.

Selecting an executor by fitting error would favor the whole-program predictor, whereas evaluation on new action transitions favors shared-step execution. The shared-step model carries action history through its recursively updated state. A targeted follow-up would train the same model with only a post-action rollout loss, testing how much its advantage depends on local transition supervision.

Fusion determines the entity information available to the executor. The exact blind spots create a reason to combine observations, and fused beliefs improve response prediction over either single modality. The fused model also predicts the order effect more accurately than the population predictor. The diagonal formulation gives the best result among the tested belief models. Together with Proposition 1, these comparisons separate the complementary observations that constrain an entity from the execution rule that maps its coordinates and an action sequence to a response.

The present evidence covers one real apparatus, two modalities, and 19 test surfaces, together with a controlled system evaluated on 64 test entities. Cluster tests new entities within a known query grid; the controlled system additionally withholds action transitions. Extending this design to real sequential contact tasks would connect the controlled analysis with richer interaction dynamics.

7 Conclusion

We evaluated shared physical representations through responses predicted from different sensory observations. On Cluster Haptic, separately fitted audio and acceleration compilers preserve surface-specific behavior and improve on population prediction. In a separate controlled system, shared step dynamics predict new action orders more accurately than a whole-program predictor across three independent initializations. Complementary fusion improves prediction, with the diagonal belief performing best among the evaluated fusion models.

AI USE STATEMENT

In this work, we used generative AI tools to support research ideation and execution, retrieve and identify relevant literature, and edit the manuscript for clarity and readability. We reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Carlton Downey, Ahmed Hefny, Byron Boots, Geoffrey J. Gordon, and Boyue Li. Predictive state recurrent neural networks. In *Advances in Neural Information Processing Systems*, volume 30, pp. 6053–6064. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/2bb0502c80b7432eee4c5847a5fd077b-Paper.pdf.

Michikuni Eguchi, Tomohiro Hayase, Yuichi Hiroi, and Takefumi Hiraki. Cluster haptic texture dataset: Haptic texture dataset with varied velocity-direction sliding contacts. *Scientific Data*, 13(1):756, 2026a. doi: 10.1038/s41597-026-06760-z. URL https://doi.org/10.1038/s41597-026-06760-z.

Michikuni Eguchi, Tomohiro Hayase, Yuichi Hiroi, and Takefumi Hiraki. Cluster haptic texture dataset: Haptic texture dataset with varied velocity-direction sliding contacts. https://doi.org/10.6084/m9.figshare.29438288.v5, 2026b. Dataset, version 5.

Marta Garnelo, Jonathan Schwarz, Dan Rosenbaum, Fabio Viola, Danilo J. Rezende, S. M. Ali Eslami, and Yee Whye Teh. Neural processes. *arXiv preprint arXiv:1807.01622*, 2018. doi: 10.48550/arXiv.1807.01622. URL https://arxiv.org/abs/1807.01622.

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. URL https://doi.org/10.1198/016214506000001437.

Muhammad Waleed Gondal, Shruti Joshi, Nasim Rahaman, Stefan Bauer, Manuel Wüthrich, 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.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. doi: 10.1162/089976602760128018. URL https://doi.org/10.1162/089976602760128018.

Dieuwke Hupkes, Verna Dankers, Mathijs Mul, and Elia Bruni. Compositionality decomposed: How do neural networks generalise? *Journal of Artificial Intelligence Research*, 67:757–795, 2020. doi: 10.1613/jair.1.11674.

Tyler Ingebrand, Amy Zhang, and Ufuk Topcu. Zero-shot reinforcement learning via function encoders. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 21007–21019. PMLR, 2024. URL https://proceedings.mlr.press/v235/ingebrand24a.html.

Lei Jin, Yiding Ma, Xin Zhang, Chen Gao, Wei Wu, and Yong Li. TacWAM: Anchor-guided world action model with mechanics-aware tactile prediction. *arXiv preprint arXiv:2607.28391*, 2026. doi: 10.48550/arXiv.2607.28391. URL https://arxiv.org/abs/2607.28391.

Daniel Keysers, Nathanael Schärli, Nathan Scales, Hylke Buisman, Daniel Furrer, Sergii Kashubin, Nikola Momchev, Danila Sinopalnikov, Lukasz Stafiniak, Tibor Tihon, Dmitry Tsarkov, Xiao Wang, Marc van Zee, and Olivier Bousquet. Measuring compositional generalization: A comprehensive method on realistic data. In *International Conference on Learning Representations*, 2020.

Ilyes Khemakhem, Diederik P. Kingma, Ricardo Pio Monti, and Aapo Hyvärinen. Variational autoencoders and nonlinear ICA: A unifying framework. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2207–2217, 2020.

Brenden M. Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2873–2882, 2018.

Michael L. Littman, Richard S. Sutton, and Satinder Singh. Predictive representations of state. In *Advances in Neural Information Processing Systems*, volume 14, pp. 1555–1561. MIT Press, 2001. URL https://proceedings.neurips.cc/paper_files/paper/2001/file/1e4d36177d71bbb3558e43af9577d70e-Paper.pdf.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4114–4124, 2019.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021. doi: 10.1038/s42256-021-00302-5. URL https://doi.org/10.1038/s42256-021-00302-5.

Vaisakh Shaj, Dieter Büchler, Rohit Sonker, Philipp Becker, and Gerhard Neumann. Hidden parameter recurrent state space models for changing dynamics scenarios. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=ds8yZOUsea.

Cosma Rohilla Shalizi and James P. Crutchfield. Computational mechanics: Pattern and prediction, structure and simplicity. *Journal of Statistical Physics*, 104(3–4):817–879, 2001. doi: 10.1023/A:1010388907793. URL https://arxiv.org/abs/cond-mat/9907176.

Shuyao Shang, Yuqi Wang, Ruopeng Gao, Xu Chen, Tieniu Tan, Lue Fan, and Zhaoxiang Zhang. PhiZero: A world model built around physical language. *arXiv preprint arXiv:2607.28624*, 2026. doi: 10.48550/arXiv.2607.28624. URL https://arxiv.org/abs/2607.28624.

Thomas M. Sutter, Imant Daunhawer, and Julia E. Vogt. Generalized multimodal ELBO. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=5Y21V0RDBV.

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. *arXiv preprint arXiv:2607.27017*, 2026. doi: 10.48550/arXiv.2607.27017. URL https://arxiv.org/abs/2607.27017.

Mike Wu and Noah D. Goodman. Multimodal generative models for scalable weakly-supervised learning. In *Advances in Neural Information Processing Systems*, volume 31, pp. 5580–5590, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/hash/1102a326d5f7c9e04fc3c89d0ede88c9-Abstract.html.

Dingling Yao, Danru Xu, Sébastien Lachapelle, Sara Magliacane, Perouz Taslakian, Georg Martius, Julius von Kügelgen, and Francesco Locatello. Multi-view causal representation learning with partial observability. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=OGtnhKQJms.

Yuhui Zhang, Elaine Sui, and Serena Yeung-Levy. Connect, collapse, corrupt: Learning cross-modal tasks with uni-modal data. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ttXg3SKAg5.

A Formal Definitions and Ambiguity Construction

A.1 Response Equivalence

For a fixed apparatus α and query family $\mathcal{Q}$, define

$$\omega \equiv_{\alpha,\mathcal{Q}} \omega' \quad \Longleftrightarrow \quad \mathcal{R}^{\alpha}_{\omega}(q) = \mathcal{R}^{\alpha}_{\omega'}(q) \quad \text{for every } q \in \mathcal{Q}.$$

The quotient $\Omega/\equiv_{\alpha,\mathcal{Q}}$ groups entities with the same response law under these observations. A response chart gives coordinates to the response profiles represented by the training data. Its scope is therefore conditional on both the apparatus and the query family. For a belief ν over coordinates, the predictive law over the stacked responses is the pushforward $(G_{\mathcal{Q}})_{\#}\nu$.

A.2 Proof of Proposition 1

Let $U, V \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$ and let the two modalities observe

$$E_1 = U + \epsilon_1, \qquad E_2 = V + \epsilon_2, \qquad \epsilon_1, \epsilon_2 \overset{\text{iid}}{\sim} \mathcal{N}(0, \tau^2), \qquad 0 < \tau^2 < \infty. \tag{8}$$

For observed primitive queries, set $\mathcal{R}(A) = U$ and $\mathcal{R}(B) = V$. The response profile lies exactly in the rank-two chart $Z = (U, V)$. Writing $\alpha = \tau^2/(1+\tau^2)$, the unimodal and fused posterior covariances are

$$\Sigma_1 = \operatorname{diag}(\alpha, 1), \quad \Sigma_2 = \operatorname{diag}(1, \alpha), \quad \Sigma_{12} = \alpha I. \tag{9}$$

Thus $\operatorname{tr}\Sigma_{12} = 2\alpha < 1 + \alpha = \operatorname{tr}\Sigma_1 = \operatorname{tr}\Sigma_2$.

Now index two systems by an unobserved composition law $\sigma \in \{-1, +1\}$:

$$\mathcal{R}_{\sigma}(AB) = U + V + \sigma\gamma UV, \tag{10}$$

$$\mathcal{R}_{\sigma}(BA) = U + V - \sigma\gamma UV, \qquad \gamma > 0. \tag{11}$$

The systems induce the same joint distribution over all modality evidence and observed-query responses. Both ordered responses have population mean zero. Under squared loss, each modality strictly reduces Bayes risk for either ordered response because, for example,

$$\mathbb{E}[\mathcal{R}_{\sigma}(AB) \mid E_1] = \frac{E_1}{1+\tau^2}, \tag{12}$$

whose variance $1/(1+\tau^2)$ is the Bayes-risk reduction relative to the zero population predictor.

The commutator is $C_{\sigma} = 2\sigma\gamma UV$. Any learner using only the common observed law produces the same predictor $\widehat{C}$ under both signs. Pointwise,

$$\frac{(\widehat{C} - 2\gamma UV)^2 + (\widehat{C} + 2\gamma UV)^2}{2} = \widehat{C}^2 + 4\gamma^2 U^2 V^2. \tag{13}$$

Taking expectations and using $\mathbb{E}[U^2V^2] = 1$ shows that at least one indistinguishable system has commutator MSE at least $4\gamma^2$. Exact low-rank compression, useful marginal evidence, and reduced fused uncertainty therefore leave the unseen composition law unidentified. □

B Cluster Haptic Implementation and Additional Results

We use Cluster Haptic v5 with a category-balanced split of 80 training, 19 development, and 19 test surface identities. All response normalizers, chart bases, and compiler statistics use the training identities.

Response construction. We rotate acceleration into path-parallel, path-perpendicular, and surface-normal axes. The central two-thirds of the measured scan path is resampled at 5 kHz. Welch spectra use 20-mm Hann windows with 10-mm hops in spatial coordinates. Each axis is interpolated to 64 frequencies from 1 to 16 cycles/mm, producing a 3×64 natural-log spatial-PSD response on a geometric frequency grid. Audio evidence contains 32 $\log_{10}$ mean-PSD values over frequency bands from 100 Hz to 18 kHz.

Table 2: **Response prediction on Cluster Haptic.** Repeat-1 NMMSE on 19 development and 19 test surfaces, each evaluated on 67 bridge queries. The population chart-center baseline equals one. Test targets enter no fitting or model selection.

	Development		Test	
Modality	P_0	P_1	P_0	P_1
Audio	0.8636	0.8050	0.8602	0.8298
Acceleration	0.7328	0.7386	0.6873	0.6870
Population	1.0000	1.0000	1.0000	1.0000

Frozen chart and compilers. A rank-four basis compresses each standardized 192-dimensional acceleration response. Concatenating coefficients across all 80 queries gives a 320-dimensional surface profile; its centered, fixed-sign rank-three SVD defines $\theta \in \mathbb{R}^3$. For branch (m, P), the compiler uses standardized evidence dimension $d_{m,P}$ and the fixed kernel

$$K_{m,P}(x, x') = 1 + \frac{x^\top x'}{d_{m,P}}, \qquad \widehat{\theta} = K_{x,T}(K_{T,T} + 10^{-3} I)^{-1} \Theta_T.$$

The panel assignment is fixed before fitting. No surface identifier or material category enters a compiler.

Evidence and target separation. The safe cache contains repeat-0 axis evidence for all surfaces, training repeat-1 responses for all 80 queries, and development repeat-1 responses for the 67 bridge queries. Test repeat-1 responses are absent. The 13 axis queries are assigned to P_0 or P_1 by the fixed parity of their categorical query-coordinate codes, yielding six and seven queries. This allocation is fixed before fitting the four compilers. No surface identifier or material category is supplied as an input, although the branches share response-coordinate targets for the same training surfaces.

Response-distance estimators. For test entity i, write $\widehat{R}_u^{(i)} = [G(C_u(e_{i,u}), q)]_{q \in \mathcal{Q}_{\mathrm{bridge}}}$ for branch u. Let $\mathcal{O} = \{(a, P_0; b, P_1), (a, P_1; b, P_0)\}$ contain the two cross-panel orientations, $\mathcal{U}$ the four branches, and $\bar{R} = G_{\mathcal{Q}_{\mathrm{bridge}}}(0)$. Using the training-normalized squared distance of Section 3.2,

$$D_{\mathrm{same}} = \frac{1}{2N} \sum_{i=1}^{N} \sum_{(u,v) \in \mathcal{O}} D_{\mathcal{Q}_{\mathrm{bridge}}}(\widehat{R}_u^{(i)}, \widehat{R}_v^{(i)}),$$

$$D_{\mathrm{wrong}} = \frac{1}{2N(N-1)} \sum_{i \neq j} \sum_{(u,v) \in \mathcal{O}} D_{\mathcal{Q}_{\mathrm{bridge}}}(\widehat{R}_u^{(i)}, \widehat{R}_v^{(j)}),$$

$$D_{\mathrm{pop}} = \frac{1}{4N} \sum_{i=1}^{N} \sum_{u \in \mathcal{U}} D_{\mathcal{Q}_{\mathrm{bridge}}}(\widehat{R}_u^{(i)}, \bar{R}).$$

For $N = 19$, these average 38 same-entity, 684 wrong-entity, and 76 population terms. The per-surface wrong distance in Figure 2d first averages over the other 18 surfaces and then over the two orientations. Each surface's same distance is smaller than this mean wrong distance. This ordering does not require every individual wrong pairing to be farther away.

Prediction error and development results. NMMSE is the sum of squared errors across the standardized response features, queries, and entities divided by the corresponding population prediction error. Table 2 reports development and test values; Figure 2c shows the test comparison. This error normalization differs from the training-response variation used for $D_{\mathcal{Q}_{\mathrm{bridge}}}$.

Evaluation protocol. The split, query panels, normalizers, response chart, and compiler coefficients are fixed before accessing test repeat-1 targets. Those targets are subsequently extracted from the raw archive with the same feature pipeline to evaluate predictive accuracy. No chart, compiler, threshold, or panel assignment was revised after target access, and all four branches are reported.

C Controlled-System Implementation and Additional Results

C.1 Simulator State and Discrete Updates

The simulator stores displacement x_t, velocity v_t, plastic displacement p_t, and restoring force f_t. With unit mass and $h = \Delta t = 0.005$, one force-input step applies

$$v_{t+1} = v_t + h(u_t - cv_t - f_t), \qquad x_{t+1} = x_t + hv_{t+1}, \tag{14}$$

$$f_{t+1}^{\text{trial}} = k(x_{t+1} - p_t), \qquad f_{t+1} = \text{clip}(f_{t+1}^{\text{trial}}, -F_y, F_y), \tag{15}$$

$$p_{t+1} = \begin{cases} x_{t+1} - f_{t+1}/k, & |f_{t+1}^{\text{trial}}| > F_y, \\ p_t, & |f_{t+1}^{\text{trial}}| \le F_y. \end{cases} \tag{16}$$

Velocity is updated before displacement, using the previous restoring force. The force and plastic state are then updated at the new displacement. During elastic loading or unloading, p_t remains fixed; after either yield threshold is exceeded, the plastic update keeps the restoring force at that threshold. These are the steps implemented by `_orq_simulate_bank` and `_orq_update_force` in `pilot_cross_context.py`.

Each action program starts at $x_0 = v_0 = p_0 = f_0 = 0$. The stored trajectory contains $(x_{t+1}, v_{t+1}, f_{t+1})$ after each input step, and the final 60 stored states form the scored response. Free decay instead starts at $x_0 = 0.035$, $v_0 = p_0 = 0$, and $f_0 = kx_0$, with $u_t = 0$ throughout. Quasistatic hysteresis starts with $p = 0$ and imposes a 90-point displacement path $0 \to 0.55 \to -0.55 \to 0.35 \to 0$. Its four segments contain 17, 33, 27, and 13 equally spaced points, respectively; each excludes its right endpoint except the last. At each imposed displacement, the same force-return and plastic updates apply with velocity fixed to zero. No dynamic integration is used for this settled observation.

C.2 Entity and Query Splits

The simulator uses $\Delta t = 0.005$. Training entities form a 5^3 grid over $k \in [1.2, 2.4]$, $c \in [0.12, 0.52]$, and $F_y \in [0.18, 0.42]$. Development and test entities occupy disjoint interleaved 4^3 grids. The free-decay modality records 200 steps from $x_0 = 0.035$; its maximum elastic force remains below the minimum yield force. The hysteresis modality records a 90-point fully settled quasistatic path.

Primitive pulses have amplitudes $u_A = 4.0$, $u_B = -2.8$, and $u_C = 1.4$, each with 70 driven and 30 coast steps. The action-program split is

$$\begin{aligned} \mathcal{Q}_{\text{fit}} = \{&A, B, C, AA, BB, BC, CB, CC, AAA, BBB, BBC, BCB, \\ &BCC, CBB, CBC, CCB, CCC\}, \\ \mathcal{Q}_{\text{dev}} = \{&AC, CA, AAC, ACA, ACB, ACC, BCA, CAA, CAC, CCA\}, \\ \mathcal{Q}_{\text{test}} = \{&AB, BA\}. \end{aligned}$$

Length-three programs containing an AB or BA adjacency are excluded from fitting and development. The 17 fitting, 10 development, and two test programs, together with these 10 excluded programs, partition all 39 words of length one to three.

C.3 Exact Blindness and Response Relevance

For free decay, the initial displacement satisfies $\max_k k|x_0| < \min F_y$. Free decay can only reduce mechanical energy, so the trajectory remains elastic and $p = 0$ throughout. Its complete evidence sequence is therefore invariant to yield force. Every hysteresis point is evaluated after the system has fully settled. Velocity is exactly zero, the damping term vanishes, and the hysteresis sequence is invariant to c.

The test response remains sensitive to every hidden coordinate. Adjacent-level changes in stiffness, damping, and yield force produce normalized response RMS values 0.09509, 0.03925, and 0.09041. The AB/BA commutator RMS is 0.62016. Wrong-entity fusion is matched within stiffness and replaces evidence only when at least one complementary coordinate changes. The wrong-entity control includes 1,536 valid fused-entity terms and 3,072 response terms; every replacement changes observable evidence in at least one complementary coordinate.

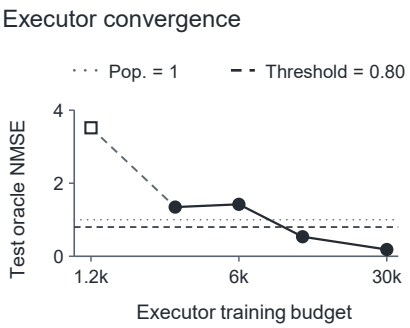


Figure 4: **Executor training-budget analysis.** Each point reports the checkpoint selected by development oracle NMSE within the stated budget. The initial 1,200-update model does not meet the predefined 0.80 threshold. Larger budgets are evaluated in a separate convergence analysis. Only reported test values are connected, including the increase at 6,000 updates. The denominator is the corresponding executor's zero-coordinate error.

C.4 Response Chart, Executor, and Compilers

Each fitting entity contributes a flattened normalized profile of size $17 \times 60 \times 3 = 3060$. The chart is centered before SVD; each component sign is fixed by requiring the largest-magnitude loading to be positive; scores are divided by their train standard deviations. The relative Frobenius residual is

$$\sqrt{\frac{\|R - \widehat{R}\|_F^2}{\|R - \bar{R}\|_F^2}} = 0.102265, \tag{17}$$

equivalent to unexplained energy 0.0104582.

The step executor receives normalized (x, v, f), the action scaled by $1/4$, and three chart coordinates. Its residual MLP has widths $7 \to 128 \to 128 \to 3$, SiLU activations, and 17,923 parameters. It receives neither plastic displacement p nor the true parameter vector (k, c, F_y). Training minimizes one-step increment MSE using simulator states and transitions from the complete fitting trajectories, including their driven portions. AdamW uses learning rate 10^{-3}, weight decay 10^{-5}, and batch size 128. At test time, the same network is applied recursively without observed intermediate states.

The initial update budget is 1,200. Development oracle NMSE selects between checkpoints at half and all of the budget, hence 600 and 1,200 updates for this run. The same selection rule is used for extended budgets of 3,000, 6,000, 12,000, and 30,000 updates. Figure 4 reports these budgets; the retained result series does not separately record the selected update count at each budget. The runner increases compiler budgets along with executor budgets and refits the compilers for each analysis. The splits, chart construction, step architecture, seed, and thresholds stay fixed. Oracle metrics bypass all compilers and therefore diagnose the executor.

For budgets of 1,200, 3,000, 6,000, 12,000, and 30,000 updates, test oracle NMSE is 3.5175, 1.3464, 1.4220, 0.5348, and 0.1844, respectively. Development oracle NMSE is 1.2238 at the initial budget and 0.0756 at the final budget.

Each modality encoder is a 32-dimensional GRU. In the rank-two-factor model, its head returns $L_m \in \mathbb{R}^{3\times 2}$ and $a_m \in \mathbb{R}^2$, with $\Lambda_m = L_m L_m^\top$ and $\eta_m = L_m a_m$. A separate diagonal model returns scales $l_m \in \mathbb{R}^3$ and observations $a_m \in \mathbb{R}^3$, with $\Lambda_m = \mathrm{diag}(l_m^2)$ and $\eta_m = l_m \odot a_m$; this permits rank three. Both are trained from their own initialization. The reported minimum fused-information eigenvalue refers to $\Lambda_a + \Lambda_b$, excluding prior precision I. It is distinct from the rank of either factor and from the covariance $(I + \Lambda_a + \Lambda_b)^{-1}$.

Compiler training compares the executed free-decay, hysteresis, and fused predictive distributions with true fitting responses, averaging their three energy scores. The executor remains frozen throughout this optimization. AdamW uses learning rate 10^{-3} and weight decay 10^{-5}; each update samples eight entities and one fitting query. The point control has separate three-dimensional modality heads and a learned fusion head, and is trained with the corresponding singleton-distribution score.

Table 3: **Executor comparison before baseline normalization.** MSE averages over entities, programs, 60 post-action steps, and three channels standardized using training responses. Selection uses development MSE. The fixed query-only row supplies a common denominator within each column.

Model	Seed	Selected update	Fitting MSE	Dev. MSE	Test MSE
Query-only	1001	15,000	0.038488	0.202754	0.916450
Shared-step	0	30,000	0.004363	0.000780	0.007462
Whole-program	0	15,000	0.000147	0.273157	1.086167
Shared-step	1	30,000	0.005254	0.001508	0.005883
Whole-program	1	30,000	0.000102	0.230814	0.724599
Shared-step	2	30,000	0.001918	0.001610	0.004869
Whole-program	2	15,000	0.000217	0.181938	0.497677

The direct diagnostic encodes both evidence sequences with GRUs, encodes the ordered action tokens with another GRU, and predicts the 60×3 post-action trajectory with an MLP. It bypasses both chart and executor. It is distinct from the whole-program executor in Figure 3b, which receives the response coordinate and ordered action tokens.

Three-seed executor comparison. We run this comparison separately from the budget and fusion analyses. The simulator, 125/64/64 entity grids, 17/10/2 program split, training-response normalizer, and rank-three chart remain fixed. Each architecture uses initialization seeds 0, 1, and 2, with a separate minibatch generator initialized from the corresponding seed. The shared-step network and transition loss are specified above. The whole-program network has a three-token embedding of width 32, a one-layer GRU with 32 hidden units, and a $35 \to 52 \to 180$ head with SiLU between its linear layers. Its input concatenates the final valid-token hidden state and the entity's three chart coordinates. Packing uses the true word lengths, so padding never acts as an extra primitive. The output is the normalized 60×3 post-action trajectory. The model has 17,844 trainable parameters, compared with 17,923 for the shared-step network.

Both architectures use AdamW with learning rate 10^{-3}, weight decay 10^{-5}, $(\beta_1, \beta_2) = (0.9, 0.999)$, and $\epsilon = 10^{-8}$. Each run makes 30,000 updates with batch size 128. Shared-step batches sample transitions; whole-program batches sample entity–program pairs. The development post-action response MSE selects between 15,000 and 30,000 updates for both models. All three shared-step runs select 30,000 updates; whole-program seeds 0, 1, and 2 select 15,000, 30,000, and 15,000 updates. No test entity response enters these choices.

The query-only baseline uses the same token embedding and GRU, with a $32 \to 52 \to 180$ head and 17,688 parameters. It receives no entity coordinate or sensory evidence. One run with seed 1001 uses the same optimizer, budget, and development selection rule, selecting 15,000 updates. At evaluation, its predicted post-action response for a query is identical for every entity. The same frozen predictions supply the denominator for all six executor runs, including for the unseen programs. Table 3 reports MSE before this division. We save both candidate checkpoints, selected weights, all minibatch losses, per-entity predictions, targets, and normalizers in `outputs/executor_comparison`; `executor_comparison.py` reproduces the experiment.

C.5 Predictive Scores and Normalization

For posterior $\mathcal{N}(\mu, \Sigma)$, a Cholesky factor maps the eight lexicographically ordered nodes in $\{-1, +1\}^3$ to $z_s = \mu + L_\Sigma s$, each with weight $w_s = 1/8$. Executing each node on AB and BA gives a $D = 2 \times 60 \times 3 = 360$ dimensional response vector. For predictive law F, the joint energy score is

$$\mathrm{ES}(F, r) = \frac{1}{\sqrt{D}} \left(\mathbb{E}\|X - r\|_2 - \frac{1}{2}\mathbb{E}\|X - X'\|_2 \right), \quad X, X' \overset{\text{iid}}{\sim} F. \tag{18}$$

The implementation evaluates the finite-distribution form

$$\sum_s w_s \frac{\|\widehat{r}_s - r\|_2}{\sqrt{360}} - \frac{1}{2} \sum_{s,t} w_s w_t \frac{\|\widehat{r}_s - \widehat{r}_t\|_2}{\sqrt{360}}. \tag{19}$$

The score is proper for this finite predictive distribution; the eight-node representation is not the exact nonlinear pushforward of a continuous Gaussian.

Three error normalizations serve different comparisons. For training budget B, let G_B be the executor selected by development error and let $\mathrm{MSE}_{\mathcal{E},\mathcal{Q}}$ average squared error over the evaluated entities, queries, time steps, and normalized channels. Figure 4 uses

$$\mathrm{OracleNMSE}(B) = \frac{\mathrm{MSE}_{\mathcal{E},\mathcal{Q}}[G_B(\theta_i^*, q), r_i(q)]}{\mathrm{MSE}_{\mathcal{E},\mathcal{Q}}[G_B(0, q), r_i(q)]}. \tag{20}$$

Both errors use the same selected executor and evaluation set. Its zero-coordinate prediction changes with B. The runner emits `oracle_response_mse` and `population_response_mse`, but the saved budget series retains their ratio only. The raw errors and selected update counts cannot be recovered from these ratios.

Figure 3b instead uses the query-only predictor $H(q)$ defined above. For executor $G_{m,s}$ with architecture m and seed s,

$$\mathrm{NMSE}_{m,s}(\mathcal{E}, \mathcal{Q}) = \frac{\mathrm{MSE}_{\mathcal{E},\mathcal{Q}}[G_{m,s}(\theta_i^*, q), r_i(q)]}{\mathrm{MSE}_{\mathcal{E},\mathcal{Q}}[H(q), r_i(q)]}. \tag{21}$$

The denominator is 0.0384883 on fitting programs, 0.2027545 on development programs, and 0.9164496 on test AB/BA. Within each split, it is identical for every architecture and seed. Figure 3b shows all three run values and their arithmetic mean. Because the denominator is shared, the relative ordering of MSE and NMSE is identical.

Table 1 uses mean-response MSE divided by the MSE of the executed population prior $\mathcal{N}(0, I)$. That prior is evaluated using the same eight nodes, so its mean prediction generally differs from $G(0, q)$ for a nonlinear executor. The oracle budget analysis, three-seed executor comparison, and fusion analysis consequently use distinct model selections and normalizations.

For commutator error, let $\widehat{\Delta}_i$ be the difference between predictive mean post-action responses for AB and BA on entity i. The relative RMS error is

$$\frac{\sqrt{\mathrm{mean}_{i,t,d}(\widehat{\Delta}_{i,t,d} - \Delta_{i,t,d})^2}}{\sqrt{\mathrm{mean}_{i,t,d}\,\Delta_{i,t,d}^2}}. \tag{22}$$

A zero predicted order effect therefore has relative error one. This metric is a relative RMS error, not an NMSE.

Diagnostic controls. Wrong-entity fusion has joint energy score 0.10217 and NMSE 0.8398 on the same test responses. Its order-effect error is unreported. The direct diagnostic, which bypasses the response chart and executor, has joint energy score 0.96737, NMSE 44.873, and order-effect error 0.47015. Its architecture is described above; it is distinct from the whole-program executor in Figure 3b.

The zero-order-effect control predicts $\widehat{\Delta} = 0$, giving relative order-effect error 1.00000. It specifies only the difference between the two orders, so it does not define complete response predictions for computing energy score or NMSE. Free-decay and hysteresis order-effect errors were not reported, and their test predictive trajectories were not retained. Table 1b therefore reports the four available population, fused, and point-belief comparisons.

C.6 Additional Controlled-System Diagnostics

The controlled system separates complementary sensory information from action-order prediction. The entity and program splits, physical parameters, response chart, and oracle threshold are fixed before evaluating test responses. Table 4 summarizes the physical and representation properties underlying this experiment. The blind directions isolate the information absent from each modality, while the factor-effect measurements show that the missing quantities affect the target post-action response. The AB/BA difference measures the effect of action order after the inputs stop.

The learned-model diagnostics use the 30,000-update convergence analysis. Oracle NMSE evaluates execution from exact response coordinates relative to the corresponding zero-coordinate prediction. The fused-information eigenvalue describes the directions constrained by the learned evidence factors; posterior trace measures the remaining coordinate uncertainty after combining them with the prior. Model performance comparisons appear in Figure 3 and Table 1.

Table 4: **Physical and representation diagnostics of the controlled system.** Blind-direction values are maximum evidence variations over the indicated parameter, with the other parameters fixed. Factor-effect RMS compares adjacent parameter levels in normalized test responses. Learned-model diagnostics use the 30,000-update training budget, with the executor selected by development error and fusion from the rank-two-factor model. The information eigenvalue excludes prior precision; trace reduction is relative to the smaller mean posterior trace of the two single modalities.

Diagnostic	Observed value	Interpretation
Free-decay variation over F_y	0	Free decay is invariant to yield force.
Hysteresis variation over c	0	Settled hysteresis is invariant to damping.
Target-response RMS from changing k, c, F_y	k: 0.09509 c: 0.03925 F_y: 0.09041	All three physical parameters affect the test response.
Chart relative residual	0.102265	Three coordinates retain 98.95% of centered fitting-response variation.
Oracle NMSE, dev / test	0.0756 / 0.1844	Exact coordinates give lower error than the zero-coordinate prediction.
AB/BA order-effect RMS	0.62016	Changing action order changes the zero-input response.
Fused information $\lambda_{\min}$	0.5867	The evidence factors jointly constrain all three chart directions.
Mean posterior-trace reduction	68%	Fusion reduces coordinate uncertainty relative to either single modality.